TPC Benchmark™ C Full Disclosure Report

for

IBM[®] **@server™** xSeries[®] 235 using

Microsoft® SQL Server 2000 Standard Edition and

Microsoft Windows® Server 2003 Standard Edition

TPC-C Version 5.1

Submitted for Review July 10, 2003



First Edition - July 2003

THE INFORMATION CONTAINED IN THIS DOCUMENT IS DISTRIBUTED ON AN AS IS BASIS WITHOUT ANY WARRANTY EITHER EXPRESSED OR IMPLIED. The use of this information or the implementation of any of these techniques is the customer's responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item has been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environment do so at their own risk.

In this document, any references made to an IBM licensed program are not intended to state or imply that only IBM's licensed program may be used; any functionally equivalent program may be used.

This publication was produced in the United States. IBM may not offer the products, services, or features discussed in this document in other countries, and the information is subject to change without notice. Consult your local IBM representative for information on products and services available in your area.

© Copyright International Business Machines Corporation 2003. All rights reserved.

Permission is hereby granted to reproduce this document in whole or in part, provided the copyright notice as printed above is set forth in full text on the title page of each item reproduced.

U.S. Government Users - Documentation related to restricted rights: Use, duplication, or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

Trademarks

IBM, the IBM logo,the IBM e-business logo, and xSeries are trademarks or registered trademarks of International Business Machines Corporation.

The following terms used in this publication are trademarks of other companies as follows: TPC Benchmark, tpmC, and \$/tpmC trademark of Transaction Processing Performance Council; Intel and Pentium are registered trademarks of Intel Corporation; Microsoft, Windows and BenchCraft are trademarks or registered trademarks of Microsoft Corporation. Other company, product, or service names, which may be denoted by two asterisks (**), may be trademarks or service marks of others.

Notes

- ¹ MHz only measures microprocessor internal clock speed, not application performance. Many factors affect application performance.
- ² When referring to hard disk capacity, GB, or gigabyte, means one thousand million bytes. Total user-accessible capacity may be less.

Abstract

IBM Corporation conducted the TPC BenchmarkTMC on the IBM @server xSeries 235 configured as a client/server system. This report documents the full disclosure information required by the TPC BenchmarkTMC Standard Specification, Revision 5.1, including the methodology used to achieve the reported results. All testing fully complied with this revision level.

The software used on the x235 system includes Microsoft Windows Server 2003 Standard Edition operating system and Microsoft SQL Server 2000 Standard Edition database.

Two standard metrics, transactions per minute-C (tpmC) and price per tpmC (\$/tpmC), are reported as required by the TPC Benchmark C Standard Specification.

The benchmark results are summarized in the following table.

Hardware	Software	Total System Cost	tpmC	\$/tpmC	Total Solution Availability Date
IBM @server xSeries 235	Microsoft SQL Server 2000 Standard Edition	\$46,539	18,936.05	\$2.46	July 10, 2003
	Microsoft Windows Server 2003 Standard Edition				

The results of the benchmark and test methodology used were audited by Bradley J. Askins of InfoSizing, Inc. The auditor's attestation letter is contained in Section 9 of this report.

7 01	IBM	@ser	verm xSe	eries® 23	5 c/s	TPC-C R	lev. 5.1	
	Mi	With Microsoft® SQL Server 2000 Report Date: July 10, 200					0, 2003	
Total System Cost		TPC-C Throughput Price/Performance Availability			ility Date			
\$46,539	18,	936.05	tpmC	\$2.4	6 / tp	mС	July 1	0, 2003
Processors Dat	abase Ma	pase Manager O		g System	Other Soft		ware	Number of Users
Database Server: 1 Intel® Xeon™ DP 3.06GHz Client: 1 Xeon DP 2.4GHz Microsoft SQL Server 2000 Standard Edition Microsoft Windows® Server 2003 Standard Edition Microsoft Visual C++ Microsoft COM+ 1				15,300				
1 x IBM xSeries 235 1 x IBM xSeries 235 1 x 3.06GHz/512KB Xeon DP 3 x Mylex AcceleRAID A352 Adapters 1 x IBM xSeries 225 1 x 2.4GHz Xeon DP 2 x 73.4GB Drive 2 x 73.4GB Drives 3 x EXP300 Storage 42 x18.2GB Drives								
System Component Processors Cache Memory Disk Controllers Disk Drives Total Storage Tape Drive	Qty 1 2 2 3 1 42 2 1	512KF 256MI 1GB F RDIM Mylex Adapt Integr 18.2GI 73.4GI 18.2GI 863.43	Hz Xeon D B L2 Cache B CCC SDRA M AcceleRA er with 32 ated Ultra B (15000 r B (10000 r	M ID A352 MB 320 SCSI pm) pm) pm)	Qty 1 4 1 1 1	512KB 1 256MB Ultra32	Xeon DP L2 Cache 0 SCSI II (10000 rj	nterface

IBM Corporation

IBM @server xSeries 235 c/s with Microsoft SQL Server 2000

TPC-C Rev. 5.1

Report Date: July 10, 2003

Description	Order Number	Thir Brand	d-Party Pricing	Unit Price	Qty	Ext. Price	3-Yr. Maint.*
		Diana	og	11100			manic.
Server Hardware							
xSeries 235 w/ 1 x 3.06GHz/512KB Xeon DP 2 x 256MB PC2100 ECC SDRAM	8671-81X	IBM	1	\$2,949	1	\$2,949	\$689
Integrated Dual-Channel Ultra320 SCSI					_	4 0 4 0	
1GB ECC SDRAM RDIMM Memory Kit	33L5039	IBM	1	659	2	1,318	0
18.2GB 10K Ultra160 SCSI Drive	06P5754	IBM	1	275	1	275	C
Mylex AcceleRAID A352 SCSI Adapter (+2)	08P2420	Mylex	3	580	5	2,900	C
Netfinity 4.2M Ultra2 SCSI Cable	03K9311	IBM	1 1	105	6	630	90
E54 15" (13.8" Viewable) Color Monitor* 20/40GB Internal SCSI Tape Drive	633147N 00N7991	IBM IBM	1	129 699	1 1	129 699	90
Storage Hardware	00117 99 1	IDIVI	ı	099	,	099	
EXP300 Rack Storage Enclosure*	3531-1RU	IBM	1	3,179	3	9,537	600
18.2GB 15K Ultra160 SCSI Drive	06P5767	IBM	1	329	42	13,818	000
73.4GB 10K Ultra160 SCSI Drive	06P5756	IBM	1	639	2	1,278	0
NetBAY11 Enterprise Rack*	9306110	IBM	1	519	1	519	168
NetBATTT Enterprise Nack	9300110	IDIVI	•		ubtotal	\$34,052	\$1,547
Server Software				•	ubiolai	ψ04,002	Ψ1,041
	000 04070		•	4 000		04.000	
Microsoft SQL Server 2000 Standard Edition		Microsoft	2	4,999	1	\$4,999	\$0
Microsoft Windows Server 2003 SE	P73-00295	Microsoft	2	738	1	738	0
Three-Year Maintenance for Software		Microsoft	2	1,950	3 _		5,850
				S	ubtotal	\$5,737	\$5,850
Client Hardware							
xSeries 225 / 2.4GHz/512KB Xeon DP* 2 x 256MB PC2100 ECC SDRAM	8647-3AX	IBM	1	1,269	1	\$1,269	\$689
256MB PC2100 ECC SDRAM RDIMM	33L5037	IBM	1	165	2	330	0
18.2GB 10K Ultra160 SCSI Drive	06P5754	IBM	1	275	1	275	0
IBM 10/100 Dual-Port Server Adapter	22P4901	IBM	1	209	2	418	0
E54 15" (13.8" Viewable) Color Monitor*	633147N	IBM	1	129	1	129	90
				S	ubtotal	\$2,421	\$779
Client Software							
Microsoft Windows 2000 Server with COM+	C11-00821	Microsoft	2	738	1	738	0
Microsoft Visual C++ Standard	254-00170	Microsoft	2	109	1 _	109	0
				S	ubtotal	\$847	\$0
User Connectivity							
Cross-over Cable (7 ft.)*** (+2)	CBLC5C7		4	2	3	6	0
				S	ubtotal	\$6	\$0
					Total	\$43,063	\$8,176
IBM hardware discount of 14%; prices vary			1		-	\$4,700	\$0
Notes: * The standard 3-year warranties on IE upgraded to 7x24, 4-hour response time cov		e peen	Three-Yea	ar Cost of	Owners	hip:	\$46,539
** Five-year warranty. *** 10% or minimum 2		d in place of					
on-site service (products have a 5-year return				tp	mC Rati	ng: 18	3,936.05
1 - IBM List Price, 1-866-426-0472; 2 - Micros	soft Corp.; 3 - Cor	mputer					
Giants; 4 - LanAdapters.com	_				\$ / tpi	mC:	\$2.46
Audited by Bradley J. Askins of InfoSizing, Inc. Prices used in TPC benchmarks reflect the actual		uauld m £		aga of 41 '	atad	nomont-	

Prices used in TPC benchmarks reflect the actual prices a customer would pay for a one-time purchase of the stated components. Individually negotiated discounts are not permitted. Special prices based on assumptions about past or future purchases are not permitted. All discounts reflect standard pricing policies for the listed components. For complete details, see the pricing sections of the TPC benchmark specification. If you find that stated prices are not available according to these terms, please inform the TPC at pricing@tpc.org. Thank you.

Numerical Quantities Summary						
MQTh, Computed Maximum Qualified Throughput: 18,936.05 tpmC						
Response Times (in seconds)	90th Percentile	Average	Maximum			
New-Order	0.82	0.56	6.38			
Payment	0.54	0.32	5.08			
Delivery	0.41	0.21	1.03			
Stock Level	4.54	2.54	6.92			
Order Status	0.75	0.49	7.65			
Delivery (Deferred)	1.85	1.25	4.75			
Menu	0.42	0.22	1.08			
Response Time Delay Add	ded for Emulated Compone	nts: 0.1 Seconds				
Transaction Mix (in percent of total transactions) Percent						
New-Order	44.87					
0.40.			11.07			
Payment			43.00			
11011 010101						
Payment			43.00			
Payment Delivery			43.00 4.06			
Payment Delivery Stock-Level	Minimum	Average	43.00 4.06 4.03			
Payment Delivery Stock-Level Order Status Keying/Think Times	Minimum 18.00 / 0.00	Average 18.01 / 12.04	43.00 4.06 4.03 4.04			
Payment Delivery Stock-Level Order Status Keying/Think Times (in seconds)	-		43.00 4.06 4.03 4.04 Maximum			
Payment Delivery Stock-Level Order Status Keying/Think Times (in seconds) New Order	18.00 / 0.00	18.01 / 12.04	43.00 4.06 4.03 4.04 Maximum 18.03 / 120.50			

Order Status	2.00 / 0.00	2.01 / 10.01	2.02 / 100.50			
Test Duration						
Ramp-up time	30 minutes 15 seconds					
Measurement interval	120 minutes					
Number of checkpoints	4					
Checkpoint interval	30 minutes					
Number of transactions (al	Number of transactions (all types) completed in measurement interval					

Table of Contents

Abstract	
Numerical Quantities Summary	
Preface	11
General Items	
Application Code Disclosure and Definition Statements	
Benchmark Sponsor	
Parameter Settings	
Configuration Diagrams	
Clause 1: Logical Database Design Related Items	
Table Definitions	
Physical Organization of the Database	
Insert and Delete Operations	
Horizontal or Vertical Partitioning	
Replication	
Table Attributes	14
Clause 2: Transaction and Terminal Profiles Related	
Items	
Random Number Generation	
Screen Layout	
Terminal Verification	
Intelligent Terminals	
Transaction Profiles	
Deferred Delivery Mechanism	16
Clause 3: Transaction and System Properties Related	
Items	
Atomicity Requirements	
Consistency Requirements	
Isolation Requirements	
Durability Requirements	
Clause 4: Scaling and Database Population Related Items	
Cardinality of Tables	
Distribution of Tables and Logs	
Database Model Implemented	
Partitions/Replications Mapping	
60-Day Space Requirement	21
Clause 5: Performance Metrics and Response Time	
Related Items	
Measured tpmC	
Response Times	
Keying/Think Times	
Response Time Frequency Distribution Curves	
Performance Curve for Response Time vs. Throughput	
New Order Think Time Distribution	
Throughput vs. Elapsed Time	
Steady State Methodology	
Work Performed during Steady State	
Checkpoints	
Measurement Interval	
Transaction Mix	
Percentage of Total Mix	
Number of Checkpoints	28
Clause 6: SUT, Driver and Communication Definition	
Related Items	29

Description of RTE	
Emulated Components	
Benchmarked and Targeted System Configuration Diagrams	
Network Configuration	
Network Bandwidth	
Operator Intervention	
Clause 7: Pricing Related Items	
Hardware and Software Components	30
Availability Date	30
Measured tpmC	
Country-Specific Pricing	30
Usage Pricing	30
System Pricing	31
Clause 9: Audit Related Items	32
Auditor	32
Availability of the Full Disclosure Report	32
Attestation letter	33
Appendix A: Source Code	35
client utils.c	
client utils.h	
dlldata.c	
error.h	
install.c	
install.h	
install.rc	
install com.cpp	
license.txt	
mon client.c	
mon client.h	
readme.txt	
readregistry.cpp	
readregistry.h	
resource.h	
resource tpcc rc.h	
rtetime.h	
spinlock.h	
tpcc.cpp	
tpcc.def	
tpcc.h	
tpcc.rc	
tpcc com.cpp	
tpcc com.h	
tpcc com all.dsp	
tpcc_com_un.usptpcc_com_ps.def	
tpcc_com_ps.h	
tpcc_com_ps.id	
tpcc com ps i.c	
tpcc_com_ps_p.c	
tpcc_dblib.cpptncc_dblib.k	114
T	124
T · = · · · · FF	_
tpcc_enc.h	
tpcc_odbc.cpp	
tpcc_odbc.h	
tpcc_tux.cpp	13/

tpcc_tux.h	139
trans.h	140
tuxapp.cpp	141
tuxapp.h	145
tuxmain.c	145
txn base.h	146
txnlog.h	147
webclnt.dsp	149
webclnt.dsw	150
Stored Procedures	152
neword.sql	152
payment.sql	
ordstat.sql	
delivery.sqldelivery.sql	
stocklev.sqlstocklev.sql	157
version.sql	158
null-txn.sql	
Appendix B: Database Design	
Database Build	
backup.sql	
backupdev.sql	
createdb.sql	
dbopt1.sql	
dbopt2.sql	
idxcuscl.sql	
idxcusnc.sql	
idxdiscl.sql	
idxitmcl.sql	
idxnodcl.sql	
idxodlcl.sql	
idxordcl.sql	
idxordnc.sql	
idxstkcl.sql	
idxwarcl.sql	
removedb.sql	
restore.sql	
RunSQLcfg.sql	
sqlshutdown.sql	
tables.sql	
verify TpccLoad.sql	
version.sql	
Load Source Code	
getargs.c	
random.c	
strings.c	
time.c	
tpcc.h	
tpccldr.c	
tpccldr.mak	
Appendix C: Tunable Parameters	
Microsoft SQL Server 2000	
Configuration Parameters	
Microsoft Windows Server 2003	
System Information Report for the x235	
Disk Controller Configuration Parameters	
210h Commonor Cominguration i aramicació	250

Integrated Ultra320 Interface	230
Mylex AcceleRAID A352 Adapter 1	231
Mylex AcceleRAID A352 Adapter 2	232
Mylex AcceleRAID A352 Adapter 3	233
Microsoft Windows 2000 Server with COM+	234
Client Configuration Parameters	234
Client System Information Report	235
RTE Input Parameters	256
Appendix D: 60-Day Space	261
Appendix E: Third-Party Ouotations	262

Preface

The TPC Benchmark $^{\text{TM}}$ C was developed by the Transaction Processing Performance Council (TPC). The TPC was founded to define transaction processing benchmarks and to disseminate objective, verifiable performance data to the industry. This full disclosure report is based on the TPC Benchmark C Standard Specification Version 5.0.

The TPC describes this benchmark in Clause 0.1 of the specification as follows:

TPC Benchmark C is an On Line Transaction Processing (OLTP) workload. It is a mixture of read-only and update-intensive transactions that simulate the activities found in complex OLTP application environments. It does so by exercising a breadth of system components associated with environments, which are characterized by:

- The simultaneous execution of multiple transaction types that span a breadth of complexity
- On-line and deferred transaction execution modes
- · Multiple on-line terminal sessions
- Moderate system and application execution time
- · Significant disk input/output
- Transaction integrity (ACID properties)
- · Non-uniform distribution of data access through primary and secondary keys
- Databases consisting of many tables with a wide variety of sizes, attributes and relationships
- Contention on data access and update

The performance metric reported by TPC-C is a "business throughput" measuring the number of orders processed per minute. Multiple transactions are used to simulate the business activity of processing an order, and each transaction is subject to a response time constraint. The performance metric for this benchmark is expressed in transactions-per-minute-C (tpmC). To be compliant with the TPC-C standard, all references to tpmC results must include the tpmC rate, the associated price-per-tpmC, and the availability date of the priced configuration.

Despite the fact that this benchmark offers a rich environment that emulates many OLTP applications, this benchmark does not reflect the entire range of OLTP requirements. In addition, the extent to which a customer can achieve the results reported by a vendor is highly dependent on how closely TPC-C approximates the customer application. The relative performance of systems derived from this benchmark does not necessarily hold for other workloads or environments. Extrapolations to any other environment are not recommended.

Benchmark results are highly dependent upon workload, specific application requirements, and systems design and implementation. Relative system performance will vary as a result of these and other factors. Therefore, TPC-C should not be used as a substitute for a specific customer application benchmarking when critical capacity planning and/or product evaluation decisions are contemplated.

General Items

Benchmark Sponsor

A statement identifying the benchmark sponsor(s) and other participating companies must be provided.

This benchmark was sponsored by International Business Machines Corporation.

Application Code Disclosure and Definition Statements

The application program (as defined in Clause 2.1.7) must be disclosed. This includes, but is not limited to, the code implementing the five transactions and the terminal input and output functions.

Appendix A contains all source code implemented in this benchmark.

Parameter Settings

Settings must be provided for all customer-tunable parameters and options that have been changed from the defaults found in actual products, including but not limited to:

- Database tuning options
- Recovery/commit options
- Consistency/locking options
- Operating system and application configuration parameters.
- Compilation and linkage options and run-time optimizations used to create/install applications, OS, and/or databases.

This requirement can be satisfied by providing a full list of all parameters and options.

Appendix C contains the tunable parameters for the database, the operating system, and the transaction monitor.

Configuration Diagrams

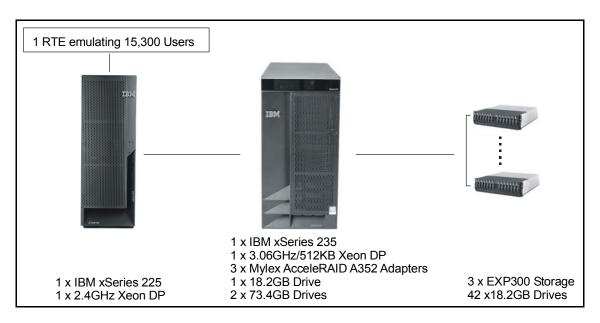
Diagrams of both measured and priced configurations must be provided, accompanied by a description of the differences.

The configuration diagrams for the tested and priced systems are provided on the following pages.

The Remote Terminal Emulator (RTE) used for these TPC Benchmark C tests is the Microsoft BenchCraft RTE. Under Version 5.0, the components of the configuration being emulated by the RTE are the workstations and the Ethernet hubs. Appendix C contains a listing of the RTE scripts and inputs used in the benchmark testing.

The benchmarked configuration used an IBM xSeries 225 system as the client, which executed the terminal I/O and submitted transactions to COM+ servers, which are also running on the clients. These COM+ servers forwarded the transaction requests to the server, and returned the results to the RTE. Microsoft SQL Server 2000 Standard Edition is the DBMS executing on the server

Measured and Priced Configuration



The measured and priced configurations were identical.

Clause 1: Logical Database Design Related Items

Table Definitions

Listings must be provided for all table definition statements and all other statements used to set up the database. Appendix B contains the code used to define and load the database tables.

Physical Organization of the Database

The physical organization of tables and indexes within the database must be disclosed. Physical space was allocated to Microsoft SQL Server on the server disks as detailed in Figure 4-2.

Insert and Delete Operations

It must be ascertained that insert and/or delete operations to any of the tables can occur concurrently with the TPC-C transaction mix. Furthermore, any restriction in the SUT database implementation that precludes inserts beyond the limits defined in Clause 1.4.11 must be disclosed. This includes the maximum number of rows that can be inserted and the maximum key value for these new rows.

All insert and delete functions were fully operational during the running of the benchmark. The space required for an additional 5 percent of the initial table cardinality was allocated to Microsoft SQL Server 2000 and priced as static space.

Horizontal or Vertical Partitioning

While there are few restrictions placed upon horizontal or vertical partitioning of tables and rows in the TPC-C benchmark (see Clause 1.6), any such partitioning must be disclosed.

Partitioning was not used in this benchmark.

Replication

Replication tables, if used, must be disclosed (see Clause 1.4.6).

Replication was not used in this benchmark.

Table Attributes

Additional and/or duplicated attributes in any table must be disclosed, along with a statement on the impact on performance (see Clause 1.4.7).

No additional attributes were used in this benchmark.

Clause 2: Transaction and Terminal Profiles Related Items

Random Number Generation

The method of verification for the random number generation must be disclosed.

The seeds and offsets for the random number generator were collected and verified to be different for each driver. The auditor selected samples of the generated numbers from the database. The samples were verified to have no discernible patterns.

Screen Layout

The actual layouts of the terminal input/out screens must be disclosed.

All screen layouts followed the TPC Benchmark C Standard Specification.

Terminal Verification

The method used to verify that the emulated terminals provide all the features described in Clause 2.2.2.4 must be explained. Although not specifically priced, the type and model of the terminals used must for the demonstration in 8.1.3.3 must be disclosed and commercially available (including supporting software and maintenance).

The auditor verified terminal features by direct experimentation. The benchmarked configuration uses Microsoft Internet Explorer 5.0 and HTML scripts as the terminal interface.

Intelligent Terminals

Any usage of presentation managers or intelligent terminals must be explained.

The terminals emulated in the priced configuration are IBM PC desktop computer systems. All processing of the input/output screens was handled by the xSeries 225 client. The screen input/output was managed via HTML strings that comply with the HTML Version 2.0 specification. A listing of the code used to implement the intelligent terminals is provided in Appendix A. All data manipulation was handled by the xSeries 235 database server.

Transaction Profiles

The percentage of home and remote order-lines in the New-Order transactions must be disclosed. The percentage of New-Order transactions that were rolled back as a result of an unused item number must be disclosed.

The number of items per orders entered by New-Order transactions must be disclosed. The percentage of home and remote Payment transactions must be disclosed. The percentage of Payment and Order-Status transactions that used non-primary key (C LAST) access to the database must be disclosed.

The percentage of Delivery transactions that were skipped as a result of an insufficient number of rows in the NEW-ORDER table must be disclosed.

The mix (i.e., percentages) of transaction types seen by the SUT must be disclosed.

Table 2-1. Transaction Statistics

New Order	Value (%)
Home warehouse order lines	99.00
Remote warehouse order lines	1.00
Rolled back transactions	0.98
Average number of items per order	10.00
Payment	
Home warehouse payment transactions	85.06
Remote warehouse payment transactions	14.94
Non-Primary Key Access	
Payment transactions using C_LAST	60.03
Order-Status transactions using C_LAST	59.89
Delivery	
Delivery transactions skipped	0
Transaction Mix	
New-Order	44.87
Payment	43.00
Delivery	4.06
Stock Level	4.03
Order Status	4.04

Deferred Delivery Mechanism

The queuing mechanism used to defer execution of the Delivery transaction must be disclosed.

The deferred delivery operation is queued by making an entry in an array within the application process (tpcc.dll) running on the client. Background threads within the application asynchronously process the queued delivery transactions.

The source code is listed in Appendix A.

Clause 3: Transaction and System Properties Related Items

The results of the ACID test must be disclosed, along with a description of how the ACID requirements were met. This includes disclosing which case was followed for the execution of Isolation Test 7.

Atomicity Requirements

The system under test must guarantee that database transactions are atomic; the system will either perform all individual operations on the data, or will assure that no partially completed operations leave any effects on the data.

All ACID tests were conducted according to specification.

Completed Transactions

The following steps were performed to verify the Atomicity of completed transactions.

- 1. The balance was retrieved from the CUSTOMER table for a random Customer, District and Warehouse, giving BALANCE 1.
- 2. The Payment transaction was executed for the Customer, District and Warehouse used in step 1.
- 3. The balance was retrieved again for the Customer used in step 1 and step 2, giving BALANCE_2. It was verified that BALANCE_1 was greater than BALANCE_2 by AMT.

Aborted Transactions

The following steps were performed to verify the Atomicity of the aborted Payment transaction:

- 1. The Payment application code was changed to execute a rollback of the transaction instead of performing the commit.
- 2. Using the balance, BALANCE_2, from the CUSTOMER table retrieved for the completed transaction, the Payment transaction was executed for the Customer, District and Warehouse used in step 1 of section 3.1.1. The transaction rolled back due to the change in the application code from step 1.
- 3. The balance was retrieved again for the Customer used for step 2, giving BALANCE_3. It was verified that BALANCE_2 was equal to BALANCE_3.

Consistency Requirements

Consistency is the property of the application that requires any execution of a database transaction to take the database from one consistent state to another, assuming that the database is initially in a consistent state.

Consistency conditions one through four were tested using a shell script to issue queries to the database. The results of the queries demonstrated that the database was consistent for all four tests.

Isolation Requirements

Sufficient conditions must be enabled at either the system or the application level to ensure that the required isolation defined in Clause 3.4.1 is obtained.

Isolation tests one through seven were run using shell scripts to issue queries to the database. Each script included timestamps to demonstrate the concurrency of operations. The results of the queries were captured and placed in files. The auditor reviewed the results and verified that the isolation requirements had been met.

In addition, the phantom tests and the stock-level tests were run and verified.

Case A was followed for Isolation test seven.

Durability Requirements

The tested system must guarantee durability: the ability to preserve the effects of committed transactions and ensure database consistency after recovery from any one of the failures listed in Clause 3.5.3.

- Permanent irrecoverable failure of any single durable medium containing TPC-C database tables or recovery log data (this test includes failure of all or part of memory)
- Instantaneous interruption (system crash/system hang) in processing that requires system reboot to recover
- Failure of all or part of memory (loss of contents)

Loss of Data Test

The following steps were successfully performed to pass the Durability test of failure of a disk unit with database tables:

- 1. The contents of the database were backed up to several database dump devices during the initial database load. There were no dump devices on the disk array from which a drive was removed as part of this test.
- 2. The current count of the total number of orders was determined by the sum of D_NEXT_O_ID for all rows in the district table giving SUM1.
- 3. A test was started with 10 percent of the total users submitting transactions.
- 4. A disk containing a portion of each of the tables in the tpcc database was removed causing SQL Server to report errors accessing that device.
- 5. The run was aborted and SQL Server was restarted. Upon restart, the database tpcc reported numerous errors relating to the failed database device.
- 6. The transaction log was dumped to disk and the failed disk was replaced with a spare disk and was recovered.
- 7. The database was recovered and restored from the backup dump devices. Afterwards, the transaction log was applied to the database.
- 8. Step 2 was repeated to obtain the current count of the total number of orders giving SUM2.
- 9. It was verified that the sum of D_NEXT_O_ID after the database is recovered is greater than or equal to the sum of D_NEXT_O_ID before the run, plus all new order transactions completed during the run minus any rollback transactions.
- 10. Consistency Condition 3 was verified.

Combined Loss of Log and Loss of System Test (Instantaneous Interruption and Loss of Memory)

- 1. The current count of the total number of orders was determined by the sum of D_NEXT_O_ID for all rows in the district table gving SUM1.
- 2. The test started with a full load with all users submitting transactions. A checkpoint was issued, and the system continued to run for another 5 minutes.
- 3. One of the log disk drives was removed. Since the log disk was mirrored, SQL Server continued to process transactions without interruption.
- 4. The test continued for another 3 minutes.
- 5. The server under test was powered off, which removed power from the system and the memory.
- 6. The server was powered on again.
- 7. SQL Server was started to initiate automatic recovery from its log.
- 8. Step 1 was repeated to obtain the current count of the total number of orders giving SUM2.
- 9. It was verified that the sum of D_NEXT_O_ID after the database is recovered is greater than or equal to the sum of D_NEXT_O_ID before the run, plus all new order transactions completed during the run minus any rollback transactions.

Clause 4: Scaling and Database Population Related Items

Cardinality of Tables

The cardinality (e.g., the number of rows) of each table, as it existed at the start of the benchmark run (see Clause 4.2), must be disclosed. If the database was over-scaled and inactive rows of the WAREHOUSE table were deleted (see Clause 4.2.2), the cardinality of the WAREHOUSE table as initially configured and the number of rows deleted must be disclosed.

The database was originally built with 1,530 warehouses, and the audited run used all 1,530 warehouses.

Table 4-1. Initial Cardinality of Tables

Table Name	Rows
Warehouse	1,530
District	15,300
Item	100,000
New Order	13,770,000
History	45,900,000
Orders	45,900,000
Customer	45,900,000
Order Line	459,001,971
Stock	153,000,000
Inactive Warehouses	0

Distribution of Tables and Logs

The distribution of tables and logs across all media must be explicitly depicted for the tested and priced systems.) Figure 4-2 depicts the database configuration of the tested system to meet the 8-hour steady state requirement.

Figure 4-2. Data Distribution for the Benchmarked Configuration

Controller	Drives	Partition	Size	Use
0	2 - 73.4GB	E:	45000MB	Logfile
1	14 - 18.2GB	F: G:	29200MB 14000MB	Customer and Stock Misc.
2	14 - 18.2GB	H: I: Y:	29200MB 14000MB 80000MB (NTFS)	Customer and Stock Misc. Backup 1
3	14 - 18.2GB	J: K: Z:	29200MB 14000MB 80000MB (NTFS)	Customer and Stock Misc. Backup 2

Database Model Implemented

A statement must be provided that describes:

- 1. The database model implemented by the DBMS used (e.g., relational, network, hierarchical)
- 2. The database interface (e.g., embedded, call level) and access language (e.g., SQL, DL/1, COBOL, read/write) used to implement the TPC-C transactions. If more than one interface/access language is used to implement TPC-C, each interface/access language must be described and a list of which interface/access language is used with which transaction type must be disclosed.

Microsoft SQL Server 2000 Standard Edition is a relational database. The interface used was Microsoft SQL Server stored procedures accessed with Remote Procedure Calls embedded in C code using the Microsoft DBLIB interface.

Partitions/Replications Mapping

The mapping of database partitions/replications must be explicitly described.

The database was neither partitioned nor replicated.

60-Day Space Requirement

Details of the 60-day space computations, along with proof that the database is configured to sustain 8 hours of growth for the dynamic tables (Order, Order-Line, and History) must be disclosed (see Clause 4.2.3).

See Appendix D for details about how the 60-day space requirements were calculated.

Clause 5: Performance Metrics and Response Time Related Items

Measured tpmC

Measured tpmC must be reported.

Measured tpmC: 18,936.05 tpmC

Price per tpmC: \$2.46 per tpmC

Response Times

Ninetieth percentile, maximum and average response times must be reported for all transaction types as well as for the Menu response time.

The TPC-C requirements for the average response time and the 90th percentile were met. Table 5-1 provides the response times for each of the transaction types and the menu for the measured system.

Table 5-1. Response Times in Seconds

Transaction Type	Average	Maximum	90 %-tile
New-Order	0.56	6.38	0.82
Payment	0.32	5.08	0.54
Delivery	0.21	1.03	0.41
Stock Level	2.54	6.92	4.54
Order Status	0.49	7.65	0.75
Delivery (Deferred)	1.25	4.75	1.85
Menu	0.22	1.08	0.42

Keying/Think Times

The minimum, the average, and the maximum keying and think times must be reported for each transaction type.) Table 5-2 lists the keying/think times for the measured system.

Table 5-2. Keying/Think Times

Transaction Type	Average	Minimum	Maximum	
New-Order	18.01 / 12.04	18.01 / 12.04 18.00 / 0.00		
Payment	3.01 / 12.04	3.00 / 0.00	3.03 / 120.50	
Delivery	2.01 / 5.03	2.00 / 0.00	2.04 / 50.50	
Stock Level	2.01 / 5.03	2.00 / 0.00	2.03 / 50.50	
Order Status	2.01 / 10.01	2.00 / 0.00	2.02 / 100.50	

Response Time Frequency Distribution Curves

Response time frequency distribution curves (see Clause 5.6.1) must be reported for each transaction type.

Figure 5-1. New-Order Transaction - Response Time Frequency Distribution

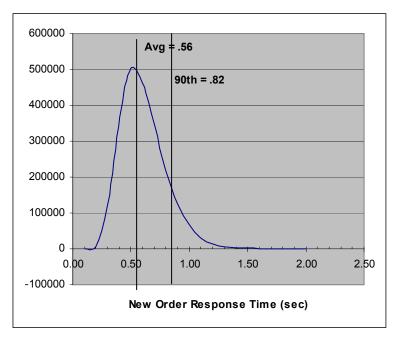


Figure 5-2. Payment Transaction - Response Time Frequency Distribution

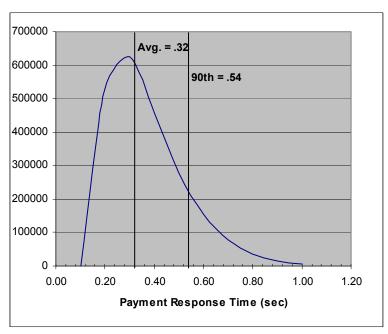


Figure 5-3. Order-Status Transaction - Response Time Frequency Distribution

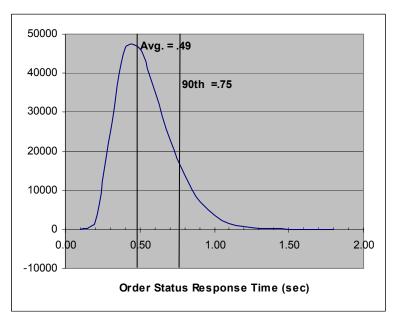
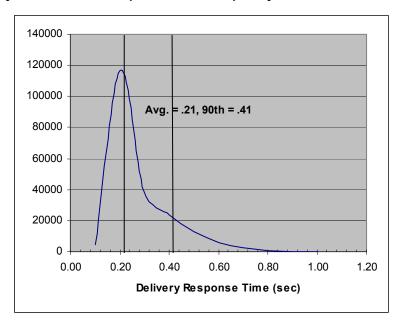


Figure 5-4. Delivery Transaction - Response Time Frequency Distribution



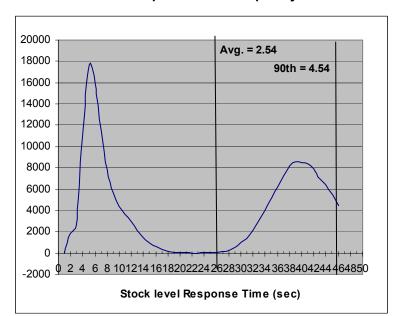
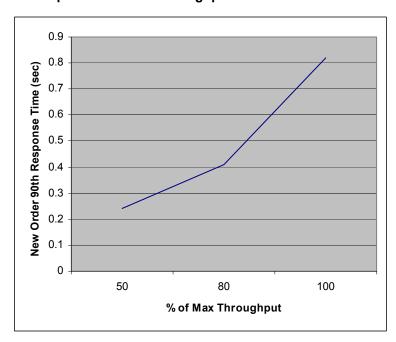


Figure 5-5. Stock-Level Transaction - Response Time Frequency Distribution

Performance Curve for Response Time vs. Throughput

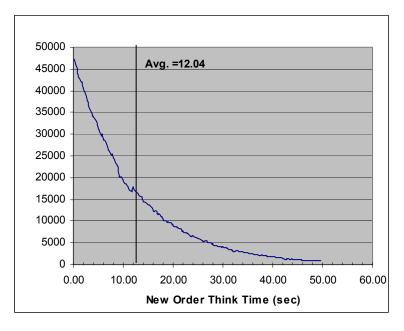
The performance curve for response time vs. throughput (see Clause 5.6.2) must be reported for the New-Order transaction.





New Order Think Time Distribution

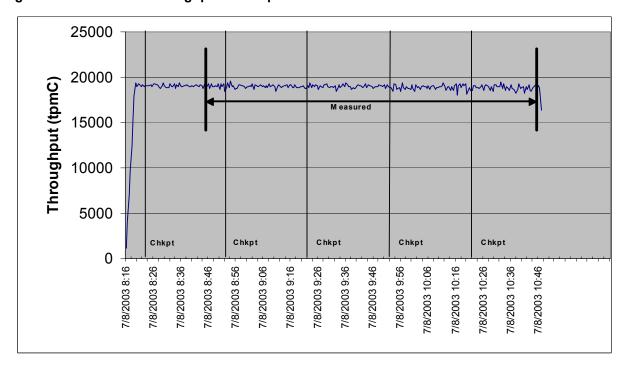
Figure 5-7. New-Order Think Time Distribution



Throughput vs. Elapsed Time

A graph of throughput vs. elapsed time (see Clause 5.6.5) must be reported for the New-Order transaction.

Figure 5-8. New-Order Throughput vs. Elapsed Time



Steady State Methodology

The method used to determine that the SUT had reached a steady state prior to commencing the measurement interval (see Clause 5.5) must be described.

Figure 5-8 shows that the system was in steady state at the beginning of the measurement interval.

Work Performed during Steady State

A description of how the work normally performed during a sustained test (e.g., checkpointing, writing redo/undo log records) actually occurred during the measurement interval must be reported.

Transaction Flow

The RTE generated the required input data to choose a transaction from the menu. This data was time-stamped. The response for the requested transaction was verified and time-stamped in the RTE log files.

The RTE generated the required input data for the chosen transaction. It waited to complete the minimum required key time before transmitting the input screen. The transmission was time-stamped. The return of the screen with the required response data was time-stamped. The difference between these two time-stamps was the response time for that transaction and was logged in the RTE log. The RTE then waited the required think time interval before repeating the process starting at selecting another transaction from the menu.

The RTE transmissions were sent to application processes running on the client machines through Ethernet LANs. These client application processes handled all screen I/O as well as all requests to the database on the server. The applications communicated with the database server over another Ethernet LAN using Microsoft SQL Server DBLIB library and RPC calls.

Checkpoints

Checkpoints were executed on the server during the ramp-up phase and at 30-minute intervals. The measured run contained four checkpoints. SQL Server was started with trace flag 3502, which caused it to log the occurrence of the checkpoint. This information was used to verify that the checkpoints occurred at the appropriate times during the test run.

During a checkpoint, SQL Server flushes all dirty pages from its cache to disk. It places a record in the database transaction log indicating that the checkpoint has completed and that all transactions, which were committed prior to the checkpoint have been written to disk.

Measurement Interval

A statement of the duration of the measurement interval for the reported Maximum Qualified Throughput (tpmC) must be included.

The measurement interval was 120 minutes.

Transaction Mix

The method of regulation of the transaction mix (e.g., card decks or weighted random distribution) must be described. If weighted distribution is used and the RTE adjusts the weights associated with each transaction type, the maximum adjustments to the weight from the initial value must be disclosed. (8.1.6.13)

See Table 5-3

The RTE was given a weighted random distribution, which was not adjusted during the run.

© IBM Corporation TPC Benchmark C Full Disclosure Report - July 2003

Percentage of Total Mix

The percentage of the total mix for each transaction type must be disclosed. See Table 5-3.

Table 5-3. Transaction Statistics and Transaction Mix

New Order	Value (%)
Home warehouse order lines	99.00
Remote warehouse order lines	1.00
Rolled back transactions	0.98
Average number of items per order	10.00
Payment	
Home warehouse payment transactions	85.06
Remote warehouse payment transactions	14.94
Non-Primary Key Access	
Payment transactions using C_LAST	60.03
Order-Status transactions using C_LAST	59.89
Delivery	
Delivery transactions skipped	0
Transaction Mix	
New-Order	44.87
Payment	43.00
Delivery	4.06
Stock Level	4.03
Order-Status	4.04

Number of Checkpoints

The number of checkpoints in the Measurement Interval, the time in seconds from the start of the Measurement Interval to the first checkpoint, and the Checkpoint Interval must be disclosed.

Checkpoints were performed during the ramp-up period and during each measured run interval. The first measurement interval checkpoint started 5 minutes and 3 seconds after the start of the measurement interval. The four checkpoints in the measured interval are shown in Table 5-4.

Table 5-4. Checkpoint Start Time and Duration

Checkpoint	Start Time Duration	
1	08:52:03 a.m.	14 minutes 38 seconds
2	09:22:01 a.m.	15 minutes 52 seconds
3	09:51:59 a.m.	16 minutes 49 seconds
4	10:21:57 a.m.	17 minutes 20 seconds

The checkpoint interval was 30 minutes.

Clause 6: SUT, Driver and Communication Definition Related Items

Description of RTE

The RTE input parameters, code fragments, functions, etc., used to generate each transaction input field must be disclosed.

The RTE used was Microsoft BenchCraft 1.1.2 RTE. Benchcraft is a proprietary tool provided by Microsoft and is not commercially available. The RTE input is listed in Appendix C.

Emulated Components

It must be demonstrated that the functionality and performance of the components being emulated in the Driver System are equivalent to that of the priced system. The results of the test described in Clause 6.6.3.4 must be disclosed.

No components were emulated.

Benchmarked and Targeted System Configuration Diagrams

A complete functional diagram of both the benchmarked configuration and the configuration of the proposed (target) system must be disclosed. A detailed list of all software and hardware functionality being performed on the Driver System, and its interface to the SUT must be disclosed (see Clause 6.6.3.6).

The driver RTE generated the transaction input data and transmitted it to the client in HTML format. The driver RTE received the output from the System under Test, time-stamped it, and forwarded it to the Master RTE for post-test processing. No other functionality was included on the driver RTE.

Detailed diagrams of the benchmarked and priced configurations are provided in the section called "General Items" at the beginning of this document.

Network Configuration

The network configurations of both the tested services and the proposed (target) services which are being represented and a thorough explanation of exactly which parts of the proposed configuration are being replaced with the Driver System must be disclosed (see Clause 6.6.4).

See the measured and priced configuration diagrams (pages 13 and 14) for details about the network configuration.

Network Bandwidth

The bandwidth of the network(s) used in the tested/priced configuration must be disclosed.

The Ethernet used in the LAN complies with the IEEE.802.3 standard. The LANs that connected the driver RTEs to the clients had a bandwidth of 10Mbps. The LAN that connected the client to the server had a bandwidth of 100Mbps.

Operator Intervention

If the configuration requires operator intervention (see Clause 6.6.6), the mechanism and the frequency of this intervention must be disclosed.

The configuration did not require any operator intervention to sustain the reported throughput.

Clause 7: Pricing Related Items

Hardware and Software Components

A detailed list of the hardware and software used in the priced system must be reported. Each separately orderable item must have a vendor part number, description and release/revision level, and either general availability status or committed delivery date. If package-pricing is used, vendor part number of the package and a description uniquely identifying each of the components of the package must be disclosed.

Pricing source(s) and effective date(s) must also be reported.

The total 3-year price of the entire configuration must be reported, including: hardware, software, and maintenance charges. Separate component pricing is recommended. The basis of all discounts used must be disclosed.

A detailed list of all hardware and software, including the 3-year price, is provided in the Executive Summary at the front of this report. All third-party quotations are included in Appendix E at the end of this document.

Availability Date

The committed delivery date for general availability (availability date) of products used in the price calculations must be reported. When the priced system includes products with different availability dates, the reported availability for the priced system must be the date at which all components are committed to be available.

All hardware and software used in this benchmark are currently available.

Measured tpmC

A statement of the measured tpmC, as well as the respective calculations for the 3-year pricing, price/performance (price/tpmC) and the availability date must be included.

Maximum Qualified Throughput: 18,936.05 tpmC
 Price per tpmC: \$2.46 per tpmC
 Three-year cost of ownership: \$46,539

Country-Specific Pricing

Additional Clause 7 related items may be included in the Full Disclosure Report for each country-specific priced configuration. Country-specific pricing is subject to Clause 7.1.7.

The configuration is priced for the United States of America.

Usage Pricing

For any usage pricing, the sponsor must disclose:

- Usage level at which the component was priced.
- A statement of the company policy allowing such pricing.

The component pricing based on usage is shown below:

- 1 Microsoft Windows Server 2003 Standard Edition
- 1 Microsoft SQL Server 2000 Standard Edition (based on per-processor price)
- 3-year support for hardware components (except for components for which 10 percent spares are provided)

System Pricing

System pricing should include subtotals for the following components: Server Hardware, Server Software, Client Hardware, Client Software, and Network Components used for terminal connection (see Clause 7.2.2.3). System pricing must include line item indication where non-sponsoring companies' brands are used. System pricing must also include line item indication of third-party pricing.

A detailed list of all hardware and software, including the 3-year price, is provided in the Executive Summary at the front of this report. All third-party quotations are included in Appendix E at the end of this document.

Clause 9: Audit Related Items

Auditor

The auditor's name, address, phone number, and a copy of the auditor's attestation letter indicating compliance must be included in the Full Disclosure Report.

This implementation of the TPC-C benchmark was audited by Bradley J. Askins of InfoSizing, Inc. The auditor's attestation letter is provided in this section.

Availability of the Full Disclosure Report

The Full Disclosure Report must be readily available to the public at a reasonable charge, similar to the charges for similar documents by the test sponsor. The report must be made available when results are made public. In order to use the phrase "TPC Benchmark"C," the Full Disclosure Report must have been submitted to the TPC Administrator as well as written permission obtained to distribute same.

The TPC Benchmark C Full Disclosure Report is available at www.tpc.org.





Benchmark Sponsor: Kamran Amini

Manager, xSeries Performance

IBM Systems Group 3039 Cornwallis Road

Research Triangle Park, NC 27709

July 9, 2003

I verified the TPC BenchmarkTM C performance for the following Client/Server configuration:

Platform: IBM @server xSeries 235 c/s

Operating system: Microsoft Windows Server 2003 Standard Server

Database Manager: Microsoft SQL Server 2000 Standard Edition

Transaction Manager: Microsoft COM+

The results were:

CPU's Speed	Memory	Disks	NewOrder 90% Response Time	tpmC
Server: IBM @server xSeries 235				
1 x Xeon DP (3.06GHz)	2.5 GB Main (512KB L2 Cache)	43 x 18.2 GB 2 x 73.4 GB	0.82 Seconds	18,936.05
Clients: One (1) IBM @server xSeries 225 (Specification for each)				
1 x Xeon DP (2.4 GHz)	1 GB Main (512 KB L2 Cache per processor)	1 x 18.2 GB	n/a	n/a

In my opinion, these performance results were produced in compliance with the TPC's requirements for the benchmark. The following verification items were given special attention:

- The database records were the proper size
- The database was properly scaled and populated

- The required ACID properties were met
- The transactions were correctly implemented
- Input data was generated according to the specified percentages
- The transaction cycle times included the required keying and think times
- The reported response times were correctly measured.
- All 90% response times were under the specified maximums
- At least 90% of all delivery transactions met the 80 Second completion time limit
- The reported measurement interval was 120 minutes (7200 seconds)
- The reported measurement interval was representative of steady state conditions
- Four checkpoints were taken during the reported measurement interval
- The 60 day storage requirement was correctly computed
- The system pricing was verified for major components and maintenance

Additional Audit Notes:

None.

Respectfully Yours,

François Raab, President

Bradley J. Askins, Auditor

Smlef Af-

Appendix A: Source Code

client_utils.c

```
/* client_utils.c
#include <stdio.h>
#include <time.h>
#include <windows.h>
#include <winperf.h>
#include <winsock.h>
#include "client utils.h"
#define Li2Double(x) ((double)((x).HighPart) * 4.294967296E9 +
(double)((x).LowPart))
static LARGE_INTEGER pFreq;
static double sFreq;
static int print thread id = 1;
static int user_id = 0;
static char *user code = "C";
   get thread id
     A function that returns the thread ID of the current thread
*/
static int get_thread_id()
  return(GetCurrentThreadId());
   get_prefix
           Format the output prefix for printing:
           It contains the user_id, 'C' or 'T' depending on whether it
           is a terminal or a client and optional a thread identifier
           The prefix is written in the buffer passed in by the caller.
static void get_prefix(char *buffer)
  if (print thread id) {
                       int thread_id = get_thread_id();
                      sprintf(buffer, "%s(%d-%s-%d)%s".
                      user id < 10?" ": user id < 100?" ": "",
                       user_id,
                       user_code,
                       thread id,
                       thread_id < 10 ? " " : "");
  } else {
                       sprintf(buffer, "%s(%2d-%s)",
                      user id < 10?"": "", user id, user code);
     A var-arg function that appends the current time and
     other data to the print request and sends it to stderr
     if it is not a web client, to a file if it is
void err_printf(char *format, ...)
  time_t cur_timet;
  char time str[30];
```

```
char line_prefix[50];
  va_list ap;
  va_start(ap, format);
  cur timet = time(&cur timet);
  strftime(time str, 29, "%X", localtime(&cur timet));
  get_prefix(line_prefix);
  fprintf(ERROUT, "%s %s - ", line_prefix, time_str);
  vfprintf(ERROUT, format, ap);
  fflush(ERROUT);
  va_end(ap);
  encina error message
  Report an encina error message by interpreting it and writing
* it to both the logfile (if any) and to standard error
void encina_error_message(char *msg, unsigned long n)
  char errorMsg[ENCINA MAX STATUS STRING SIZE];
  encina_StatusToString(n, ENCINA_MAX_STATUS_STRING SIZE,
  err_printf("ERROR: %s. Error code = %s (%d 0x%x) \n", msg, errorMsg, n,
n);
int get time init()
   QueryPerformanceFrequency(&pFreq);
  sFreq=Li2Double(pFreq);
  return 0;
int get local time(time type *timeP)
  double cur t;
  LARGE_INTEGER counter;
  QueryPerformanceCounter(&counter);
  cur t = Li2Double(counter) / sFreq;
  timeP->sec = (long)cur t;
   timeP->usec = ((long)cur_t - timeP->sec) * 1000000;*/
  timeP->usec = (long)((cur t - timeP->sec) * 1000000);
          return 0:
  time diff ms
     Return the difference in miliseconds between two times
int time_diff_ms(struct timeval *t2, struct timeval *t1)
  int t_diff;
  t diff = (t2->tv usec + 1000000 - t1->tv usec + 500) / 1000 +
       (t2->tv\_sec - t1->tv\_sec - 1) * 1000;
  return(t_diff);
```

```
* perfClntDataInit:
                                                                            extern char log_file_name[];
    Initialization for the shared file mapping.
                                                                             extern void logprintf( char *format, ...);
                                                                             extern void err printf( char *format, ...);
                                                                             extern void encina_error_message(char *msg, unsigned long n);
* return: pointer to the shared memory space
                                                                             extern int time_diff_ms(struct timeval *t2, struct timeval *t1);
* This routine creates a named mapped memory section that is used
* to communicate the TPCC performance data to the extensible
                                                                             typedef struct {
* counter DLL for NT perfmon.
                                                                               int num;
                                                                               int errs;
total_tran_count_t *perfCIntDataInit()
                                                                               double RTtotal[2]; // 1 for server RT and 0 for client RT
                                                                               int RTcount;
  HANDLE hMappedObject;
                                                                              tran_info_t;
  total tran count t *pClntInfo = NULL;
  TCHAR szMappedObjectName[] =
TEXT("TPCC_CLNT_COUNTER_BLOCK");
                                                                             * total_tran_count_t
    /* create named section for the performance data */
                                                                             * structure that holds the total count of transaction of each type
    hMappedObject = CreateFileMapping((HANDLE)0xFFFFFFFF,
                                                                             * as well as the reposne times.
             NULL
             PAGE READWRITE,
                                                                             typedef struct {
            sizeof(total tran count t),
                                                                               tran info t tran[MAX TRAN TYPE + 1];
            szMappedObjectName);
                                                                               int errors;
    if (hMappedObject == NULL) {
                                                                               double time:
       err_printf("perfClntDataInit: CreateFileMapping failed %x\n",
                                                                             total_tran_count_t;
             GetLastError());
       pClntInfo = NULL;
                                                                             /* enc status t
    } else {
                                                                             * structure that holds error information
       /* map the section and assign the counter block pointer
       * to this section of memory
                                                                             typedef struct {
                                                                               int status;
       pClntInfo = (total_tran_count_t *) MapViewOfFile(hMappedObject,
                                                                               int line;
                FILE_MAP_ALL_ACCESS,
                                                                               char file[268];
                                                                               unsigned long encinaError;
                                                                               char errorMsg[ENCINA_MAX_STATUS_STRING_SIZE];
                0.
                0);
                                                                              enc status t;
       if (pClntInfo == NULL) {
         err printf("perfClntDataInit: MapViewOfFile failed %x\n",
                                                                             #define FALSE 0
                                                                             #define TRUE 1
               GetLastError());
                                                                             #define DPRINT(args) if (0) err printf args
       else {
         err_printf("perfClntDataInit: MapViewOfFile success \n");
                                                                             #define CHECK ENVIRON(str,var) if (str == NULL) { fprintf(ERROUT, \
                                                                                                    "%s environment variable is not defined.\n",var); }
  return(pClntInfo);
                                                                             #define CHK_STATUS(st, val, _errMsg)
                                                                                  enc_status.status=val;
                                                                                                        FILE
                                                                                  strcpy(enc status.file,
client utils.h
                                                                                  enc_status.line= __LINE__;
                                                                                  enc status.encinaError = st;
#ifndef TPCC CLIENT UTILS H
                                                                                  if(_errMsg)strcpy(enc_status.errorMsg, _errMsg);
#define TPCC_CLIENT_UTILS_H
                                                                                  if(st!=1) return;
#include <stdio.h>
#include <time.h>
                                                                             #define UTIL IDENT(a)
#include <dce/rpc.h>
#include <dce/dce error.h>
                                                                                    ENCINA C ANSI STRING TOKEN SUPPORT
#include <encina/encina.h>
                                                                             #define UTIL_STRING(a)
                                                                                                           # a
#include <stdlib.h>
                                                                             #define UTIL CONCAT(a, b)
                                                                                                            a ## b
#include <utils/trace.h>
                                                                             #else /* ENCINA_C_ANSI_STRING_TOKEN_SUPPORT */
#include <winsock.h>
                                                                             #define UTIL_STRING(a)
#include "mon client.h"
                                                                             #define UTIL_CONCAT(a, b)
                                                                                                            UTIL IDENT(a)b
#include "../include/tpcc_type.h"
                                                                             #endif /* ENCINA_C_ANSI_STRING_TOKEN_SUPPORT */
extern FILE * errtpcc;
                                                                             /* ENCINA_CALL: Make fail-fast calls on the various services. */
extern FILE *logtpcc;
                                                                             #define ENCINA CALL(proc name, call)
extern int debug;
```

```
#ifdef __cplusplus
  unsigned long _status;
                                                                        } /*extern "C" */
  ENCINA CALL RC(proc name, call, status);
                                                                        #endif
  if (_status) exit_program(_status);
                                                                        /* end of generated dlldata file */
#define ENCINA_CALL_RC(proc_name,call,rc)
                                                                        error.h
  char errorMsg[ENCINA MAX STATUS STRING SIZE];
                                                                                 FILE:
                                                                                                     ERROR.H
  DPRINT(("ENCINA_CALL_RC: before call %s\n", proc_name));
                                                                                                               Microsoft TPC-C Kit Ver.
                                                                        4.20.000
                                                                                                               Copyright Microsoft, 1999
  DPRINT(("ENCINA_CALL_RC: after call %s\n", proc_name));
                                                                                 All Rights Reserved
  if (rc) {
                                                                                                               Version 4.10.000 audited by
    encina StatusToString(rc, ENCINA MAX STATUS STRING SIZE,
                                                                        Richard Gimarc, Performance Metrics, 3/17/99
                errorMsg);
                                                                                 PURPOSE:
                                                                                                     Header file for error exception classes.
    err printf( "%x \n", rc);
    err_printf( "%s \n", _errorMsg);
                                                                           Change history:
    err_printf( "%s \n", proc_name);
                                                                                           4.20.000 - updated rev number to match kit
                                                                                           4.21.000 - fixed bug: ~CBaseErr needed to be declared
                                                                        virtual
void err_printf(char *format, ...);
void encina error message(char *msg, unsigned long n);
                                                                        #pragma once
int get_time_init();
int get local time(time type *timeP);
                                                                        #ifndef_INC_STRING
int time_diff_ms(struct timeval *t2, struct timeval *t1);
                                                                                  #include <string.h>
                                                                        #endif
#endif /* TPCC_CLIENT_UTILS_H */
                                                                        const int m_szMsg_size = 512;
                                                                        const int m szApp size = 64;
dlldata.c
                                                                        const int m_szLoc_size = 64;
//error message structure used in ErrorText routines
 DllData file -- generated by MIDL compiler
                                                                        typedef struct _SERRORMSG
    DO NOT ALTER THIS FILE
                                                                                  int
                                                                                                     iError:
                                                                        //error id of message
 This file is regenerated by MIDL on every IDL file compile.
                                                                                  char
                                                                                           szMsg[256];
                                                                                                                                   //message to
                                                                        sent to browser
 To completely reconstruct this file, delete it and rerun MIDL
                                                                        } SERRORMSG;
 on all the IDL files in this DLL, specifying this file for the
                                                                        #define ERR FATAL LEVEL
 /dlldata command line option
                                                                        #define ERR_WARNING_LEVEL
                                                                                                                         2
**********************
                                                                        #define ERR INFORMATION LEVEL
#include <rpcproxy.h>
                                                                        #define ERR_TYPE_LOGIC
                                                                                                     //logic error in program; internal error
#ifdef __cplusplus
                                                                        #define ERR_SUCCESS
extern "C" {
                                                                                                               //success (a non-error error)
#endif
                                                                        #define ERR_BAD_ITEM_ID
                                                                                                     //expected abort record in txnRecord
EXTERN_PROXY_FILE( tpcc_com_ps )
                                                                        #define ERR_TYPE_DELIVERY_POST
                                                                                           //expected delivery post failed
                                                                        #define
                                                                                  ERR_TYPE_WEBDLL
PROXYFILE_LIST_START
                                                                                                               //tpcc web generated error
/* Start of list */
                                                                                  ERR_TYPE_SQL
                                                                        #define
REFERENCE PROXY FILE( tpcc com ps ),
                                                                                                     //sql server generated error
                                                                                 ERR TYPE DBLIB
/* End of list */
                                                                        #define
PROXYFILE_LIST_END
                                                                                                     //dblib generated error
                                                                        #define
                                                                                 ERR TYPE ODBC
                                                                                                     //odbc generated error
                                                                                  6
DLLDATA_ROUTINES( aProxyFileList, GET_DLL_CLSID )
```

```
#define ERR_TYPE_SOCKET
                                                                                    {
                              //error on communication socket client rte
                                                                                             m_idMsg
                                                                                                                  = idMsg;
                                                                                             m_szApp
only
                                                                                                                  = new char[m szApp size];
#define ERR_TYPE_DEADLOCK
                                                                                              m_szMsg
                                                                                                                  = new char[m_szMsg_size];
                              //dblib and odbc only deadlock condition
                                                                                                                  = NULL:
                                                                                              m_szLoc
#define ERR TYPE COM
                              //error from COM call
                                                                                             GetModuleFileName(GetModuleHandle(NULL),
#define ERR TYPE TUXEDO
                                                                          m_szApp, m_szApp_size);
          10
                              //tuxedo error
                                                                                              LoadString(GetModuleHandle(NULL), idMsg,
#define ERR_TYPE_OS
                                                                          m_szMsg, m_szMsg_size);
                                        //operating system error
#define ERR_TYPE_MEMORY
                              //memory allocation error
                                                                                    CBaseErr(LPCTSTR szMsg)
#define ERR_TYPE_TPCC_ODBC
          13
                              //error from tpcc odbc txn module
                                                                                             m idMsg
#define ERR TYPE TPCC DBLIB
                                                                                             m_szApp
                                                                                                                  = new char[m_szApp_size];
          14
                              //error from tpcc dblib txn module
                                                                                             m szMsg
                                                                                                                  = new char[m_szMsg_size];
#define ERR_TYPE_DELISRV
                                                                                              m_szLoc
                                                                                                                  = NULL;
                    //delivery server error
#define ERR TYPE TXNLOG
                                                                                             GetModuleFileName(GetModuleHandle(NULL),
                              //txn log error
          16
                                                                          m_szApp, m_szApp_size);
#define ERR TYPE BCCONN
                                                                                              strcpy(m_szMsg, szMsg);
          17
                              //Benchcraft connection class
#define ERR TYPE TPCC CONN
          18
                              //Benchcraft connection class
                                                                                    void SetError(char *szMsg, LPCTSTR szLocation)
#define ERR_TYPE_ENCINA
                              //Encina error
                                                                                             if (szMsg != NULL)
          19
#define ERR_TYPE_COMPONENT
                                                                                                        strcpy(m_szMsg, szMsg);
          20
                              //error from COM component
                                                                                             else
#define ERR_TYPE_RTE
                                                                                                       m_szMsg[0] = 0;
          21
                              //Benchcraft rte
#define ERR_TYPE_AUTOMATION
                                                                                             if (szLocation != NULL)
          22
                              //Benchcraft automation errors
                                                                                                        delete [] m szLoc;
                                                                                                        m_szLoc = new char[strlen(szLocation)+1];
class CBaseErr
                                                                                                        strcpy(m szLoc, szLocation);
public:
                                                                                             else
          char
                    *m_szApp;
                                                                                                        delete [] m_szLoc;
          char
                    *m_szMsg;
                    *m szLoc;// code location where the error occurred
                                                                                                        m szLoc = NULL;
          char
          int
                              m_idMsg;
                                                                                    }
          CBaseErr(void)
                                                                                    virtual void Draw(HWND hwnd, LPCTSTR szStr = NULL)
                    m_idMsg
                    m szMsg
                                        = new char[m szMsg size];
                                                                                             int
                    m szApp
                                        = new char[m szApp size];
                                                                                             char
                                                                                                        szTmp[512];
                                        = NULL;
                    m_szLoc
                                                                                             if (szStr)
                                                                                                        j = wsprintf(szTmp, "%s\n",szStr);
                    m_szMsg[0]
                                        = 0.
                    m szApp[0]
                                        = 0;
                                                                                             if (m szLoc)
                                                                                                         += wsprintf(szTmp+j,
                    GetModuleFileName(GetModuleHandle(NULL),
                                                                          "Location=%s\n",m szLoc);
m_szApp, m_szApp_size);
                                                                                                       j += wsprintf(szTmp+j, "%s\n", m_szMsg);
          }
          virtual ~CBaseErr(void)
                                                                                             ::MessageBox(hwnd, szTmp, m_szApp, MB_OK);
                    if (m_szMsg)
                                                                                    char *GetApp(void) { return m szApp; }
                              delete [] m_szMsg;
                    if (m_szApp)
                                                                                    char *GetMsg(void) { return m_szMsg; }
                              delete [] m_szApp;
                                                                                    char *GetLocation(void) { return m_szLoc; }
                    if (m_szLoc)
                                                                                    virtual int ErrorType() = 0;
                                                                                                                  // a value which distinguishes the
                              delete [] m_szLoc;
          };
                                                                          kind of error that occurred
                                                                                    virtual int ErrorNum() = 0;
                                                                                                                            // an error value
          CBaseErr(int idMsg)
                                                                          specific to the error type
```

```
virtual char *ErrorText() = 0;
                                        // a string (i.e., human readable)
                                                                          public:
representation of the error
                                                                                    CMemoryErr(void);
                                                                                    int ErrorType() { return ERR_TYPE_MEMORY;}
                                                                                    int ErrorNum() { return 0;}
class CSocketErr: public CBaseErr
                                                                                    char *ErrorText() { return "Insufficient Memory to continue.";}
                                                                          };
public:
          enum Action
                                                                          install.c
                    eNone,
                                                                                    FILE:
                                                                                                         INSTALL.C
                    eSend
                                                                                                                   Microsoft TPC-C Kit Ver.
                    eSocket,
                                                                          4.20.000
                    eConnect
                                                                                                                   Copyright Microsoft, 1999
          };
                                                                                    All Rights Reserved
          CSocketErr(Action eAction, LPCTSTR szLocation);
                                                                                                                   not audited
          CSocketErr(int iError) { m_errId = iError; };
          int
                              m_errId;
                                                                                    PURPOSE:
                                                                                                         Automated installation application for TPC-C
          Action
                    m eAction;
                                                                           Web Kit
                                                                                    Contact: Charles Levine (clevine@microsoft.com)
          int ErrorType() { return ERR TYPE SOCKET;};
          int ErrorNum() { return m_errId;};
                                                                             Change history:
          char *ErrorText(void);
                                                                                               4.20.000 - added COM installation steps
};
class CSystemErr: public CBaseErr
                                                                          #include <windows.h>
                                                                          #include <direct.h>
public:
                                                                          #include <io.h>
          enum Action
                                                                          #include <stdlib.h>
                                                                          #include <stdio.h>
                    eNone,
                    eTransactNamedPipe,
                                                                          #include <commetrl.h>
                                                                          #include "..\..\common\src\ReadRegistry.h"
                    eWaitNamedPipe,
                    eSetNamedPipeHandleState,
                                                                          #include "resource.h"
                    eCreateFile,
                    eCreateProcess,
                                                                                                                   WM_USER+100
                                                                                    WM_INITTEXT
                    eCallNamedPipe,
                                                                          #define
                    eCreateEvent,
                    eCreateThread,
                                                                          HICON
                                                                                              hIcon;
                                                                          HINSTANCE
                                                                                              hInst:
                    eVirtualAlloc,
                    eReadFile,
                    eWriteFile,
                                                                          DWORD
                                                                                                         versionExeMS;
                                                                          DWORD
                                                                                                         versionExeLS;
                    eMapViewOfFile,
                                                                          DWORD
                    eCreateFileMapping,
                                                                                                         versionExeMM;
                                                                          DWORD
                                                                                                         versionDllMS;
                    eInitializeSecurityDescriptor,
                    eSetSecurityDescriptorDacl,
                                                                          DWORD
                                                                                                         versionDllLS;
                    eCreateNamedPipe,
                    eConnectNamedPipe,
                                                                          // TPC-C registry settings
                                                                          TPCCREGISTRYDATA
                    eWaitForSingleObject,
                                                                                                         Reg;
                    eRegOpenKeyEx,
                                                                                                         iPoolThreadLimit;
                    eRegQueryValueEx,
                                                                          static
                                                                                    int
                                                                          static
                                                                                    int
                                                                                                         iThreadTimeout;
          };
                                                                          static
                                                                                    int
                                                                                                         iListenBackLog;
                                                                                                         iAcceptExOutstanding;
                                                                          static
                                                                                    int
          CSystemErr(Action eAction, LPCTSTR szLocation);
                                                                                                         iMaxPhysicalMemory;
          void Draw(HWND hwnd, LPCTSTR szStr = NULL);
                                                                          static
                                                                                    int
                                                                          //max physical memory in MB
                                                                          static
                                                                                    char
                                                                                              szLastFileName[64];
                                                                                                                             // last file we worked
          int
                              m errId;
                                                                          on (for error reporting)
          Action
                    m_eAction;
          int ErrorType() { return ERR_TYPE_OS;}
                                                                          BOOL
                                                                                    CALLBACK
                                                                                                         LicenseDlgProc(HWND hwnd, UINT uMsg,
          int ErrorNum() { return m_errId;}
                                                                          WPARAM wParam, LPARAM lParam);
          char *ErrorText() { return m_szMsg; }
                                                                                                         UpdatedDlgProc(HWND hwnd, UINT uMsg,
                                                                          BOOL
                                                                                    CALLBACK
};
                                                                          WPARAM wParam, LPARAM lParam);
                                                                          BOOL
                                                                                    CALLBACK
                                                                                                         MainDlgProc(HWND hwnd, UINT uMsg,
class CMemoryErr: public CBaseErr
                                                                          WPARAM wParam, LPARAM lParam);
```

```
BOOL
                            CopyDlgProc(HWND hwnd, UINT uMsg,
         CALLBACK
                                                                                                  SendMessage( GetDlgItem(hwnd,
                                                                     IDR LICENSE1), WM SETFONT, (WPARAM)hFont, MAKELPARAM(0, 0)
WPARAM wParam, LPARAM lParam);
                            ProcessOK(HWND hwnd, char *szDllPath);
static
         void
                            ReadRegistrySettings(void);
                                                                                                  PostMessage(hwnd, WM_INITTEXT,
static
         void
                            WriteRegistrySettings(char *szDllPath);
                                                                     (WPARAM)0, (LPARAM)0);
static
         void
         BOOL
                            RegisterDLL(char *szFileName);
                                                                                                  return TRUE;
static
                                     CopyFiles(HWND hDlg, char
static
         int
                                                                                        case WM INITTEXT:
                                                                                                  hResInfo = FindResource(hInst,
*szDllPath);
                                                                     MAKEINTRESOURCE(IDR LICENSE1), "LICENSE");
static
         BOOL
                            GetInstallPath(char *szDllPath);
                            GetVersionInfo(char *szDLLPath, char
                                                                                                  dwSize = SizeofResource(hInst, hResInfo);
         void
static
                                                                                                  hRes = LoadResource(hInst, hResInfo );
*szExePath);
         BOOL
                            CheckWWWebService(void);
                                                                                                  pSrc = (BYTE *)LockResource(hRes);
static
static
         BOOL
                            StartWWWebService(void);
                                                                                                  pDst = (unsigned char *)malloc(dwSize+1);
                            StopWWWebService(void);
         BOOL
                                                                                                  if (pDst)
static
                            UpdateDialog(HWND hDlg);
static
         void
                                                                                                           memcpy(pDst, pSrc, dwSize);
BOOL install com(char *szDllPath);
                                                                                                           pDst[dwSize] = 0;
                                                                                                           SetDlgItemText(hwnd,
#include "..\..\common\src\ReadRegistry.cpp"
                                                                     IDC_LICENSE, (const char *)pDst);
                                                                                                           free(pDst);
int WINAPI WinMain( HINSTANCE hInstance, HINSTANCE hPrevInstance,
LPSTR lpCmdLine, int nCmdShow)
                                                                                                  else
                                                                                                           SetDlgItemText(hwnd,
         int iRc;
                                                                     IDC LICENSE, (const char *)pSrc);
                                                                                                  return TRUE;
         hInst = hInstance;
                                                                                        case WM DESTROY:
                                                                                                  DeleteObject(hFont);
         InitCommonControls();
                                                                                                  return TRUE;
                                                                                        case WM COMMAND:
                                                                                                  if(wParam == IDOK)
         hIcon = LoadIcon(hInstance,
MAKEINTRESOURCE(IDI ICON1));
                                                                                                           EndDialog(hwnd, TRUE);
                                                                                                  if ( wParam == IDCANCEL )
         iRc = DialogBox(hInstance.
                                                                                                           EndDialog(hwnd, FALSE);
MAKEINTRESOURCE(IDD DIALOG4), GetDesktopWindow(),
                                                                                        default:
LicenseDlgProc);
                                                                                                  break:
         if (iRc)
                                                                               return FALSE;
                   iRc = DialogBox(hInstance,
MAKEINTRESOURCE(IDD_DIALOG1), GetDesktopWindow(),
MainDlgProc);
                                                                     BOOL CALLBACK UpdatedDlgProc(HWND hwnd, UINT uMsg, WPARAM
                   if (iRc)
                                                                     wParam, LPARAM lParam)
                            DialogBoxParam(hInstance,
                                                                               switch(uMsg)
MAKEINTRESOURCE(IDD DIALOG2), GetDesktopWindow(),
UpdatedDlgProc, (LPARAM)iRc);
                                                                                        case WM INITDIALOG:
                                                                                                  switch(lParam)
                                                                                                           case 1:
         DestroyIcon(hIcon);
                                                                                                           case 2:
                                                                                                                     SetDlgItemText(hwnd,
                                                                     IDC_RESULTS, "TPC-C Web Client Installed");
                                                                                                                     break;
BOOL CALLBACK LicenseDlgProc(HWND hwnd, UINT uMsg, WPARAM
wParam, LPARAM lParam)
                                                                                                  return TRUE;
                                                                                        case WM COMMAND:
         HGLOBAL
                                               hRes;
                                                                                                  if ( wParam == IDOK )
         HRSRC
                                      hResInfo;
                                                                                                           EndDialog(hwnd, TRUE);
         BYTE
                                      *pSrc, *pDst;
                                                                                                  break:
         DWORD
                                      dwSize;
                                                                                        default:
         static HFONT
                            hFont;
                                                                                                  break;
         switch(uMsg)
                                                                               return FALSE;
                  case WM INITDIALOG:
                                                                     BOOL CALLBACK MainDlgProc(HWND hwnd, UINT uMsg, WPARAM
                            0, 0, 0, 0, 0, "Arial");
                                                                     wParam, LPARAM lParam)
                                                                               PAINTSTRUCT
                                                                                                           ps;
```

```
MEMORYSTATUS memoryStatus;
                                                                                                    SetDlgItemInt(hwnd,
                                                                       ED MAXCONNECTION, Reg.dwMaxConnections, FALSE);
         OSVERSIONINFO VI;
                                      szTmp[256];
                                                                                                    SetDlgItemInt(hwnd.
         char
                             szDllPath[256];
                                                                       ED_MAXDELIVERIES, Reg.dwMaxPendingDeliveries, FALSE);
         static char
                             szExePath[256];
                                                                                                    SetDlgItemInt(hwnd,
         static char
                                                                       ED IIS MAX THEAD POOL LIMIT, iPoolThreadLimit, FALSE);
                                                                                                    SetDlgItemInt(hwnd,
         switch(uMsg)
                                                                       ED_IIS_THREAD_TIMEOUT, iThreadTimeout, FALSE);
                   case WM INITDIALOG:
                                                                                                    SetDlgItemInt(hwnd,
                                                                       ED IIS LISTEN BACKLOG, iListenBackLog, FALSE);
                             GlobalMemoryStatus(&memoryStatus);
                                                                                                    SetDlgItemInt(hwnd,
                             iMaxPhysicalMemory=
(memoryStatus.dwTotalPhys/1048576);
                                                                       ED WEB_SERVICE_BACKLOG_QUEUE_SIZE, iAcceptExOutstanding,
                                                                       FALSE);
                             if ( GetInstallPath(szDllPath) )
                                                                                                    CheckDlgButton(hwnd, IDC DBLIB, 0);
                                                                                                    CheckDlgButton(hwnd, IDC_ODBC, 0);
                                      MessageBox(hwnd, "Error internet
                                                                                                    if (Reg.eDB Protocol == DBLIB)
service inetsry is not installed.", NULL, MB ICONSTOP | MB OK);
                                                                                                             CheckDlgButton(hwnd,
                                      EndDialog(hwnd, FALSE);
                                      return TRUE:
                                                                       IDC_DBLIB, 1);
                                                                                                    else
                                                                                                             CheckDlgButton(hwnd,
                             // set default values
                                                                       IDC ODBC, 1);
                             ZeroMemory( &Reg, sizeof(Reg) );
                             Reg.dwNumberOfDeliveryThreads = 4;
                                                                                                    // check OS version level for COM. Must be
                             Reg.dwMaxConnections = 100;
                                                                       at least Windows 2000
                             Reg.dwMaxPendingDeliveries
                                                         = 100:
                                                                                                    VI.dwOSVersionInfoSize = sizeof(VI);
                             Reg.eDB Protocol = DBLIB;
                                                                                                    GetVersionEx( &VI );
                             Reg.eTxnMon = None;
                                                                                                    if (VI.dwMajorVersion < 5)
                             strcpy(Reg.szDbServer,
                                                                                                    {
"");
                                                                                                             HWND hDlg = GetDlgItem(
                             strcpy(Reg.szDbName,
                                                                       hwnd, IDC TM MTS );
"tpcc");
                                                                                                             EnableWindow( hDlg, 0 );
                                                                                                                                          //
                             strcpy(Reg.szDbUser,
                                                          "sa");
                                                                       disable COM option
                             strcpy(Reg.szDbPassword,
                                                          "");
                                                                                                             if(Reg.eTxnMon == COM)
                                                                                                                       Reg.eTxnMon = None;
                             iPoolThreadLimit = iMaxPhysicalMemory *
2;
                             iThreadTimeout = 86400;
                                                                                                    CheckDlgButton(hwnd, IDC TM NONE,
                             iListenBackLog = 15;
                                                                       0);
                             iAcceptExOutstanding = 40;
                                                                                                    CheckDlgButton(hwnd, IDC_TM_TUXEDO,
                                                                       0);
                             ReadTPCCRegistrySettings( &Reg );
                                                                                                    CheckDlgButton(hwnd, IDC TM MTS, 0);
                             ReadRegistrySettings();
                                                                                                    CheckDlgButton(hwnd, IDC TM ENCINA,
                                                                       0);
                             GetModuleFileName(hInst, szExePath,
                                                                                                    switch (Reg.eTxnMon)
sizeof(szExePath));
                             GetVersionInfo(szDllPath, szExePath);
                                                                                                    case None:
                                                                                                             CheckDlgButton(hwnd,
                             wsprintf(szTmp, "Version %d.%2.2d.%3.3d",
                                                                       IDC_TM_NONE, 1);
versionExeMS, versionExeMM,
                            versionExeLS);
                                                                                                             break;
                             SetDlgItemText(hwnd, IDC_VERSION,
                                                                                                    case TUXEDO:
                                                                                                             CheckDlgButton(hwnd,
szTmp);
                                                                       IDC_TM_TUXEDO, 1);
                             SetDlgItemText(hwnd, IDC PATH,
                                                                                                             break;
szDllPath);
                                                                                                    case ENCINA:
                                                                                                              CheckDlgButton(hwnd,
                             SetDlgItemText(hwnd, ED DB SERVER,
                                                                       IDC TM ENCINA, 1);
Reg.szDbServer);
                                                                                                             break:
                             SetDlgItemText(hwnd, ED DB USER ID,
                                                                                                    case COM
Reg.szDbUser);
                                                                                                              CheckDlgButton(hwnd,
                             SetDlgItemText(hwnd,
                                                                       IDC_TM_MTS,
                                                                                          1);
ED DB PASSWORD, Reg.szDbPassword);
                                                                                                             break;
                             SetDlgItemText(hwnd, ED_DB_NAME,
Reg.szDbName);
                                                                                                    return TRUE;
                             SetDlgItemInt(hwnd, ED THREADS,
                                                                                          case WM PAINT:
Reg.dwNumberOfDeliveryThreads, FALSE);
                                                                                                    if (IsIconic(hwnd))
```

```
BeginPaint(hwnd, &ps);
                                                                                            rc = 1;
                                       DrawIcon(ps.hdc, 0, 0, hIcon);
                                       EndPaint(hwnd, &ps);
                                                                                  else if ( IsDlgButtonChecked(hwnd, IDC ODBC) )
                                       return TRUE;
                                                                                            Reg.eDB_Protocol = ODBC;
                                                                                            rc = 2;
                             break;
                   case WM COMMAND:
                             if ( HIWORD(wParam) == BN CLICKED )
                                                                                  if (IsDlgButtonChecked(hwnd, IDC_TM_NONE))
                                       switch( LOWORD(wParam) )
                                                                                            Reg.eTxnMon = None;
                                                                                  else if ( IsDlgButtonChecked(hwnd, IDC_TM_TUXEDO) )
                                                 case IDC DBLIB:
                                                                                            Reg.eTxnMon = TUXEDO;
                                                           return
                                                                                  else if (IsDlgButtonChecked(hwnd, IDC TM MTS))
TRUE;
                                                                                            Reg.eTxnMon = COM;
                                                                                  else if ( IsDlgButtonChecked(hwnd, IDC_TM ENCINA) )
                                                 case IDC ODBC:
                                                           return
                                                                                            Reg.eTxnMon = ENCINA;
TRUE;
                                                                                  iPoolThreadLimit = GetDlgItemInt(hwnd,
                                                 case IDOK:
                                                                        ED_IIS_MAX_THEAD_POOL_LIMIT, &d, FALSE);
ProcessOK(hwnd, szDllPath);
                                                                                  iThreadTimeout = GetDlgItemInt(hwnd,
                                                                        ED_IIS_THREAD_TIMEOUT, &d, FALSE);
                                                           return
                                                                        iListenBackLog = GetDlgItemInt(hwnd, ED_IIS_LISTEN_BACKLOG, &d, FALSE);
TRUE;
                                                 case IDCANCEL:
                                                                                  iAcceptExOutstanding = GetDlgItemInt(hwnd,
EndDialog(hwnd, FALSE);
                                                                        ED_WEB_SERVICE_BACKLOG_QUEUE_SIZE, &d, FALSE);
                                                           return
TRUE;
                                                                                  ShowWindow(hwnd, SW_HIDE);
                                                 default:
                                                           return
                                                                                  hDlg = CreateDialog(hInst,
                                                                        MAKEINTRESOURCE(IDD_DIALOG3), hwnd, CopyDlgProc);
FALSE;
                                                                                  ShowWindow(hDlg, SW_SHOWNA);
                                                                                  UpdateDialog(hDlg);
                             break:
                    default:
                                                                                  // write binaries to inetpub\wwwroot
                             break:
                                                                                  rc = CopyFiles(hDlg, szDllPath);
          return FALSE;
                                                                                  if (!rc)
}
                                                                                            ShowWindow(hwnd, SW_SHOWNA);
static void ProcessOK(HWND hwnd, char *szDllPath)
                                                                                            DestroyWindow(hDlg);
                                                                                            strcpy( szErrTxt, "Error(s) occured when creating " );
                                                                                            strcat( szErrTxt, szLastFileName );
          int
                             d;
          HWND
                   hDlg;
                                                                                            MessageBox(hwnd, szErrTxt, NULL, MB ICONSTOP |
                                                                        MB_OK);
          int
                             rc:
                                                                                            EndDialog(hwnd, 0);
                    szFullName[256];
          char
                                                                                            return;
                   szErrTxt[128];
          char
          // read settings from dialog
                                                                                  // update registry
          Reg.dwNumberOfDeliveryThreads = GetDlgItemInt(hwnd,
                                                                                  SetDlgItemText(hDlg, IDC STATUS, "Updating Registry.");
ED_THREADS, &d, FALSE);
                                                                                  SendDlgItemMessage(hDlg, IDC_PROGRESS1, PBM_STEPIT, 0,
          Reg.dwMaxConnections = GetDlgItemInt(hwnd,
                                                                        0);
ED_MAXCONNECTION, &d, FALSE);
                                                                                  UpdateDialog(hDlg);
          Reg.dwMaxPendingDeliveries = GetDlgItemInt(hwnd,
                                                                                  WriteRegistrySettings(szDllPath);
ED MAXDELIVERIES, &d, FALSE);
                                                                                  // register com proxy stub
          GetDlgItemText(hwnd, ED_DB_SERVER, Reg.szDbServer,
                                                                                  strcpy(szFullName, szDllPath);
sizeof(Reg.szDbServer));
                                                                                  strcat(szFullName, "tpcc_com_ps.dll");
          GetDlgItemText(hwnd, ED_DB_USER_ID, Reg.szDbUser,
                                                                                  if (!RegisterDLL(szFullName))
sizeof(Reg.szDbUser));
          GetDlgItemText(hwnd, ED_DB_PASSWORD, Reg.szDbPassword,
                                                                                            ShowWindow(hwnd, SW SHOWNA);
sizeof(Reg.szDbPassword));
                                                                                            DestroyWindow(hDlg);
                                                                                            strcpy( szErrTxt, "Error occured when registering " );
          GetDlgItemText(hwnd, ED_DB_NAME, Reg.szDbName,
sizeof(Reg.szDbName));
                                                                                            strcat( szErrTxt, szFullName );
                                                                                            MessageBox(hwnd, szErrTxt, NULL, MB_ICONSTOP |
          if ( IsDlgButtonChecked(hwnd, IDC DBLIB) )
                                                                        MB OK);
                                                                                            EndDialog(hwnd, 0);
                    Reg.eDB Protocol = DBLIB;
                                                                                            return;
```

```
}
                                                                                 {
                                                                                          size = sizeof(iAcceptExOutstanding);
         // if using COM
                                                                                          if (RegQueryValueEx(hKey, "AcceptExOutstanding", 0,
                                                                       &type, (char *)&iAcceptExOutstanding, &size) == ERROR_SUCCESS)
         if (Reg.eTxnMon == COM)
                                                                                                    if (!iAcceptExOutstanding)
                   SetDlgItemText(hDlg, IDC STATUS, "Configuring
                                                                                                              iAcceptExOutstanding = 40;
COM.");
                   SendDlgItemMessage(hDlg, IDC PROGRESS1,
                                                                                           RegCloseKey(hKey);
PBM STEPIT, 0, 0);
                   UpdateDialog(hDlg);
                   if (install_com(szDllPath))
                                                                       static void WriteRegistrySettings(char *szDllPath)
                             ShowWindow(hwnd, SW_SHOWNA);
                                                                                 HKEY
                                                                                          hKey;
                             DestroyWindow(hDlg);
                                                                                 DWORD dwDisposition;
                             strcpy( szErrTxt, "Error occured when
                                                                                 char
                                                                                           szTmp[256];
configuring COM settings.");
                                                                                 char
                                                                                           *ptr;
                             MessageBox(hwnd, szErrTxt, NULL,
                                                                                                    iRc;
MB ICONSTOP | MB OK);
                                                                                 if (RegCreateKeyEx(HKEY LOCAL MACHINE,
                             EndDialog(hwnd, 0);
                                                                        "SOFTWARE\\Microsoft\\TPCC", 0, NULL,
                             return;
                                                                       REG OPTION NON VOLATILE, KEY ALL ACCESS, NULL, &hKey,
                                                                       &dwDisposition) == ERROR SUCCESS)
         Sleep(100);
                                                                                           strcpy(szTmp, szDllPath);
                                                                                          ptr = strstr(szTmp, "tpcc");
         ShowWindow(hwnd, SW SHOWNA);
                                                                                           if (ptr)
         DestroyWindow(hDlg);
                                                                                                    *ptr = 0;
         EndDialog(hwnd, rc);
                                                                                           RegSetValueEx(hKey, "Path", 0, REG_SZ, szTmp,
         return;
                                                                       strlen(szTmp)+1);
                                                                                          RegSetValueEx(hKey, "NumberOfDeliveryThreads", 0,
                                                                       REG DWORD, (char *)&Reg.dwNumberOfDeliveryThreads,
static void ReadRegistrySettings(void)
                                                                       sizeof(Reg.dwNumberOfDeliveryThreads));
                                                                                           RegSetValueEx(hKey, "MaxConnections", 0,
         HKEY
                   hKey;
         DWORD size;
                                                                       REG_DWORD, (char *)&Reg.dwMaxConnections,
         DWORD type;
                                                                       sizeof(Reg.dwMaxConnections));
                                                                                          RegSetValueEx(hKey, "MaxPendingDeliveries", 0,
                                                                       REG DWORD, (char *)&Reg.dwMaxPendingDeliveries,
         if ( RegOpenKeyEx(HKEY_LOCAL_MACHINE,
"SYSTEM\\CurrentControlSet\\Services\\Inetinfo\\Parameters", 0,
                                                                       sizeof(Reg.dwMaxPendingDeliveries));
KEY_READ, &hKey) == ERROR_SUCCESS )
                                                                                           RegSetValueEx(hKey, "DB Protocol", 0, REG SZ,
                                                                       szDBNames[Reg.eDB_Protocol], strlen(szDBNames[Reg.eDB_Protocol])+1);
                   size = sizeof(iPoolThreadLimit);
                   if (RegQueryValueEx(hKey, "PoolThreadLimit", 0,
                                                                                           RegSetValueEx(hKey, "TxnMonitor", 0, REG SZ,
&type, (char *)&iPoolThreadLimit, &size) == ERROR_SUCCESS)
                                                                       szTxnMonNames[Reg.eTxnMon], strlen(szTxnMonNames[Reg.eTxnMon])+1);
                             if (!iPoolThreadLimit)
                                      iPoolThreadLimit =
                                                                                           RegSetValueEx(hKey, "DbServer", 0, REG SZ,
                                                                       Reg.szDbServer, strlen(Reg.szDbServer)+1);
iMaxPhysicalMemory * 2;
                                                                                           RegSetValueEx(hKey, "DbName", 0, REG SZ,
                                                                       Reg.szDbName, strlen(Reg.szDbName)+1);
                   size = sizeof(iThreadTimeout);
                   if ( RegQueryValueEx(hKey, "ThreadTimeout", 0,
                                                                                           RegSetValueEx(hKey, "DbUser", 0, REG SZ,
&type, (char *)&iThreadTimeout, &size) == ERROR_SUCCESS)
                                                                       Reg.szDbUser, strlen(Reg.szDbUser)+1);
                             if (!iThreadTimeout)
                                                                                           RegSetValueEx(hKey, "DbPassword", 0, REG SZ,
                                      iThreadTimeout = 86400;
                                                                       Reg.szDbPassword, strlen(Reg.szDbPassword)+1);
                   size = sizeof(iListenBackLog);
                                                                                           strcpy(szTmp, "YES");
                                                                                           RegSetValueEx(hKey, "COM_SinglePool", 0, REG SZ,
                   if ( RegQueryValueEx(hKey, "ListenBackLog", 0,
&type, (char *)&iListenBackLog, &size) == ERROR SUCCESS)
                                                                       szTmp, strlen(szTmp)+1);
                             if (!iListenBackLog)
                                      iListenBackLog = 15;
                                                                                           RegFlushKev(hKev);
                                                                                           RegCloseKey(hKey);
                   RegCloseKey(hKey);
                                                                                 if ((iRc=RegCreateKeyEx(HKEY_LOCAL_MACHINE,
         if (RegOpenKeyEx(HKEY LOCAL MACHINE,
                                                                        "SYSTEM\\CurrentControlSet\\Services\\Inetinfo\\Parameters", 0, NULL,
                                                                       REG_OPTION_NON_VOLATILE, KEY_ALL_ACCESS, NULL, &hKey,
"SYSTEM\\CurrentControlSet\\Services\\W3SVC\\Parameters", 0,
KEY_READ, &hKey) == ERROR_SUCCESS )
                                                                       &dwDisposition)) == \overline{ERROR} SUCCESS)
```

```
DWORD
                   RegSetValueEx(hKey, "PoolThreadLimit", 0,
                                                                                   char
                                                                                                                 szFullName[256];
REG DWORD, (char *)&iPoolThreadLimit, sizeof(iPoolThreadLimit));
                    RegSetValueEx(hKey, "ThreadTimeout", 0,
                                                                                   hResInfo = FindResource(hInst,
REG DWORD, (char *)&iThreadTimeout, sizeof(iThreadTimeout));
                                                                         MAKEINTRESOURCE(iResourceId), szResourceName);
                    RegSetValueEx(hKey, "ListenBackLog", 0,
REG DWORD, (char *)&iListenBackLog, sizeof(iListenBackLog));
                                                                                   strcpy(szFullName, szDllPath);
                                                                                   strcat(szFullName, szFileName);
                    RegFlushKey(hKey);
                    RegCloseKey(hKey);
                                                                                   dwSize = SizeofResource(hInst, hResInfo);
                                                                                   hDLL = LoadResource(hInst, hResInfo);
                                                                                   pSrc = (BYTE *)LockResource(hDLL);
         if ((iRc=RegCreateKeyEx(HKEY LOCAL MACHINE,
                                                                                   remove(szFullName);
"SYSTEM\\CurrentControlSet\\Services\\W3SVC\\Parameters", 0, NULL,
REG OPTION NON VOLATILE, KEY ALL ACCESS, NULL, &hKey,
                                                                                   if (!(hFile = CreateFile(szFullName, GENERIC WRITE, 0, NULL,
                                                                         CREATE ALWAYS, FILE ATTRIBUTE NORMAL, NULL)))
&dwDisposition)) == ERROR SUCCESS)
                                                                                            return FALSE;
                    RegSetValueEx(hKey, "AcceptExOutstanding", 0,
REG DWORD, (char *)&iAcceptExOutstanding,
                                                                                   if (!WriteFile(hFile, pSrc, dwSize, &d, NULL))
sizeof(iAcceptExOutstanding));
                                                                                            return FALSE;
                    RegFlushKey(hKey);
                                                                                   CloseHandle(hFile);
                    RegCloseKey(hKey);
                                                                                   UnlockResource(hDLL);
                                                                                   FreeResource(hDLL);
                                                                                   return TRUE;
         return:
BOOL CALLBACK CopyDlgProc(HWND hwnd, UINT uMsg, WPARAM
                                                                         static int CopyFiles(HWND hDlg, char *szDllPath)
wParam, LPARAM lParam)
                                                                                   BOOL
                                                                                                       bSvcRunning;
          if ( uMsg == WM_INITDIALOG )
                                                                                   bSvcRunning = CheckWWWebService();
                    SendDlgItemMessage(hwnd, IDC PROGRESS1,
                                                                                   if (bSvcRunning)
PBM_SETRANGE, 0, MAKELPARAM(0, 16));
                    SendDlgItemMessage(hwnd, IDC PROGRESS1,
                                                                                             SetDlgItemText(hDlg, IDC STATUS, "Stopping Web
PBM_SETSTEP, (WPARAM)1, 0);
                                                                         Service.");
                   return TRUE;
                                                                                            SendDlgItemMessage(hDlg, IDC PROGRESS1,
                                                                         PBM_STEPIT, 0, 0);
         return FALSE;
                                                                                             UpdateDialog(hDlg);
                                                                                            StopWWWebService();
BOOL RegisterDLL(char *szFileName)
                                                                                             SendDlgItemMessage(hDlg, IDC PROGRESS1,
                                                                         PBM_STEPIT, 0, 0);
          HINSTANCE
                                                                                             UpdateDialog(hDlg);
                             hLib;
          FARPROC
                                        lpDllEntryPoint;
                                                                                   SetDlgItemText(hDlg, IDC STATUS, "Copying Files...");
          hLib = LoadLibrary(szFileName);
          if(hLib == NULL)
                                                                                   SendDlgItemMessage(hDlg, IDC_PROGRESS1, PBM_STEPIT, 0,
                   return FALSE;
                                                                         0);
          // Find the entry point.
                                                                                   UpdateDialog(hDlg);
          lpDllEntryPoint = GetProcAddress(hLib, "DllRegisterServer");
          if (lpDllEntryPoint != NULL)
                                                                                   // install TPCC.DLL
                                                                                   strcpy( szLastFileName, "tpcc.dll" ); if (!FileFromResource( "TPCCDLL", IDR_TPCCDLL, szDllPath,
          {
                   return ((*lpDllEntryPoint)() == S OK);
                                                                         szLastFileName ))
         else
                                                                                   SendDlgItemMessage(hDlg, IDC PROGRESS1, PBM STEPIT, 0,
                   return FALSE;
                                       //unable to locate entry point
                                                                         0);
                                                                                   UpdateDialog(hDlg);
BOOL FileFromResource( char *szResourceName, int iResourceId, char
*szDllPath, char *szFileName)
                                                                                   // install tpcc dblib.dll
                                                                                   strcpy( szLastFileName, "tpcc_dblib.dll" );
if (!FileFromResource( "DBLIB_DLL", IDR_DBLIB_DLL,
          HGLOBAL
                                                 hDLL;
          HRSRC
                                       hResInfo;
                                                                         szDllPath, szLastFileName ))
          HANDLE
                                       hFile;
                                                                                   SendDlgItemMessage(hDlg, IDC_PROGRESS1, PBM_STEPIT, 0,
          DWORD
                                        dwSize;
          BYTE
                                        *pSrc;
                                                                         0);
```

```
SetDlgItemText(hDlg, IDC_STATUS, "Starting Web
          UpdateDialog(hDlg);
                                                                         Service.");
          // install tpcc_odbc.dll
                                                                                             SendDlgItemMessage(hDlg, IDC PROGRESS1,
          strcpy( szLastFileName, "tpcc_odbc.dll" );
                                                                         PBM STEPIT, 0, 0);
          if (!FileFromResource( "ODBC_DLL", IDR_ODBC_DLL,
                                                                                             UpdateDialog(hDlg);
szDllPath, szLastFileName ))
                                                                                             StartWWWebService();
                    return 0;
                                                                                   }
          SendDlgItemMessage(hDlg, IDC PROGRESS1, PBM STEPIT, 0,
0);
                                                                                   SendDlgItemMessage(hDlg, IDC PROGRESS1, PBM STEPIT, 0,
                                                                         0);
          UpdateDialog(hDlg);
                                                                                   UpdateDialog(hDlg);
          // install tuxapp.exe
          strcpy( szLastFileName, "tuxapp.exe" );
                                                                                   return 1;
          if (!FileFromResource( "TUXEDO_APP", IDR_TUXEDO_APP,
szDllPath, szLastFileName ))
                    return 0;
                                                                         static BOOL GetInstallPath(char *szDllPath)
          SendDlgItemMessage(hDlg, IDC PROGRESS1, PBM STEPIT, 0,
                                                                           HKEY hKey;
0);
          UpdateDialog(hDlg);
                                                                                   BYTE
                                                                                             szData[256];
                                                                                   DWORD sv;
                                                                                   BOOL
          // install tpcc_tuxedo.dll
                                                                                             bRc;
          strcpy( szLastFileName, "tpcc_tuxedo.dll" );
                                                                                   int
                                                                                                       len;
          if (!FileFromResource( "TUXEDO_DLL", IDR_TUXEDO_DLL,
                                                                                   char
                                                                                             *ptr;
szDllPath, szLastFileName ))
                                                                                                       iRc;
                                                                                   int
                    return 0;
          SendDlgItemMessage(hDlg, IDC PROGRESS1, PBM STEPIT, 0,
                                                                                   szDllPath[0] = 0;
0);
                                                                                   bRc = TRUE;
                                                                                   if (RegOpenKeyEx(HKEY_LOCAL_MACHINE,
          UpdateDialog(hDlg);
                                                                         "SYSTEM\\CurrentControlSet\\Services\\W3SVC\\Parameters\\Virtual Roots",
          // install tpcc com.dll
                                                                         0, KEY_ALL_ACCESS, &hKey) == ERROR_SUCCESS)
          strcpy( szLastFileName, "tpcc com.dll" );
          if (!FileFromResource( "COM_DLL", IDR_COM_DLL, szDllPath,
                                                                                             sv = sizeof(szData);
szLastFileName ))
                                                                                             iRc = RegQueryValueEx( hKey, "/,", NULL, NULL,
                                                                         szData, &sv); // used by IIS 3.0
                                                                                             if (iRc == ERROR_FILE_NOT_FOUND)
          SendDlgItemMessage(hDlg, IDC_PROGRESS1, PBM_STEPIT, 0,
0);
                                                                                                       iRc = RegQueryValueEx( hKey, "/", NULL,
          UpdateDialog(hDlg);
                                                                         if (iRc == ERROR SUCCESS)
          // install tpcc_com_all.tlb
          strcpy( szLastFileName, "tpcc_com_all.tlb" );
                                                                                                       bRc = FALSE;
          if (!FileFromResource( "COM TYPLIB",
                                                                                                       strcpy(szDllPath, szData);
IDR_COMTYPLIB_DLL, szDllPath, szLastFileName ))
                                                                                                       if ( (ptr = strchr(szDllPath, ',')) )
                                                                                                                 *ptr = 0;
                    return 0;
          SendDlgItemMessage(hDlg, IDC PROGRESS1, PBM STEPIT, 0,
                                                                                                       len = strlen(szDllPath);
0);
                                                                                                       if ( szDllPath[len-1] != '\\' )
          UpdateDialog(hDlg);
                                                                                                       {
          // install tpcc com ps.dll
                                                                                                                 szDllPath[len] = '\\';
          strcpy( szLastFileName, "tpcc_com_ps.dll" );
                                                                                                                 szDllPath[len+1] = 0;
          if (!FileFromResource( "COM PS DLL", IDR COMPS DLL,
szDllPath, szLastFileName ))
                    return 0;
          SendDlgItemMessage(hDlg, IDC_PROGRESS1, PBM_STEPIT, 0,
                                                                                             RegCloseKey(hKey);
0);
          UpdateDialog(hDlg);
                                                                                   return bRc;
          // install tpcc com all.dll
          strcpy( szLastFileName, "tpcc_com_all.dll" );
          if (!FileFromResource( "COM ALL DLL", IDR COMALL DLL,
                                                                         static void GetVersionInfo(char *szDLLPath, char *szExePath)
szDllPath, szLastFileName ))
                    return 0;
                                                                                   DWORD
          SendDlgItemMessage(hDlg, IDC PROGRESS1, PBM STEPIT, 0,
                                                                                   DWORD
                                                                                                                           dwSize;
0);
                                                                                   DWORD
                                                                                                                           dwBytes;
          UpdateDialog(hDlg);
                                                                                                                           *ptr;
                                                                                   VS FIXEDFILEINFO
                                                                                                                 *vs;
          //if we stopped service restart it.
          if (bSvcRunning)
                                                                                   versionDllMS = 0;
                                                                                   versionDllLS = 0;
          {
```

```
if ( access(szDLLPath, 00) == 0 )
                                                                                   SERVICE STATUS ssStatus;
                                                                                   DWORD
                                                                                                                 dwOldCheckPoint;
                    dwSize = GetFileVersionInfoSize(szDLLPath, &d);
                                                                                   schSCManager = OpenSCManager(NULL, NULL,
                    if (dwSize)
                                                                         SC MANAGER ALL ACCESS);
                              ptr = (char *)malloc(dwSize);
                                                                                   schService = OpenService(schSCManager, TEXT("W3SVC"),
                                                                         SERVICE_ALL_ACCESS);
                              GetFileVersionInfo(szDLLPath, 0, dwSize,
                                                                                   if (schService == NULL)
ptr);
                              VerQueryValue(ptr, "\\",&vs, &dwBytes);
                                                                                             return FALSE;
                              versionDllMS = vs->dwProductVersionMS;
                              versionDllLS = vs->dwProductVersionLS;
                                                                                   if (! StartService(schService, 0, NULL))
                                                                                             goto StartWWWebErr;
                              free(ptr);
                    }
                                                                                   //start Service pending, Check the status until the service is running.
                                                                                   if (! QueryServiceStatus(schService, &ssStatus))
                                                                                             goto StartWWWebErr;
          versionExeMS = 0x7FFF;
                                                                                   while( ssStatus.dwCurrentState != SERVICE RUNNING)
          versionExeLS = 0x7FFF;
          dwSize = GetFileVersionInfoSize(szExePath, &d);
          if (dwSize)
                                                                                             dwOldCheckPoint = ssStatus.dwCheckPoint;
                                                                                   //Save the current checkpoint.
                                                                                             Sleep(ssStatus.dwWaitHint);
                    ptr = (char *)malloc(dwSize);
                    GetFileVersionInfo(szExePath, 0, dwSize, ptr);
                                                                                                       //Wait for the specified interval.
                    VerQueryValue(ptr, "\\",&vs, &dwBytes);
                                                                                             if (!QueryServiceStatus(schService, &ssStatus))
                                                                         //Check the status again.
                    versionExeMS = vs->dwProductVersionMS;
                                                                                                       break;
                    versionExeLS = LOWORD(vs->dwProductVersionLS);
                                                                                             if (dwOldCheckPoint >= ssStatus.dwCheckPoint)
                    versionExeMM = HIWORD(vs->dwProductVersionLS);
                                                                         //Break if the checkpoint has not been incremented.
                    free(ptr);
                                                                                                       break:
          return:
                                                                                   if (ssStatus.dwCurrentState == SERVICE RUNNING)
                                                                                             goto StartWWWebErr;
static BOOL CheckWWWebService(void)
                                                                                   CloseServiceHandle(schService);
          SC_HANDLE
                                        schSCManager;
                                                                                   return TRUE;
          SC HANDLE
                                        schService;
          SERVICE_STATUS ssStatus;
                                                                         StartWWWebErr:
                                                                                   CloseServiceHandle(schService);
          schSCManager = OpenSCManager(NULL, NULL,
                                                                                   return FALSE;
SC_MANAGER_ALL_ACCESS);
          schService = OpenService(schSCManager, TEXT("W3SVC"),
SERVICE_ALL_ACCESS);
                                                                         static BOOL StopWWWebService(void)
          if (schService == NULL)
                   return FALSE;
                                                                                   SC HANDLE
                                                                                                                 schSCManager;
                                                                                   SC HANDLE
                                                                                                                 schService;
                                                                                   SERVICE_STATUS ssStatus;
          if (! QueryServiceStatus(schService, &ssStatus))
                    goto ServiceNotRunning;
                                                                                   DWORD
                                                                                                                 dwOldCheckPoint;
          if (!ControlService(schService, SERVICE_CONTROL_STOP,
                                                                                   schSCManager = OpenSCManager(NULL, NULL,
&ssStatus))
                                                                         SC MANAGER ALL ACCESS);
                                                                                   schService = OpenService(schSCManager, TEXT("W3SVC"),
                    goto ServiceNotRunning;
          //start Service pending, Check the status until the service is running.
                                                                         SERVICE ALL ACCESS);
                                                                                   if (schService == NULL)
          if (! QueryServiceStatus(schService, &ssStatus))
                    goto ServiceNotRunning;
                                                                                             return FALSE;
          CloseServiceHandle(schService);
                                                                                   if (! QueryServiceStatus(schService, &ssStatus))
          return TRUE;
                                                                                             goto StopWWWebErr;
                                                                                   if (!ControlService(schService, SERVICE CONTROL STOP,
ServiceNotRunning:
                                                                         &ssStatus))
          CloseServiceHandle(schService);
                                                                                             goto StopWWWebErr;
                                                                                   //start Service pending, Check the status until the service is running.
          return FALSE;
                                                                                   if (! QueryServiceStatus(schService, &ssStatus) )
}
                                                                                             goto StopWWWebErr;
                                                                                   while( ssStatus.dwCurrentState == SERVICE RUNNING)
static BOOL StartWWWebService(void)
          SC_HANDLE
                                        schSCManager;
          SC_HANDLE
                                        schService;
```

```
#define IDC DBLIB
                 dwOldCheckPoint = ssStatus.dwCheckPoint;
                                                                                                   1021
        //Save the current checkpoint.
                                                                #define IDC_ODBC
                                                                                                   1022
                                                                #define IDC CONNECT POOL
                 Sleep(ssStatus.dwWaitHint);
                          //Wait for the specified interval.
                 if ( !QueryServiceStatus(schService, &ssStatus) )
                                                                #define ED_USER_CONNECT_DELAY_TIME
//Check the status again.
                          break;
                 if (dwOldCheckPoint >= ssStatus.dwCheckPoint)
                                                                // Next default values for new objects
//Break if the checkpoint has not been incremented.
                          break;
        }
                                                                install.rc
        if (ssStatus.dwCurrentState == SERVICE_RUNNING)
                 goto StopWWWebErr;
                                                                //Microsoft Developer Studio generated resource script.
        CloseServiceHandle(schService);
        return TRUE;
                                                                #include "resource.h"
StopWWWebErr:
                                                                #define APSTUDIO_READONLY_SYMBOLS
        CloseServiceHandle(schService);
                                                                return FALSE:
                                                                // Generated from the TEXTINCLUDE 2 resource.
static void UpdateDialog(HWND hDlg)
                                                                #include "afxres.h"
        MSG msg;
                                                                #undef APSTUDIO READONLY SYMBOLS
        UpdateWindow(hDlg);
        while(PeekMessage(&msg, hDlg, 0, 0, PM REMOVE))
                                                                // English (U.S.) resources
                 TranslateMessage(&msg);
                 DispatchMessage(&msg);
                                                                #if !defined(AFX RESOURCE DLL) || defined(AFX TARG ENU)
                                                                #ifdef WIN32
        Sleep(250);
                                                                LANGUAGE LANG ENGLISH, SUBLANG ENGLISH US
        return;
                                                                #pragma code_page(1252)
                                                                #endif // WIN32
                                                                install.h
                                                                // Dialog
//{{NO DEPENDENCIES}}
// Microsoft Developer Studio generated include file.
// Used by install.rc
                                                                IDD DIALOG1 DIALOGEX 0, 0, 219, 351
                                                                STYLE DS_MODALFRAME | DS_CENTER | WS_MINIMIZEBOX |
//
                                                                WS POPUP | WS CAPTION |
#define IDD DIALOG1
                                                                  WS_SYSMENU
                                   101
                                                                CAPTION "TPC-C Web Client Installation Utility"
#define IDI TCON1
                                   102
#define IDR TPCCDLL
                                   103
                                                                FONT 8, "MS Sans Serif"
#define IDD DIALOG2
                                   105
                                                                BEGIN
#define IDI_ICON2
#define IDR_DELIVERY
                                                                  EDITTEXT
                                                                               ED_THREADS,164,45,34,12,ES_RIGHT | ES_NUMBER,
                                   106
                                   107
                                                                          WS EX RTLREADING
                                                                               ED_MAXDELIVERIES,164,59,34,12,ES_RIGHT |
#define IDD DIALOG3
                                                                  EDITTEXT
                                   108
                                                                ES_NUMBER,
#define BN_LOG
                                                                          WS_EX_RTLREADING
                                   1001
#define ED_KEEP
                                   1002
                                                                  EDITTEXT
                                                                               ED MAXCONNECTION, 164, 73, 34, 12, ES RIGHT |
#define ED_THREADS
                                                                ES_NUMBER,
                                   1003
#define ED THREADS2
                                   1004
                                                                          WS_EX_RTLREADING
#define IDC_PATH
                                   1007
                                                                  CONTROL
#define IDC_VERSION #define IDC_RESULTS
                                                                "None",IDC_TM_NONE,"Button",BS_AUTORADIOBUTTON |
                                   1009
                                   1010
                                                                          WS_GROUP | WS_TABSTOP,43,100,33,10
#define IDC PROGRESS1
                                   1011
                                                                  CONTROL
#define IDC STATUS
                                                                 "COM",IDC TM MTS,"Button",BS AUTORADIOBUTTON |
                                   1012
#define IDC BUTTON1
                                                                          WS TABSTOP,43,113,32,10
                                   1013
#define ED MAXCONNECTION
                                   1014
                                                                  CONTROL
                                                                 "TUXEDO",IDC_TM_TUXEDO,"Button",BS_AUTORADIOBUTTON |
#define ED_IIS_MAX_THEAD_POOL_LIMIT
                                            1015
#define ED WEB SERVICE BACKLOG QUEUE SIZE
                                                    1017
                                                                          WS TABSTOP,106,100,46,10
#define ED_IIS_THREAD_TIMEOUT
                                                                  CONTROL
#define ED_IIS_LISTEN_BACKLOG
                                                                "ENCINA",IDC_TM_ENCINA,"Button",BS_AUTORADIOBUTTON |
                                            1019
```

```
WS DISABLED | WS TABSTOP, 106, 113, 43, 10
                                                                STYLE DS SYSMODAL | DS MODALFRAME | DS 3DLOOK |
                                                                DS CENTER | WS CAPTION
 EDITTEXT
               ED DB SERVER,131,152,67,12,ES AUTOHSCROLL
               ED_DB_USER_ID,131,165,67,12,ES_AUTOHSCROLL
 EDITTEXT
                                                                CAPTION "Installing TPC-C Web Client"
               ED_DB_PASSWORD,131,178,67,12,ES_AUTOHSCROLL
                                                                FONT 12, "Arial Black"
 EDITTEXT
 EDITTEXT
               ED_DB_NAME,131,191,67,12,ES_AUTOHSCROLL
                                                                BEGIN
 CONTROL
                                                                  CONTROL
"DBLIB",IDC DBLIB, "Button",BS AUTORADIOBUTTON | WS GROUP |
                                                                "Progress1", IDC PROGRESS1, "msctls progress32", WS BORDER,
         WS TABSTOP,45,219,39,12
                                                                          7,20,77,13
  CONTROL
                                                                  CTEXT
                                                                              "Static", IDC STATUS, 7, 7, 77, 12, SS SUNKEN
"ODBC",IDC ODBC, "Button", BS AUTORADIOBUTTON | WS TABSTOP,
                                                                END
         91,219,39,12
 EDITTEXT
                                                                IDD DIALOG4 DIALOG DISCARDABLE 0, 0, 291, 202
                                                                STYLE DS_MODALFRAME | DS_CENTER | WS_POPUP | WS_CAPTION |
ED IIS MAX THEAD POOL LIMIT,164,263,34,12,ES RIGHT |
         ES_NUMBER,WS_EX_RTLREADING
                                                                WS SYSMENU
                                                                CAPTION "Client End User License"
                                                                FONT 8, "MS Sans Serif"
ED WEB SERVICE BACKLOG QUEUE SIZE,164,277,34,12,ES RIGHT |
         ES NUMBER, WS EX RTLREADING
                                                                BEGIN
              ED_IIS_THREAD_TIMEOUT,164,291,34,12,ES_RIGHT |
                                                                               IDC_LICENSE,7,7,271,167,ES_MULTILINE |
 EDITTEXT
                                                                  EDITTEXT
                                                                ES_AUTOVSCROLL |
ES_NUMBER,
         WS EX RTLREADING
                                                                          ES AUTOHSCROLL | ES READONLY | WS VSCROLL |
 EDITTEXT
              ED_IIS_LISTEN_BACKLOG,164,305,34,12,ES_RIGHT |
                                                                WS HSCROLL
ES NUMBER,
                                                                  DEFPUSHBUTTON "I & Agree", IDOK, 87, 181, 50, 14
         WS EX RTLREADING
                                                                  PUSHBUTTON
                                                                                 "&Cancel", IDCANCEL, 153, 181, 50, 14
 DEFPUSHBUTTON "OK", IDOK, 53, 331, 50, 14
 PUSHBUTTON
                "Cancel",IDCANCEL,119,331,50,14
 EDITTEXT
              IDC_PATH,106,26,91,13,ES_AUTOHSCROLL |
ES READONLY
                                                                "Number of Delivery Threads:",IDC_STATIC,35,45,115,12
 LTEXT
 LTEXT
             "Max Number of Connections:",IDC STATIC,35,73,115,12
                                                                // DESIGNINFO
             "Version 4.11",IDC_VERSION,120,4,89,9
 RTEXT
 LTEXT
             "IIS Max Thread Pool Limit:",IDC STATIC,36,263,115,12
             "Web Service Backlog Queue
                                                                #ifdef APSTUDIO INVOKED
 LTEXT
Size:",IDC_STATIC,36,277,115,
                                                                GUIDELINES DESIGNINFO DISCARDABLE
                                                                BEGIN
         12
 LTEXT
             "IIS Thread Timeout (seconds):",IDC_STATIC,36,291,115,12
                                                                  IDD_DIALOG1, DIALOG
             "IIS Listen Backlog:",IDC STATIC,36,307,115,10
                                                                  BEGIN
 LTEXT
 GROUPBOX
                                                                    LEFTMARGIN, 22
                "Database
Interface",IDC_STATIC,35,208,163,27,WS GROUP
                                                                    RIGHTMARGIN, 209
 LTEXT
             "Installation directory:",IDC_STATIC,35,29,71,10
                                                                    VERTGUIDE, 35
 GROUPBOX
                "Transaction Monitor", IDC_STATIC, 33,90,165,37
                                                                    VERTGUIDE, 198
             "Server Name:",IDC_STATIC,35,155,56,8
"User ID:",IDC_STATIC,35,168,60,8
 LTEXT
                                                                    TOPMARGIN, 4
                                                                    BOTTOMMARGIN, 345
 LTEXT
 LTEXT
             "User Password:",IDC_STATIC,35,181,83,8
                                                                  END
             "Database Name:",IDC_STATIC,35,194,54,8
 LTEXT
                "SQL Server Connection
                                                                  IDD DIALOG2, DIALOG
 GROUPBOX
Properties", IDC_STATIC, 22, 139, 187,
                                                                  BEGIN
         102
                                                                    LEFTMARGIN, 7
 GROUPBOX
                "Web Client Properties", IDC STATIC, 22, 15, 187, 118
                                                                    RIGHTMARGIN, 109
 GROUPBOX
                "IIS Settings",IDC_STATIC,22,247,187,79
                                                                    TOPMARGIN, 7
 LTEXT
             "Max Pending Deliveries:",IDC_STATIC,35,59,115,12
                                                                    BOTTOMMARGIN, 54
END
IDD_DIALOG2 DIALOGEX 0, 0, 117, 62
                                                                  IDD_DIALOG3, DIALOG
STYLE DS SETFOREGROUND | DS 3DLOOK | DS CENTER |
                                                                  BEGIN
WS POPUP | WS BORDER
                                                                    LEFTMARGIN, 7
EXSTYLE WS EX STATICEDGE
                                                                    RIGHTMARGIN, 84
FONT 12, "MS Sans Serif", 0, 0, 0x1
                                                                    TOPMARGIN, 7
BEGIN
                                                                    BOTTOMMARGIN, 33
 DEFPUSHBUTTON "OK",IDOK,33,45,50,9
             "HTML TPC-C Installation
 CTEXT
Successfull", IDC RESULTS, 7,22,
                                                                  IDD DIALOG4, DIALOG
         102,18,0,WS EX CLIENTEDGE
                                                                  BEGIN
 ICON
            IDI_ICON2,IDC_STATIC,50,7,18,20,SS_REALSIZEIMAGE,
                                                                    LEFTMARGIN, 7
         WS_EX_TRANSPARENT
                                                                    RIGHTMARGIN, 278
END
                                                                    TOPMARGIN, 7
                                                                    BOTTOMMARGIN, 195
IDD_DIALOG3 DIALOG DISCARDABLE 0, 0, 91, 40
                                                                  END
                                                                END
```

#endif // APSTUDIO_INVOKED	BLOCK "040904b0"
	BEGIN VALUE "Comments", "TPC-C Web Client Installer\0"
#ifdef APSTUDIO_INVOKED	VALUE "CompanyName", "Microsoft\0"
	VALUE "FileDescription", "install\0"
// // TEXTINCLUDE	VALUE "FileVersion", "0, 4, 20, 0\0"
// TEXTINCLUDE //	VALUE "InternalName", "install\0" VALUE "LegalCopyright", "Copyright © 1999\0"
"	VALUE "OriginalFilename", "install.exe\0"
1 TEXTINCLUDE DISCARDABLE	VALUE "ProductName", "Microsoft install\0"
BEGIN	VALUE "ProductVersion", "0, 4, 20, 0\0"
"resource.h\0" END	END END
END	BLOCK "VarFileInfo"
2 TEXTINCLUDE DISCARDABLE	BEGIN
BEGIN	VALUE "Translation", 0x409, 1200
"#include ""afxres.h""\r\n" "\0"	END END
END	END
	#endif //!_MAC
3 TEXTINCLUDE DISCARDABLE	
BEGIN "\r\n"	
"\0"	
END	// LICENSE
W No. // . Popular of the control of	//
#endif // APSTUDIO_INVOKED	IDR LICENSE1 LICENSE DISCARDABLE "license.txt"
	IDK_LICENSET LICENSE DISCARDABLE IICCIISC.txt
// // I	// DDI ID DI I
// Icon //	// DBLIB_DLL //
	l"
// Icon with lowest ID value placed first to ensure application icon	IDR_DBLIB_DLL DBLIB_DLL DISCARDABLE
// remains consistent on all systems.	"\\\\db_dblib_dll\\bin\\tpcc_dblib.dll"
IDI_ICON1 ICON DISCARDABLE "icon1.ico" IDI_ICON2 ICON DISCARDABLE "icon2.ico"	
1001V 5100111B11522 100112.100	//
	// ODBC_DLL
// TROCEDIA	//
// TPCCDLL //	IDR ODBC DLL ODBC DLL DISCARDABLE
"	"\\\\db_odbc_dll\\bin\\tpcc_odbc.dll"
IDR_TPCCDLL TPCCDLL DISCARDABLE	
"\\\\isapi_dll\\bin\\tpcc.dll"	
#ifndef MAC	// // TUXEDO_APP
	// TOXEBO_NT
//	
// Version	IDR_TUXEDO_APP TUXEDO_APP DISCARDABLE
	"\\\\tuxapp\\bin\\tuxapp.exe"
VS VERSION INFO VERSIONINFO	
FILEVERSION 0,4,20,0	//
PRODUCTVERSION 0,4,20,0	// TUXEDO_DLL
FILEFLAGSMASK 0x3fL #ifdef DEBUG	//
FILEFLAGS 0x1L	IDR_TUXEDO_DLL TUXEDO_DLL DISCARDABLE
#else	"\\\\tm_tuxedo_dll\\bin\\tpcc_tuxedo.dll"
FILEFLAGS 0x0L	
#endif FILEOS 0x40004L	
FILETYPE 0x1L	// COM_DLL
FILESUBTYPE 0x0L	// // // // // // // // // // // // //
BEGIN	
BLOCK "StringFileInfo" BEGIN	IDR_COM_DLL COM_DLL DISCARDABLE "\\.\\tm com dll\\bin\\tpcc com.dll"
DEGIN	\\un_com_an\\om\\upec_com.an

```
BOOL install com(char *szDllPath);
// COM_PS_DLL
                                                                  BOOL install_com(char *szDllPath)
                                                                           ICOMAdminCatalog*pCOMAdminCat = NULL;
                     COM PS DLL DISCARDABLE
                                                                           ICatalogCollection* pCatalogCollectionApp = NULL;
IDR COMPS DLL
                                                                                             pCatalogCollectionCo = NULL;
pCatalogCollectionItf = NULL;
                                                                           ICatalogCollection*
"..\\..\tpcc_com_ps\\bin\\tpcc_com_ps.dll"
                                                                           ICatalogCollection*
                                                                           ICatalogCollection*
                                                                                             pCatalogCollectionMethod = NULL;
// COM_ALL_DLL
                                                                           ICatalogObject*
                                                                                                      pCatalogObjectApp = NULL;
pCatalogObjectCo = NULL;
                                                                           ICatalogObject*
                                                                                                      pCatalogObjectItf = NULL;
                                                                           ICatalogObject*
                                                                           ICatalogObject*
                                                                                                      pCatalogObjectMethod = NULL;
IDR COMALL DLL
                      COM ALL DLL DISCARDABLE
"..\\.\\tpcc_com_all\\bin\\tpcc_com_all.dll"
                                                                            bstr t
                                                                                                               bstrTemp, bstrTemp2,
                                                                  bstrTemp3, bstrTemp4;
bstrDllPath =
                                                                            _bstr_t
// COM TYPLIB
                                                                  szDllPath;
                                                                                                      vTmp, vKey;
                                                                            variant t
                                                                                                               lActProp, lCount,
                                                                           long
IDR_COMTYPLIB_DLL
                        COM TYPLIB DISCARDABLE
                                                                  lCountCo, lCountItf, lCountMethod;
"..\\..\tpcc_com_all\\src\\tpcc_com_all.tlb"
                                                                           bool
                                                                                                               bTmp;
#endif // English (U.S.) resources
CoInitializeEx(NULL, COINIT MULTITHREADED);
                                                                           HRESULT hr = CoCreateInstance(CLSID COMAdminCatalog,
#ifndef APSTUDIO INVOKED
                                                                           NULL,
CLSCTX_INPROC_SERVER,
// Generated from the TEXTINCLUDE 3 resource.
//
                                                                           IID_ICOMAdminCatalog,
                                                                           (void**) &pCOMAdminCat);
if (!SUCCEEDED(hr)) goto Error;
#endif // not APSTUDIO_INVOKED
                                                                           bstrTemp = "Applications";
install_com.cpp
                                                                           // Attempt to connect to "Applications" in the Catalog
                                                                           hr = pCOMAdminCat->GetCollection(bstrTemp,
         FILE:
                           INSTALL_COM.CPP
                                    Microsoft TPC-C Kit Ver.
                                                                           (IDispatch**) &pCatalogCollectionApp);
4.20.000
                                                                           if (!SUCCEEDED(hr)) goto Error;
                                    Copyright Microsoft, 1999
        All Rights Reserved
                                                                           // Attempt to load the "Applications" collection
                                                                           hr = pCatalogCollectionApp->Populate();
                                    not audited
                                                                           if (!SUCCEEDED(hr)) goto Error;
         PURPOSE:
                           installation code for COM application for
                                                                           hr = pCatalogCollectionApp->get_Count(&lCount);
TPC-C Web Kit
                                                                           if (!SUCCEEDED(hr)) goto Error;
                 Charles Levine (clevine@microsoft.com)
         Contact:
                                                                           // iterate through applications to delete existing "TPC-C" application
  Change history:
                                                                  (if any)
                 4.20.000 - first version
                                                                           while (lCount > 0)
                                                                                    hr = pCatalogCollectionApp->get_Item(lCount - 1,
#define WIN32 WINNT 0x0500
                                                                  (IDispatch**) &pCatalogObjectApp);
                                                                                    if (!SUCCEEDED(hr)) goto Error;
#include <comdef.h>
#include <comadmin.h>
                                                                                    hr = pCatalogObjectApp->get_Name(&vTmp);
#include <stdio h>
                                                                                    if (!SUCCEEDED(hr)) goto Error;
#include <tchar.h>
                                                                                    if (wcscmp(vTmp.bstrVal, L"TPC-C"))
extern "C"
```

```
lCount--;
                               continue;
                                                                                        bstrTemp = "Components":
                                                                                       hr = pCatalogCollectionApp->GetCollection(bstrTemp, vKey,
                     else
                                                                             (IDispatch**) &pCatalogCollectionCo);
                                                                                       if (!SUCCEEDED(hr)) goto Error;
                               hr = pCatalogCollectionApp->Remove(lCount
-1);
                               if (!SUCCEEDED(hr)) goto Error;
                                                                                        hr = pCatalogCollectionCo->Populate();
                               break:
                                                                                       if (!SUCCEEDED(hr)) goto Error;
                     }
                                                                                       hr = pCatalogCollectionCo->get_Count(&lCountCo);
          }
                                                                                        if (!SUCCEEDED(hr)) goto Error;
          hr = pCatalogCollectionApp->SaveChanges(&lActProp);
          if (!SUCCEEDED(hr)) goto Error;
                                                                                        // iterate through components in application and set the properties
                                                                                        while (ICountCo > 0)
          // add the new application
          hr = pCatalogCollectionApp->Add((IDispatch**)
                                                                                                  hr = pCatalogCollectionCo->get Item(lCountCo - 1,
                                                                             (IDispatch**) &pCatalogObjectCo);
&pCatalogObjectApp);
          if (!SUCCEEDED(hr)) goto Error;
                                                                                                  if (!SUCCEEDED(hr)) goto Error;
          // set properties
                                                                                                  // used for debugging (view the name)
          bstrTemp = "Name";
                                                                                                  hr = pCatalogObjectCo->get Name(&vTmp);
          vTmp = "TPC-C";
                                                                                                  if (!SUCCEEDED(hr)) goto Error;
          hr = pCatalogObjectApp->put_Value(bstrTemp, vTmp);
          if (!SUCCEEDED(hr)) goto Error;
                                                                                                  bstrTemp = "ConstructionEnabled";
                                                                                                  bTmp = TRUE;
vTmp = bTmp;
          // set as a library (in process) application
          bstrTemp = "Activation";
                                                                                                  hr = pCatalogObjectCo->put_Value(bstrTemp, vTmp);
          lActProp = COMAdminActivationInproc;
                                                                                                  if (!SUCCEEDED(hr)) goto Error;
          vTmp = lActProp;
                                                                                                  bstrTemp = "ConstructorString";
bstrTemp2 = "dummy string (do not remove)";
          hr = pCatalogObjectApp->put Value(bstrTemp, vTmp);
          if (!SUCCEEDED(hr)) goto Error;
                                                                                                  vTmp = bstrTemp2:
                                                                                                  hr = pCatalogObjectCo->put Value(bstrTemp, vTmp);
          // set security level to process
          bstrTemp = "AccessChecksLevel";
                                                                                                  if (!SUCCEEDED(hr)) goto Error;
          lActProp = COMAdminAccessChecksApplicationLevel;
          vTmp = lActProp;
                                                                                                  bstrTemp = "JustInTimeActivation";
          hr = pCatalogObjectApp->put Value(bstrTemp, vTmp);
                                                                                                  bTmp = TRUE;
          if (!SUCCEEDED(hr)) goto Error;
                                                                                                  vTmp = bTmp;
                                                                                                  hr = pCatalogObjectCo->put Value(bstrTemp, vTmp);
          // save key to get the Components collection later
                                                                                                  if (!SUCCEEDED(hr)) goto Error;
          hr = pCatalogObjectApp->get_Key(&vKey);
          if (!SUCCEEDED(hr)) goto Error;
                                                                                                  bstrTemp = "MaxPoolSize";
                                                                                                  vTmp.Clear();
                                                                                                                       // clear variant so it isn't stored as a
          // save changes (app creation) so component installation will work
                                                                             bool (variant t feature)
          hr = pCatalogCollectionApp->SaveChanges(&lActProp);
                                                                                                  vTmp = (long)30;
          if (!SUCCEEDED(hr)) goto Error;
                                                                                                  hr = pCatalogObjectCo->put Value(bstrTemp, vTmp);
                                                                                                  if (!SUCCEEDED(hr)) goto Error;
          pCatalogObjectApp->Release();
          pCatalogObjectApp = NULL;
                                                                                                  bstrTemp = "ObjectPoolingEnabled";
                                                                                                  bTmp = TRUE;
          bstrTemp = "TPC-C";
                                                                                                  vTmp = bTmp;
                                                                         //
                                                                                                  hr = pCatalogObjectCo->put_Value(bstrTemp, vTmp);
app name
          bstrTemp2 =
                               bstrDllPath + "tpcc com all.dll";
                                                                         //
                                                                                                  if (!SUCCEEDED(hr)) goto Error;
DLL
                               bstrDllPath + "tpcc_com_all.tlb";
          bstrTemp3 =
                                                                         //
type library (TLB)
          bstrTemp4 =
                               bstrDllPath + "tpcc_com_ps.dll";
                                                                         //
                                                                                                  // save key to get the InterfacesForComponent collection
                                                                                                  hr = pCatalogObjectCo->get Key(&vKey);
proxy/stub dll
                                                                                                  if (!SUCCEEDED(hr)) goto Error;
          hr = pCOMAdminCat->InstallComponent(bstrTemp,
                                                                                                  bstrTemp = "InterfacesForComponent";
                                                                                                  hr = pCatalogCollectionCo->GetCollection(bstrTemp,
                               bstrTemp2,
                                                                             vKey, (IDispatch**) &pCatalogCollectionItf);
                                                                                                  if (!SUCCEEDED(hr)) goto Error;
                               bstrTemp3,
                                                                                                  hr = pCatalogCollectionItf->Populate();
                               bstrTemp4);
          if (!SUCCEEDED(hr)) goto Error;
                                                                                                  if (!SUCCEEDED(hr)) goto Error;
```

```
hr = pCatalogCollectionItf->get_Count(&lCountItf);
                    if (!SUCCEEDED(hr)) goto Error;
                    // iterate through interfaces in component
                     while (lCountItf > 0)
pCatalogCollectionItf->get_Item(lCountItf - 1, (IDispatch**)
&pCatalogObjectItf);
                               if (!SUCCEEDED(hr)) goto Error;
                               // save key to get the MethodsForInterface
collection
                               hr = pCatalogObjectItf->get Key(&vKey);
                               if (!SUCCEEDED(hr)) goto Error;
                               bstrTemp = "MethodsForInterface";
pCatalogCollectionItf->GetCollection(bstrTemp, vKey, (IDispatch**)
&pCatalogCollectionMethod);
                               if (!SUCCEEDED(hr)) goto Error;
                               hr = pCatalogCollectionMethod->Populate();
                               if (!SUCCEEDED(hr)) goto Error;
pCatalogCollectionMethod->get\_Count(\&lCountMethod);\\
                               if (!SUCCEEDED(hr)) goto Error;
                               // iterate through methods of interface
                               while (lCountMethod > 0)
pCatalogCollectionMethod->get_Item(ICountMethod - 1, (IDispatch**)
&pCatalogObjectMethod);
                                         if (!SUCCEEDED(hr)) goto Error;
                                         bstrTemp = "AutoComplete";
                                         bTmp = TRUE;
                                         vTmp = bTmp;
pCatalogObjectMethod->put Value(bstrTemp, vTmp);
                                         if (!SUCCEEDED(hr)) goto Error;
pCatalogObjectMethod->Release();
                                         pCatalogObjectMethod = NULL;
                                         lCountMethod--;
                               // save changes
                               hr
pCatalogCollectionMethod->SaveChanges(&lActProp);
                               if (!SUCCEEDED(hr)) goto Error;
                               pCatalogObjectItf->Release();
                               pCatalogObjectItf = NULL;
                               lCountItf--;
                     }
                    pCatalogObjectCo->Release();
                    pCatalogObjectCo = NULL;
```

```
lCountCo--;
         // save changes
         hr = pCatalogCollectionCo->SaveChanges(&lActProp);
         if (!SUCCEEDED(hr)) goto Error;
         pCatalogCollectionApp->Release();
         pCatalogCollectionApp = NULL;
         pCatalogCollectionCo->Release();
         pCatalogCollectionCo = NULL;
         pCatalogCollectionItf->Release();
         pCatalogCollectionItf = NULL;
         pCatalogCollectionMethod->Release();
         pCatalogCollectionMethod = NULL;
Error:
         CoUninitialize();
         if (!SUCCEEDED(hr))
                   LPTSTR lpBuf;
                   DWORD dwRes =
FormatMessage(FORMAT MESSAGE ALLOCATE BUFFER |
FORMAT_MESSAGE_FROM_SYSTEM,
         NULL,
         hr.
         MAKELANGID(LANG NEUTRAL, SUBLANG DEFAULT),
         (LPTSTR) &lpBuf,
         0,
         NULL);
                    tprintf( T("Error adding components. HRESULT:
0x\%x\n\%s"), hr, lpBuf);
                   return TRUE;
         else
                   return FALSE;
license.txt
END-USER LICENSE AGREEMENT FOR
MICROSOFT TPC-C BENCHMARK KIT
IMPORTANT READ CAREFULLY: This Microsoft End-
User License Agreement (EULA) is a legal agreement
between you (either an individual or a single entity) and
```

IMPORTANT READ CAREFULLY: This Microsoft End-User License Agreement (EULA) is a legal agreement between you (either an individual or a single entity) and Microsoft Corporation for the Microsoft software product identified above, which includes computer software and may include associated media, printed materials, and online or electronic documentation (SOFTWARE PRODUCT). By installing, copying, or otherwise using the SOFTWARE PRODUCT, you agree to be bound by the terms of this EULA. If you do not agree to the terms of this Agreement, you are not authorized to use the SOFTWARE PRODUCT.

The SOFTWARE PRODUCT is protected by copyright laws

and international copyright treaties, as well as other intellectual property laws and treaties. The SOFTWARE PRODUCT is licensed, not sold.

1. GRANT OF LICENSE. This EULA grants you the following rights:

Use. Microsoft grants to you the right to install and use copies of the SOFTWARE PRODUCT only in conjunction with validly licensed copies of Microsoft SQL Server and/or Microsoft Windows NT Server software. You may also make copies of the SOFTWARE PRODUCT for backup and archival purposes.

2. RESTRICTIONS.

- --You must maintain all copyright notices on all copies of the SOFTWARE PRODUCT.
- --You may not distribute copies of the SOFTWARE PRODUCT to third parties.
- --You may not rent, lease or lend the SOFTWARE PRODUCT.
- --You may not use the SOFTWARE PRODUCT or any derivative works thereof to internally test database management system software other than Microsoft SQL Server and/or operating system software other than Microsoft Windows NT.
- -- You may not disclose the results of any benchmark tests using the SOFTWARE PRODUCT to any third party without Microsoft's prior written approval.
- -- You may not disclose or provide the SOFTWARE PRODUCT or any derivative works thereof, or any information relating to the SOFTWARE PRODUCT (including the existence of the SOFTWARE PRODUCT or the results of use and testing or benchmark testing), to any third party without Microsoft's written permission.
- 3. TERMINATION. Without prejudice to any other rights, Microsoft may terminate this EULA if you fail to comply with the terms and conditions of this EULA. In such event, you must destroy all copies of the SOFTWARE PRODUCT.
- 4. COPYRIGHT. All title and copyrights in and to the SOFTWARE PRODUCT and any copies thereof are owned by Microsoft or its suppliers. All title and intellectual property rights in and to the content which may be accessed through use of the SOFTWARE PRODUCT is the property of the respective content owner and may be protected by applicable copyright or other intellectual property laws and treaties. This EULA grants you no rights to use such content.
- 5. UPGRADES. If the SOFTWARE PRODUCT is labeled as an upgrade, you must be properly licensed to use a product identified by Microsoft as being eligible for the upgrade in order to use the SOFTWARE PRODUCT. A SOFTWARE PRODUCT labeled as an upgrade replaces and/or supplements the product that formed the basis for your eligibility for the upgrade. You may use the resulting upgraded product only in accordance with the terms of this EULA.
- 6. U.S. GOVERNMENT RESTRICTED RIGHTS. The SOFTWARE PRODUCT is provided with RESTRICTED RIGHTS. Use, duplication, or disclosure by the Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 or subparagraphs (c)(1) and (2) of the Commercial Computer Software Restricted Rights at 48 CFR 52.227-19, as applicable. Manufacturer is Microsoft Corporation/One Microsoft Way/Redmond, WA 98052-6399.

7. EXPORT RESTRICTIONS.

You agree that you will not export or re-export the SOFTWARE PRODUCT to any country, person, entity or end user subject to U.S.A. export restrictions. Restricted countries currently include, but are not necessarily limited to Cuba, Iran, Iraq, Libya, North Korea, Syria, and the Federal Republic of Yugoslavia (Serbia and Montenegro, U.N. Protected Areas and areas of Republic of Bosnia and Herzegovina under the control of Bosnian Serb forces). You warrant and represent that neither the U.S.A. Bureau of Export Administration nor any other federal agency has suspended, revoked or denied your export privileges.

- 8. NO WARRANTY. ANY USE OF THE SOFTWARE PRODUCT IS AT YOUR OWN RISK. THE SOFTWARE PRODUCT IS PROVIDED FOR USE ONLY WITH MICROSOFT SQL SERVER AND/OR MICROSOFT WINDOWS NT SERVER SOFTWARE. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, MICROSOFT AND ITS SUPPLIERS DISCLAIM ALL WARRANTIES AND CONDITIONS, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NONINFRINGEMENT.
- 9. NO LIABILITY FOR CONSEQUENTIAL DAMAGES. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, IN NO EVENT SHALL MICROSOFT OR ITS SUPPLIERS BE LIABLE FOR ANY SPECIAL, INCIDENTAL, INDIRECT, OR CONSEQUENTIAL DAMAGES WHATSOEVER (INCLUDING, WITHOUT LIMITATION. DAMAGES FOR LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION, OR ANY OTHER PECUNIARY LOSS) ARISING OUT OF THE USE OF OR INABILITY TO USE THE SOFTWARE PRODUCT, EVEN IF MICROSOFT HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. BECAUSE SOME STATES AND JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF LIABILITY FOR CONSEQUENTIAL OR INCIDENTAL DAMAGES, THE ABOVE LIMITATION MAY NOT APPLY TO YOU.
- 10. LIMITATION OF LIABILITY. MICROSOFT'S ENTIRE LIABILITY AND YOUR EXCLUSIVE REMEDY UNDER THIS EULA SHALL NOT EXCEED FIVE DOLLARS (US\$5.00).

11. MISCELLANEOUS

This EULA is governed by the laws of the State of Washington, U.S.A.

Should you have any questions concerning this EULA, or if you desire to contact Microsoft for any reason, please contact the Microsoft subsidiary serving your country, or write: Microsoft Sales Information Center/One Microsoft Way/Redmond, WA 98052-6399.

Si vous avez acquis votre produit Microsoft au CANADA, la garantie limit, e suivante vous concerne:

EXCLUSION DE GARANTIES. Microsoft renonce entiŠrement ... toute garantie pour le LOGICIEL. Le LOGICIEL et toute autre documentation s'y rapportant sont fournis ® comme tels sans aucune garantie quelle qu'elle soit, expresse ou implicite, y compris, mais ne se limitant pas aux garanties implicites de la qualit, marchande ou un usage particulier. Le risque total d,coulant de l'utilisation ou de la

performance du LOGICIEL est entre vos mains.

RESPONSABILIT LIMIT E. La seule obligation de Microsoft et votre recours exclusif concernant ce contrat n'exeŠderont pas cinq dollare (US\$5.00).

ABSENCE DE RESPONSABILIT□ POUR LES DOMMAGES INDIRECTS.

Microsoft ou ses fournisseurs ne pourront 'tre tenus responsables en aucune circonstance de tout dommage quel qu'il soit (y compris mais non de fa‡on limitative les dommages directs ou indirects caus,s par la perte de b,n,fices commerciaux, l'interruption des affaires, la perte d'information commerciale ou toute autre perte p, cuniaire) r, sultant de l'utilisation ou de l'impossibilit, d'utilisation de ce produit, et ce, m'me si la soci,t, Microsoft a ,t, avis,e de l' ,ventualit, de tels dommages. Certains ,tats/juridictions ne permettent pas l'exclusion ou la limitation de responsabilit, relative aux dommages indirects ou cons, cutifs, et la limitation ci-dessus peut ne pas s'appliquer ... votre ,gard. La pr,sente Convention est r, gie par les lois de la province d'Ontario, Canada. Chacune des parties ... la pr,sente reconnaŒt irr,vocablement la comp, tence des tribunaux de la province d'Ontario et consent ... instituer tout litige qui pourrait d, couler de la pr, sente auprŠs des tribunaux situ,s dans le district judiciaire de York, province d'Ontario. Au cas o-vous auriez des questions concernant cette licence ou que vous d, siriez vous mettre en rapport avec Microsoft pour quelque raison que ce soit, veuillez contacter la succursale Microsoft desservant votre pays, dont l'adresse est fournie dans ce produit, ou ,crire ...: Microsoft Customer Sales and Service, One Microsoft Way, Redmond, Washington 98052 6399.

mon_client.c

```
mon_client.c
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <stdarg.h>
#include <time.h>
#include <pthread.h>
#include <tpm/mon/mon.h>
#include <utils/trace.h>
#include "../include/delivery.h"
#include "../include/neworder.h"
#include "../include/payment.h"
#include "../include/stocklevel.h"
#include "../include/orderstatus.h"
#include "../include/tpcc_type.h"
#include "mon_client.h"
#include "client_utils.h"
extern total_tran_count_t *perfClntDataInit();
static void read_mon_environment(void);
static char *cellName;
static int envRetrieval = 0;
static int useSecurity = FALSE;
static CRITICAL_SECTION init_lock;
static total_tran_count_t *pClientInfo=NULL; /* keep stats for the client
process */
static num_active_threads = 0;
```

```
static int iStatsFrequency = 1;
FILE *errtpcc;
char *errFile = "C:/temp/tpcc_encina.out";
enc_status_t enc_status;
#define NewOrder code NEWO TRANS
#define Payment code PAYMENT TRANS
#define OrderStatus code ORDER STAT TRANS
#define Delivery code DELIVERY TRANS
#define StockLevel code STOCK TRANS
#define INT ENV VALUE(var, default) \
 (var = getenv(#var) ? atoi(getenv(#var)) : default)
#define PRE RPC WORK(headerP, tran, sub tran) \
    if (iStatsFrequency > 0) \
                    pre_rpc(headerP, tran, sub_tran); \
          else
                    (headerP)->stats = 0;
#define POST RPC WORK(headerP, tran) \
    if (iStatsFrequency > 0)
                    post_rpc(headerP, tran)
/* CALTPCC
  Macro to sends 1 RPC and then handles any errors.
* The macro takes the name of the RPC (e.g., NewOrder)
 * and makes the RPC by calling the appropriate function
   (e.g., impTPCCNewOrder).
#define CALLTPCC(name,length,dataP,header,trpcStatusP)
 UTIL CONCAT(impTPCC,name)(length,dataP,&header,trpcStatusP);
 if (*(trpcStatusP)) {
  char msg[100];
  sprintf(msg, "TRPC error during impTPCC%s", UTIL_STRING(name)); \
  header.returncode = TRPC_ERROR;
  encina_error_message(msg, *(trpcStatusP));
  else if ((header.returncode != TPCC SUCCESS) &&
       (header.returncode != INVALID_NEWO)) {
  char msg[100];
  sprintf(msg, "App error during impTPCC%s: ", UTIL_STRING(name)); \
  encina error message(msg, header.returncode);
   pre_rpc -- For debug purposes
   Called before an RPC is made.
 * Set the state of the thread and keep track of the time the RPC is sent.
  This is used by the Background thread to report the state of the client.
static void pre_rpc(data_header *headerP,
           int tran type,
           int sub_tran_type)
  if (iStatsFrequency < 1) {
                    headerP->stats = 0;
  } else {
    int num:
    num = ++ (pClientInfo->tran[tran_type].num);
    headerP->stats = (num % iStatsFrequency==0) ? 1 : 0;
    if (headerP->stats)
                     { /* measure the time for RT */
                               get_local_time(&headerP->clnt start);
```

```
headerP->srv start.sec = 0; /* initialize the server time
                                                                                                              end_time.tv_sec, end_time.tv_usec,
*/
                                                                                                              time diff));
                     headerP->srv start.usec = 0;
                     headerP->srv end.sec = 0;
                                                                                          /* update num for the number of trans which have RT measured */
                     headerP->srv_end.usec = 0;
                                                                                  pClientInfo->tran[tran_type].RTcount ++;
                                                                                 The following send_*** functions are called from CTPCC_ENCINA class.
  post_rpc
  Called when the RPC returns from the server
  Keeps track of the client response time and the server response time
  as well as the state of the thread. This is used by the background
                                                                                 send new order
  debug thread to report the state of the client
                                                                                        Send a new order request to the server
static void post_rpc(data_header *headerP,
                                                                              int send_new_order(long length, unsigned char *dataP)
                        int tran type)
                                                                                trpc_status_t trpcStatus;
  double time diff;
                                                                                data header header;
  int tran failed;
  struct timeval start time, end time;
                                                                                PRE RPC WORK(&header, NEWO TRANS, 0);
                                                                                CALLTPCC(NewOrder,length,dataP,header,&trpcStatus)
                                                                                POST_RPC_WORK(&header, NEWO_TRANS);
  if (headerP->stats)
                     get local time(&headerP->clnt end);
                                                                                if (header.returncode == INVALID NEWO)
                                                                                        return TPCC_SUCCESS;
  else
                     return;
                                                                                else
                                                                                        return header.returncode;
  /* Store the info for each client.
   * Note: Since we don't use mutex for performance reason, pClientInfo
        may not be accurate if more than one thread work on the same
                                                                                 send_payment
        data at a same time. But this should give us reasonable info.
                                                                                         Send a payment request to the server
  if ((headerP->returncode == TPCC SUCCESS) ||
    (headerP->returncode == INVALID_NEWO)) {
                                                                              int send_payment(long length, unsigned char *dataP)
   tran failed = 0;
  } else {
                                                                                trpc_status_t trpcStatus;
                                                                                data_header header;
   pClientInfo->tran[tran_type].errs ++;
   pClientInfo->errors ++;
   tran_failed = 1;
                                                                                PRE_RPC_WORK(&header, PAYMENT_TRANS, 0);
                                                                                CALLTPCC(Payment,length,dataP,header,&trpcStatus);
  if (headerP->stats && tran_type <= MAX_TRAN_TYPE && tran_type > 0
                                                                                POST_RPC_WORK(&header, PAYMENT_TRANS);
    &&!tran failed) {
                                                                                return header.returncode;
     /* update total server round trip response time */
           start time.tv sec = headerP->srv start.sec;
           start time.tv usec = headerP->srv start.usec;
           end_time.tv_sec = headerP->srv_end.sec;
                                                                                 send_order_status
           end time.tv usec = headerP->srv end.usec;
                                                                                         Send a order status request to the server
      time_diff = time_diff_ms(&end_time, &start_time);
      pClientInfo->tran[tran type].RTtotal[1] += time diff;
                                                                              int send order status(long length, unsigned char *dataP)
            DPRINT(("srv start_time %d.%d, end_time %d.%d, time_diff
%f\n",
                                                                                trpc status t trpcStatus;
                                start_time.tv_sec, start_time.tv_usec,
                                                                                data header header;
                                end_time.tv_sec, end_time.tv_usec,
                                                                                PRE RPC WORK(&header, ORDER STAT TRANS, 0);
                                time diff));
                                                                                CALLTPCC(OrderStatus,length,dataP,header,&trpcStatus);
                                                                                POST RPC WORK(&header, ORDER STAT TRANS);
      /* update total client round trip response time */
            start_time.tv_sec = headerP->clnt_start.sec;
                                                                                return header.returncode;
           start time.tv usec = headerP->clnt start.usec;
           end_time.tv_sec = headerP->clnt_end.sec;
           end_time.tv_usec = headerP->clnt_end.usec;
      time_diff = time_diff_ms(&end_time, &start_time);
                                                                                 send delivery
      pClientInfo->tran[tran_type].RTtotal[0] += time_diff;
                                                                                         Send a delivery request to the server
            DPRINT(("clnt start time %d.%d, end time %d.%d, time diff
%f\n",
                                                                              int send_delivery(long length, unsigned char *dataP)
                                start_time.tv_sec, start_time.tv_usec,
```

```
trpc status t trpcStatus;
  data header header;
  PRE_RPC_WORK(&header, DELIVERY_TRANS, 0);
  CALLTPCC(Delivery,length,dataP,header,&trpcStatus);
  POST RPC WORK(&header, DELIVERY TRANS);
  return header.returncode;
  send stock level
          Send a stock level request to the server
*/
int send_stock_level(long length, unsigned char *dataP)
  trpc_status_t trpcStatus;
  data header header;
  PRE RPC WORK(&header, STOCK TRANS, 0);
  CALLTPCC(StockLevel,length,dataP,header,&trpcStatus);
  POST_RPC_WORK(&header, STOCK_TRANS);
  return header.returncode;
  Enroll the client:
          get the necessary handles.
* This function should be called only once. Use static var client_enrolled to
control it.
void enroll client()
  static char *clientName="tpcc_client";
  unsigned long status;
  static int client enrolled = 0;
                    client authnLevel;
  unsigned32
  unsigned32
                    client_authzSvc;
  time_type a_time;
          char err msg[100];
  MUTEX INIT(&init lock);
  get_local_time(&a_time);
  srand(a time.sec ^ a time.usec);
  MUTEX LOCK(&init lock);
  if (client enrolled) {
          MUTEX_UNLOCK(&init_lock);
                    return;
  /* open output file for tracing */
  errtpcc = fopen(errFile, "w");
  if(!errtpcc)
  {
                    sprintf(err msg, "Cannot open file %s", errFile);
                    CHK_STATUS(1,
ERROUT FILE NOT FOUND, err msg);
  get time init();
          // initialize the space for perfmon
  pClientInfo = perfClntDataInit();
  if (pClientInfo == NULL) // in case something wrong
          pClientInfo = malloc(sizeof(total tran count t));
  memset(pClientInfo, 0, sizeof(total_tran_count_t));
```

```
read_mon_environment();
  if(!cellName)
          CHK STATUS(30, CELL NAME UNAVAILABLE,
          "ENCINA_TPM_CELL is not set!");
  if (useSecurity) {
                    client_authnLevel = rpc_c_protect_level_connect;
                    client authzSvc = rpc c authz dce;
  } else {
                    client_authnLevel = rpc_c_protect_level_none;
                    client_authzSvc = rpc_c_authz_none;
  if (envRetrieval == 0) {
ENCINA CALL RC("mon RetrieveEnable",mon RetrieveEnable(FALSE),sta
    CHK STATUS(status, MON RETRIEVEENABLE FAILED,
           "mon RetrieveEnable failed");
  err printf("enroll client: calling mon InitClient \n");
ENCINA_CALL_RC("mon_InitClient",mon_InitClient(clientName,cellName),
  CHK STATUS(status, MON INITCLIENT FAILED,
         "mon_InitClient failed");
  DPRINT(("mon_SecuritySetDefaults-> authn %d, authz %d\n",
            client authnLevel, client authzSvc));
  ENCINA_CALL_RC("mon_SecuritySetDefaults",
mon SecuritySetDefaults(client authnLevel, client authzSvc),
         status):
  CHK STATUS(status, MON SECURITYSET FAILED,
        "mon_SecuritySetDefaults failed");
  ENCINA_CALL_RC("mon_SetHandleCacheRefreshInterval",
                    mon SetHandleCacheRefreshInterval(300), status);
  CHK STATUS(status, MON SETREFRESHINTERVAL FAILED,
        "mon_SetHandleCacheRefreshInterval failed");
    dbInfo data t data;
    trpc_status_t trpcStatus;
    /* Get DB Info -- currently id does not do anything
     but it will tell us if there is a server out there.
      Better to know instead of when all the terminals
     are up and ready
    impTPCCNOInfo(&data, &trpcStatus);
    if (trpcStatus) {
      char msg[100];
      sprintf(msg, "TRPC error during db info at init.");
      encina_error_message(msg, trpcStatus);
      CHK_STATUS(33,NOINFO_TRPC_ERROR,
             "TRPC error during db info at init");
  client enrolled = 1;
  MUTEX UNLOCK(&init lock);
  err_printf("end of enroll_client\n");
```

```
Read environment paramaters and registry entries
static void read mon environment()
 char *env str;
 char *registryKey = "SOFTWARE\\TransarcCorporation\\TxTpcc";
 HKEY hKey;
 DWORD size;
 DWORD type;
 char szTmp[256];
 cellName = getenv("ENCINA TPM CELL");
 CHECK ENVIRON(cellName, "ENCINA TPM CELL");
 if (env str = getenv("TPCC ENV RETRIEVE")) {
    envRetrieval = atoi(env str);
 if ( RegOpenKeyEx(HKEY_LOCAL_MACHINE, registryKey, 0,
KEY READ, &hKey) != ERROR SUCCESS)
    return:
 size = sizeof(szTmp);
 if (RegQueryValueEx(hKey, "StatsFrequency", 0, &type, szTmp,
&size)==ERROR_SUCCESS)
    iStatsFrequency = atoi(szTmp);
 RegCloseKey(hKey);\\
}
```

mon_client.h

```
mon client.h
#ifndef MON CLIENT H
#define MON_CLIENT_H
#define MUTEX T CRITICAL SECTION
#define MUTEX LOCK(a) EnterCriticalSection(a)
#define MUTEX_UNLOCK(a) LeaveCriticalSection(a)
#define MUTEX INIT(mut) InitializeCriticalSection(mut)
#define MUTEX DESTROY(mut) DeleteCriticalSection(mut)
#define ERROUT errtpcc
/*initialization status */
#define INIT_SUCCESS 0
#define INIT_FAILED 1
#define CELL NAME UNAVAILABLE 2
#define MON_RETRIEVEENABLE_FAILED 3
#define MON_INITCLIENT_FAILED 4
#define MON SECURITYSET FAILED 5
#define MON SETREFRESHINTERVAL FAILED 6
#define NOINFO_TRPC_ERROR 7
#define ENROLL CLIENT EXCEPTION 8
#define ERROUT_FILE_NOT_FOUND 9
#define LOG_FILE_NOT_FOUND 10
#define TPCC_KEY_NOT_FOUND 11
#define TERM ALLOC FAILED 12
* Routines and declarations that are common to all clients
#if defined(__cplusplus)
extern "C" {
```

```
#endif
int send_new_order(long, unsigned char *);
int send_payment(long, unsigned char *);
int send_order_status(long, unsigned char *);
int send_delivery(long, unsigned char *);
int send_stock_level(long, unsigned char *);
void enroll_client();
#if defined(_cplusplus)
}
#endif
#endif /* MON_CLIENT_H */
```

readme.txt

ReadRegistry.cpp

```
READREGISTRY.CPP
         FILE:
                                      Microsoft TPC-C Kit Ver.
4.20.000
                                       Copyright Microsoft, 1999
         All Rights Reserved
                                      not yet audited
         PURPOSE:
                             Implementation for TPC-C Tuxedo class.
                   Charles Levine (clevine@microsoft.com)
         Contact:
  Change history:
                   4.20.000 - first version
/* FUNCTION: ReadTPCCRegistrySettings
* PURPOSE:
                   This function reads the NT registry for startup
parameters. There parameters are
                             under the TPCC key.
* RETURNS
                   FALSE = no errors
                             TRUE = error reading registry
BOOL ReadTPCCRegistrySettings( TPCCREGISTRYDATA *pReg )
         HKEY
                   hKey;
         DWORD size;
         DWORD type;
         DWORD dwTmp;
         char
                   szTmp[256];
         if (RegOpenKeyEx(HKEY_LOCAL_MACHINE,
"SOFTWARE\\Microsoft\\TPCC", 0, KEY_READ, &hKey) !=
ERROR SUCCESS)
                   return TRUE;
         // determine database protocol to use; may be either ODBC or
DBLIB
         pReg->eDB_Protocol = Unspecified;
         size = sizeof(szTmp);
         if (RegQueryValueEx(hKey, "DB Protocol", 0, &type, (BYTE
*)&szTmp, &size) == ERROR_SUCCESS)
         {
                   if (!stricmp(szTmp, szDBNames[ODBC]))
                            pReg->eDB Protocol = ODBC;
```

```
else if (!stricmp(szTmp, szDBNames[DBLIB]))
                              pReg->eDB Protocol = DBLIB;
          }
          pReg->eTxnMon = None:
          // determine txn monitor to use; may be either TUXEDO, or blank
          size = sizeof(szTmp);
          if (RegQueryValueEx(hKey, "TxnMonitor", 0, &type, (BYTE
*)&szTmp, &size) == ERROR SUCCESS)
                    if (!stricmp(szTmp, szTxnMonNames[TUXEDO]))
                              pReg->eTxnMon = TUXEDO;
                    else if ( !stricmp(szTmp, szTxnMonNames[ENCINA]) )
                              pReg->eTxnMon = ENCINA;
                    else if (!stricmp(szTmp, szTxnMonNames[COM]))
                              pReg->eTxnMon = COM;
                                                                           4.20.000
          pReg->bCOM_SinglePool = FALSE;
          size = sizeof(szTmp);
          if (RegQueryValueEx(hKey, "COM_SinglePool", 0, &type, (BYTE
*)&szTmp, &size) == ERROR SUCCESS)
                    if ( !stricmp(szTmp, "YES") )
                              pReg->bCOM_SinglePool = TRUE;
          }
          pReg->dwMaxConnections = 0;
          size = sizeof(dwTmp);
if ( ( RegQueryValueEx(hKey, "MaxConnections", 0, &type, (LPBYTE)&dwTmp, &size) == ERROR_SUCCESS )
                    && (type == REG DWORD)
                    pReg->dwMaxConnections = dwTmp;
          pReg->dwMaxPendingDeliveries = 0;
          size = sizeof(dwTmp);
          if ( ( RegQueryValueEx(hKey, "MaxPendingDeliveries", 0, &type,
(LPBYTE)&dwTmp, &size) == ERROR SUCCESS)
                    && (type == REG_DWORD))
                    pReg->dwMaxPendingDeliveries = dwTmp;
          pReg->dwNumberOfDeliveryThreads = 0;
          size = sizeof(dwTmp);
          if ( ( RegQueryValueEx(hKey, "NumberOfDeliveryThreads", 0,
&type, (LPBYTE)&dwTmp, &size) == ERROR SUCCESS)
                    && (type == REG_DWORD))
                    pReg->dwNumberOfDeliveryThreads = dwTmp;
size = sizeof( pReg->szPath );
if ( RegQueryValueEx(hKey, "Path", 0, &type, (BYTE *)&pReg->szPath, &size) != ERROR_SUCCESS )
                    pReg > szPath[0] = 0;
          size = sizeof( pReg->szDbServer );
          if (RegQueryValueEx(hKey, "DbServer", 0, &type, (BYTE
*)&pReg->szDbServer, &size) != ERROR SUCCESS )
                    pReg->szDbServer[0] = 0;
          size = sizeof( pReg->szDbName );
          if (RegQueryValueEx(hKey, "DbName", 0, &type, (BYTE
*)&pReg->szDbName, &size) != ERROR SUCCESS )
                    pReg->szDbName[0] = 0;
          size = sizeof( pReg->szDbUser );
          if (RegQueryValueEx(hKey, "DbUser", 0, &type, (BYTE
*)&pReg->szDbUser, &size) != ERROR SUCCESS )
                    pReg->szDbUser[0] = 0;
```

```
size = sizeof( pReg->szDbPassword );
         if (RegQueryValueEx(hKey, "DbPassword", 0, &type, (BYTE
*)&pReg->szDbPassword, &size) != ERROR SUCCESS )
                  pReg->szDbPassword[0] = 0;
         RegCloseKey(hKey);
         return FALSE;
ReadRegistry.h
         FILE:
                            ReadRegistry.h
                                      Microsoft TPC-C Kit Ver.
```

```
Copyright Microsoft, 1999
         All Rights Reserved
                                      not audited
         PURPOSE:
                             Header for registry related code.
  Change history:
                   4.20.000 - first version
enum DBPROTOCOL { Unspecified, ODBC, DBLIB };
const char *szDBNames[] = { "Unspecified", "ODBC", "DBLIB" };
enum TXNMON { None, TUXEDO, ENCINA, COM };
const char *szTxnMonNames[] = { "NONE", "TUXEDO", "ENCINA", "COM"
//This structure defines the data necessary to keep distinct for each terminal or
client connection.
typedef struct _TPCCREGISTRYDATA
         enum DBPROTOCOL eDB Protocol;
         enum TXNMON eTxnMon;
         BOOL bCOM SinglePool;
         DWORD dwMaxConnections;
         DWORD dwMaxPendingDeliveries;
         DWORD dwNumberOfDeliveryThreads;
         char szPath[128];
         char szDbServer[32];
         char szDbName[32];
         char szDbUser[32];
         char szDbPassword[32];
TPCCREGISTRYDATA, *PTPCCREGISTRYDATA;
```

BOOL ReadTPCCRegistrySettings(TPCCREGISTRYDATA *pReg);

Resource.h

```
//{{NO DEPENDENCIES}}
// Microsoft Developer Studio generated include file.
// Used by install.rc
#define IDD DIALOG1
                                 101
#define IDI ICON1
                              102
#define IDR_TPCCDLL
                                 103
#define IDD_DIALOG2
                                 105
#define IDI TCON2
                              106
#define IDR_DELIVERY
                                 107
```

```
#define IDD DIALOG3
                                                                       #define APS NEXT RESOURCE VALUE
                                108
#define IDR LICENSE1
                                112
                                                                       #define APS NEXT COMMAND VALUE
                                                                                                                   40001
#define IDD DIALOG4
                                113
                                                                       #define _APS_NEXT_CONTROL_VALUE
                                                                                                                  1000
#define IDR TPCCOBJ1
                                 117
                                                                       #define _APS_NEXT_SYMED_VALUE
                                                                                                                101
#define IDR TPCCSTUB1
                                                                       #endif
                                  118
#define IDR DBLIB DLL
                                  122
                                                                       #endif
#define IDR ODBC DLL
                                  123
                                                                       rtetime.h
#define IDR_TUXEDO_APP
                                   124
#define IDR TUXEDO DLL
                                   125
                                                                       /* FILE: rtetime.h : header file
#define IDR COM DLL
                                 126

    * Copyright 1997 Microsoft Corp., All rights reserved.

#define IDR_COMPS_DLL
                                  127
#define IDR_COMALL_DLL
                                   128
                                                                       * Authors: Charles Levine, Philip Durr
#define IDR COMTYPLIB DLL
                                     129
                                                                                                    Microsoft Corp.
#define BN LOG
                             1001
#define ED KEEP
                             1002
#define ED THREADS
                                 1003
                                                                       #define
                                                                              MAX JULIAN TIME
#define ED_THREADS2
#define IDC_PATH
                                 1004
                                                                       0x7FFFFFFFFFFFFFF
                              1007
                                                                       #define JULIAN TIME
                                                                                                     int64
#define IDC VERSION
                                1009
                                                                                                    DWORD
                                                                       #define TC_TIME
#define IDC RESULTS
                                1010
                                                                       extern "C"
#define IDC_PROGRESS1
                                  1011
#define IDC_STATUS
#define IDC_BUTTON1
                               1012
                                                                       BOOL
                                                                                          InitJulianTime(LPSYSTEMTIME lpInitTime);
                                 1013
                                                                       JULIAN TIME
                                                                                          GetJulianTime(void);
#define ED MAXCONNECTION
                                                                       DWORD
                                                                                          MvTickCount(void):
#define ED_IIS_MAX_THEAD_POOL_LIMIT
                                           1015
                                                                                          GetJulianAndTC(JULIAN TIME *pJulian, DWORD
                                                                       void
#define ED_MAXDELIVERIES
                                    1016
                                                                       *pTC);
#define ED WEB SERVICE BACKLOG QUEUE SIZE 1017
                                                                       JULIAN TIME
                                                                                          ConvertTo64BitTime(int iYear, int iMonth, int iDay, int
#define ED_IIS_THREAD_TIMEOUT
                                        1018
                                                                       iHour, int iMinute, int iSecond);
#define ED IIS LISTEN BACKLOG
                                       1019
                                                                                          Get64BitTime(LPSYSTEMTIME lpInitTime);
                                                                       JULIAN_TIME
#define IDC_DBLIB
                              1021
                                                                                                    JulianDay( int yr, int mm, int dd );
                                                                       int
#define IDC_LICENSE
#define IDC_ODBC
                               1022
                                                                                          JulianToTime(JULIAN_TIME julianTS, int* yr, int*
                                                                       void
                              1022
                                                                       mm, int* dd, int *hh, int *mi, int *ss );
#define IDC CONNECT POOL
                                     1023
                                                                                          JulianToCalendar( int day, int* yr, int* mm, int* dd );
                                                                       void
#define ED DB SERVER
                                  1023
#define ED_USER_CONNECT_DELAY_TIME
                                             1024
#define ED_DB_USER_ID
                                  1024
#define IDC MTS
                             1025
                                                                       spinlock.h
#define IDC TM MTS
                                1025
#define IDC_TM_TUXEDO
                                   1026
#define IDC_TM_NONE
                                 1027
                                                                                FILE: SPINLOCK.H
#define ED DB PASSWORD
                                    1028
#define ED_DB_NAME
                                 1029
                                                                        * Copyright 1997 Microsoft Corp., All rights reserved.
#define IDC TM ENCINA
                                  1030
                                                                       * Authors: Mike Parkes, Charles Levine, Philip Durr
// Next default values for new objects
                                                                                                    Microsoft Corp.
#ifdef APSTUDIO INVOKED
#ifndef APSTUDIO READONLY SYMBOLS
                                                                       #ifndef_INC_Spinlock
#define _APS_NEXT_RESOURCE_VALUE
                                           130
#define APS NEXT COMMAND VALUE
                                            40001
                                                                                const LONG LockClosed
#define APS_NEXT_CONTROL_VALUE
                                           1031
                                                                                const LONG LockOpen
                                                                                                                       = 0:
#define APS NEXT SYMED VALUE
                                         101
#endif
#endif
                                                                                   Spinlock and Semaphore locking.
Resource_tpcc_rc.h
                                                                                   This class provides a very conservative locking scheme.
                                                                                   The assumption behind the code is that locks will be
//{{NO DEPENDENCIES}}
                                                                                   held for a very short time. When a lock is taken a memory
// Microsoft Developer Studio generated include file.
                                                                                   location is exchanged. All other threads that want this
// Used by tpcc.rc
                                                                                   lock wait by spinning and sometimes sleeping on a semaphore
                                                                                   until it becomes free again. The only other choice is not
#define IDD DIALOG1
                                101
                                                                                   to wait at all and move on to do something else. This
                                                                                   module should normally be used in conjunction with cache
// Next default values for new objects
                                                                                    aligned memory in minimize cache line misses.
#ifdef APSTUDIO INVOKED
```

#ifndef APSTUDIO_READONLY_SYMBOLS

```
inline BOOL Spinlock::ClaimLock( BOOL Wait )
                                                                                                 if (! ClaimSpinlock( (volatile LONG*) & m Spinlock ))
          class Spinlock
                                                                                                            if (Wait)
                                                                                                                      WaitForLock();
                    // Private data.
                    HANDLE
                                            Semaphore;
                                                                                                            return Wait;
                     volatile LONG
                                            m_Spinlock;
                     volatile LONG
                                            Waiting;
                                                                                                 return TRUE;
                     #ifdef _DEBUG
                               // Counters for debugging builds.
                               volatile LONG
                                                    TotalLocks;
                                                    TotalSleeps;
                               volatile LONG
                               volatile LONG
                                                    TotalSpins;
                                                                                          Release the Spinlock.
                               volatile LONG
                                                    TotalWaits;
                     #endif
                                                                                          Release the lock and if needed wakeup any sleepers.
                    public:
                                                                                                    *************
                               // Public functions.
                               Spinlock( void );
                                                                                       inline void Spinlock::ReleaseLock( void )
                               inline BOOL ClaimLock( BOOL Wait =
                                                                                                 m Spinlock = LockOpen;
TRUE);
                                                                                                 if (Waiting > 0)
                               inline void ReleaseLock( void );
                                                                                                            WakeAllSleepers();
                               ~Spinlock( void );
                               // Disabled operations.
                               Spinlock( const Spinlock & Copy );
                                                                                       #define _INC_Spinlock
                               void operator=( const Spinlock & Copy );
                                                                            #endif
                    private:
                               // Private functions.
                               inline BOOL ClaimSpinlock(volatile LONG
                                                                             tpcc.cpp
*sl );
                               void WaitForLock( void );
                                                                                       FILE:
                                                                                                            TPCC.C
                               void WakeAllSleepers( void );
                                                                                                                      Microsoft TPC-C Kit Ver.
          };
                                                                            4.20.000
                                                                                                                      Copyright Microsoft, 1999
                                                                                      All Rights Reserved
                                                                                                                      Version 4.10.000 audited by
             A guaranteed atomic exchange.
                                                                            Richard Gimarc, Performance Metrics, 3/17/99
             An attempt is made to claim the Spinlock. This action is
                                                                                                            Main module for TPCC.DLL which is an
                                                                                       PURPOSE:
             guaranteed to be atomic.
                                                                            ISAPI service dll.
                                                                                       Contact:
                                                                                                 Charles Levine (clevine@microsoft.com)
                                                                             * Change history:
                                                                                                 4.20.000 - reworked error handling; added options for
          inline BOOL Spinlock::ClaimSpinlock( volatile LONG *Spinlock )
                                                                            COM and Encina txn monitors
                     #ifdef _DEBUG
                               InterlockedIncrement( (LPLONG) &
                                                                            #include <windows.h>
TotalLocks);
                                                                            #include cess.h>
                    #endif
                                                                            #include <tchar.h>
                    return ( ((*Spinlock) == LockOpen) &&
                                                                            #include <stdio.h>
(InterlockedExchange((LPLONG)Spinlock, LockClosed) == LockOpen));
                                                                            #include <stdarg.h>
                                                                            #include <malloc.h>
                                                                            #include <stdlib.h>
                                                                            #include <string.h>
                                                                            #include <time.h>
             Claim the Spinlock.
                                                                            #include <sys\timeb.h>
                                                                            #include <io.h>
             Claim the lock if available else wait or exit.
                                                                            #include <assert.h>
                                                                            #include <sqltypes.h>
```

```
#ifdef ICECAP
                                                                           CTxnLog
#include <icapexp.h>
                                                                                                                              *txnDelilog = NULL;
#endif
                                                                                     //used to log delivery transaction information
#include "..\..\common\src\trans.h"
                                                                                                                              hWorkerSemaphore =
                                                   //tpckit transaction
                                                                           HANDLE
header contains definations of structures specific to TPC-C
                                                                           INVALID HANDLE VALUE;
#include "..\..\common\src\error.h"
                                                                           HANDLE
                                                                                                                              hDoneEvent
#include "..\..\common\src\txn base.h"
                                                                                     = INVALID_HANDLE_VALUE;
#include "..\..\common\src\ReadRegistry.h"
                                                                           HANDLE
                                                                                                                              *pDeliHandles
                                                                           = NULL;
#include "..\..\common\txnlog\include\rtetime.h"
#include "..\..\common\txnlog\include\spinlock.h"
                                                                           // configuration settings from registry
#include "..\..\common\txnlog\include\txnlog.h"
                                                                           TPCCREGISTRYDATA
                                                                                                                    Reg;
// Database layer includes
                                                                           DWORD
#include "..\..\db dblib dll\src\tpcc dblib.h"
                                                   // DBLIB
implementation of TPC-C txns
                                                                           dwNumDeliveryThreads = 4;
#include "..\..\db_odbc_dll\src\tpcc_odbc.h"
                                                                           CRITICAL SECTION
                                                   // ODBC
                                                                                                                    DelBuffCriticalSection;
implementation of TPC-C txns
                                                                           //critical section for delivery transactions cache
                                                                           DELIVERY TRANSACTION *pDelBuff
                                                                                                                                         = NULL;
// Txn monitor layer includes
                                                                           DWORD
                                                                                                                              dwDelBuffSize
#include "..\..\tm com dll\src\tpcc com.h"
                                                             // COM
                                                                           = 100;
                                                                                               // size of circular buffer for delivery txns
                                                                           DWORD
Services implementation on TPC-C txns
                                                                                                                              dwDelBuffFreeCount;
#include "..\..\tm tuxedo dll\src\tpcc tux.h"
                                                   // interface to Tuxedo
                                                                                                          // number of buffers free
libraries
                                                                           DWORD
                                                                                                                              dwDelBuffBusyIndex =
#include "..\..\tm_encina_dll\src\tpcc_enc.h"
                                                                                               // index position of entry waiting to be delivered
                                                   // interface to Encina
                                                                           0.
libraries
                                                                           DWORD
                                                                                                                              dwDelBuffFreeIndex =
                                                                                               // index position of unused entry
                                                                           0:
#include "httpext.h"
                                                   //ISAPI DLL
information header
                                                                           #include "..\..\common\src\ReadRegistry.cpp"
#include "tpcc.h"
                                                             //this dlls
specific structure, value e.t. header.
                                                                           /* FUNCTION: DllMain
                                                                            * PURPOSE:
                                                                                                This function is the entry point for the DLL. This
#define
         LEN_ERR_STRING 256
                                                                           implementation is based on the
                                                                                                          fact that DLL PROCESS ATTACH is only
// defines for Make<Txn>Form calls to distinguish input and output flavors
                                                                           called from the inet service once.
         OUTPUT FORM
#define
                                                                            * ARGUMENTS:
#define INPUT_FORM
                                                                                              HANDLE hModule
                                                                           module handle
                                                                                                                    DWORD ul reason for call
szMyComputerName[MAX_COMPUTERNAME_LENGTH+1];
                                                                           reason for call
                                                                                                                    LPVOID lpReserved
//Terminal client id structure
                                                                           reserved for future use
         Term = \{0, 0, 0, \text{NULL}\};
TERM
                                                                            * RETURNS:
                                                                                                          BOOL
                                                                                                                    FALSE
// The WEBCLIENT_VERSION string specifies the version level of this web
                                                                           errors occured in initialization
                                                                                                                                         TRUE
client interface.
// The RTE must be synchronized with the interface level on login, otherwise
                                                                                               DLL successfully initialized
the login
// will fail. This is a sanity check to catch problems resulting from mismatched
                                                                           BOOL APIENTRY DllMain(HANDLE hModule, DWORD ul reason for call,
versions
// of the RTE and web client.
                                                                           LPVOID lpReserved)
#define WEBCLIENT VERSION "410"
                                                                                     DWORD i;
                                                                                     char szEvent[LEN ERR STRING] = "\0";
          CRITICAL_SECTION
                                                   TermCriticalSection;
static
                                                                                     char szLogFile[128];
static HINSTANCE hLibInstanceTm = NULL;
                                                                                     char szDllName[128];
static HINSTANCE hLibInstanceDb = NULL;
                                                                           // debugging...
TYPE CTPCC DBLIB
                               *pCTPCC DBLIB new;
                                                                           // DebugBreak();
TYPE_CTPCC_ODBC
                                         *pCTPCC ODBC new;
TYPE_CTPCC_TUXEDO *pCTPCC_TUXEDO TYPE_CTPCC_ENCINA *pCTPCC_ENCINA_new;
                              *pCTPCC_TUXEDO_new;
                                                                                     try
TYPE_CTPCC_ENCINA *pCTPCC_ENCINA_post_init;
                                                                                               switch(ul_reason_for_call)
TYPE CTPCC COM
                              *pCTPCC COM new;
                                                                                                          case DLL_PROCESS_ATTACH:
// For deferred Delivery txns:
                                                                                                                    {
```

```
DWORD dwSize =
MAX COMPUTERNAME LENGTH+1;
                                                                      pCTPCC ENCINA new = (TYPE CTPCC ENCINA*)
                                                                      GetProcAddress(hLibInstanceTm, "CTPCC ENCINA new");
GetComputerName(szMyComputerName, &dwSize);
                                                                      pCTPCC_ENCINA_post init = (TYPE CTPCC ENCINA*)
szMyComputerName[dwSize] = 0;
                                                                      GetProcAddress(hLibInstanceTm,"CTPCC ENCINA post init");
                                      }
                                                                      (pCTPCC ENCINA new == NULL)
                                                                                                                                throw new
                                                                      CWEBCLNT ERR( ERR GETPROCADDR FAILED, szDllName,
DisableThreadLibraryCalls((HMODULE)hModule);
                                                                      GetLastError() );
InitializeCriticalSection(&TermCriticalSection);
                                                                                                             else if (Reg.eTxnMon == COM)
                                      if (ReadTPCCRegistrySettings(
&Reg))
                                                                                                                      strcpy( szDllName,
                                                throw new
                                                                      Reg.szPath);
CWEBCLNT ERR( ERR MISSING REGISTRY ENTRIES );
                                                                                                                      strcat( szDllName,
                                                                      "tpcc_com.dll");
                                                                                                                      hLibInstanceTm =
                                      dwDelBuffSize = min(
Reg.dwMaxPendingDeliveries, 10000); // min with 10000 as a sanity
                                                                      LoadLibrary( szDllName );
                                                                                                                      if (hLibInstanceTm ==
constraint
                                      dwNumDeliveryThreads = min(
                                                                      NULL)
Reg.dwNumberOfDeliveryThreads, 100); // min with 100 as a sanity constraint
                                                                                                                                throw new
                                                                      CWEBCLNT ERR( ERR LOADDLL FAILED, szDllName, GetLastError() );
                                      TermInit();
                                                                                                                      // get function pointer
                                      // load DLL for txn monitor
                                                                      to wrapper for class constructor
                                      if(Reg.eTxnMon == TUXEDO)
                                                                                                                      pCTPCC_COM_new =
                                                                      (TYPE CTPCC COM*)
                                                                      GetProcAddress(hLibInstanceTm,"CTPCC_COM_new");
                                                strcpy( szDllName,
Reg.szPath);
                                                strcat( szDllName,
                                                                      (pCTPCC_COM_new == NULL)
"tpcc tuxedo.dll");
                                                                                                                                throw new
                                                hLibInstanceTm =
                                                                      CWEBCLNT ERR( ERR GETPROCADDR FAILED, szDllName,
LoadLibrary( szDllName );
                                                                      GetLastError() );
                                                if (hLibInstanceTm ==
NULL)
                                                                                                             // load DLL for database
                                                         throw new
CWEBCLNT_ERR( ERR_LOADDLL_FAILED, szDllName, GetLastError() );
                                                                      connection
                                                                                                             if((Reg.eTxnMon == None) \parallel
                                                // get function pointer
                                                                      (dwNumDeliveryThreads > 0))
to wrapper for class constructor
                                                                                                                      if (Reg.eDB Protocol
pCTPCC_TUXEDO_new = (TYPE_CTPCC TUXEDO*)
                                                                      == DBLIB)
GetProcAddress(hLibInstanceTm, "CTPCC TUXEDO new");
                                                                                                                                strcpy(
(pCTPCC TUXEDO new == NULL)
                                                                      szDllName, Reg.szPath );
                                                         throw new
                                                                                                                                strcat(
CWEBCLNT_ERR( ERR_GETPROCADDR_FAILED, szDllName,
                                                                      szDllName, "tpcc_dblib.dll");
GetLastError());
                                                                      hLibInstanceDb = LoadLibrary( szDllName );
                                      else if (Reg.eTxnMon ==
                                                                                                                                if
ENCINA)
                                                                      (hLibInstanceDb == NULL)
                                                                      throw new CWEBCLNT ERR( ERR LOADDLL FAILED, szDllName,
                                                strcpy( szDllName,
Reg.szPath);
                                                                      GetLastError() );
                                                strcat( szDllName,
"tpcc_encina.dll");
                                                                                                                                // get
                                                hLibInstanceTm =
                                                                      function pointer to wrapper for class constructor
LoadLibrary( szDllName );
                                                if (hLibInstanceTm ==
                                                                      pCTPCC DBLIB new = (TYPE CTPCC DBLIB*)
NULL)
                                                                      GetProcAddress(hLibInstanceDb,"CTPCC DBLIB new");
                                                                                                                                if
                                                         throw new
CWEBCLNT_ERR( ERR_LOADDLL_FAILED, szDllName, GetLastError() );
                                                                      (pCTPCC_DBLIB_new == NULL)
                                                // get function pointer
to wrapper for class constructor
                                                                      throw new CWEBCLNT ERR( ERR GETPROCADDR FAILED, szDllName,
                                                                      GetLastError());
                                                                                                                      }
```

```
pDelBuff = new
                                                  else if
                                                                         DELIVERY TRANSACTION[dwDelBuffSize];
(Reg.eDB Protocol == ODBC)
                                                                                                                           // launch
                                                            strcpy(
                                                                         DeliveryWorkerThread to perform actual delivery txns
szDllName, Reg.szPath );
                                                                                                                           for(i=0;
                                                                         i<dwNumDeliveryThreads; i++)
                                                            strcat(
szDllName, "tpcc odbc.dll");
                                                                                                                           {
hLibInstanceDb = LoadLibrary( szDllName );
                                                                         pDeliHandles[i] = (HANDLE) _beginthread( DeliveryWorkerThread, 0, NULL
                                                            if
(hLibInstanceDb == NULL)
                                                                                                                                     if
                                                                         (pDeliHandles[i] == INVALID_HANDLE_VALUE)
throw new CWEBCLNT ERR( ERR LOADDLL FAILED, szDllName,
                                                                         throw new CWEBCLNT_ERR( ERR_DELIVERY_THREAD_FAILED );
GetLastError());
                                                            // get
function pointer to wrapper for class constructor
                                                                                                                 break;
pCTPCC ODBC new = (TYPE CTPCC ODBC*)
GetProcAddress(hLibInstanceDb,"CTPCC ODBC new");
                                                                                                       case DLL PROCESS DETACH:
                                                                                                                 if (dwNumDeliveryThreads)
(pCTPCC ODBC new == NULL)
                                                                                                                           if (txnDelilog !=
throw new CWEBCLNT ERR( ERR GETPROCADDR FAILED, szDllName,
                                                                         NULL)
GetLastError());
                                                                                                                                     //write event
                                                                         into txn log for STOP
                                        if (dwNumDeliveryThreads)
                                                                         txnDelilog->WriteCtrlRecToLog(TXN EVENT STOP, szMyComputerName,
                                                                         sizeof(szMyComputerName));
                                                  // for deferred delivery
txns:
                                                                                                                                     // This will
                                                  hDoneEvent =
                                                                         do a clean shutdown of the delivery log file
CreateEvent( NULL, TRUE /* manual reset */, FALSE /* initially not signalled
                                                                                                                                     CTxnLog
*/, NULL);
                                                                         *txnDelilogLocal = txnDelilog;
                                                                                                                                     txnDelilog=
                                                                         NULL;
InitializeCriticalSection(&DelBuffCriticalSection);
                                                  hWorkerSemaphore =
                                                                                                                                     delete
CreateSemaphore( NULL, 0, dwDelBuffSize, NULL );
                                                                         txnDelilogLocal;
                                                  dwDelBuffFreeCount =
dwDelBuffSize;
                                                                                                                           delete [] pDeliHandles;
                                                  InitJulianTime(NULL);
                                                                                                                           delete [] pDelBuff;
                                                                                                                           CloseHandle(
                                                  // create unique log file
name based on delilog-yymmdd-hhmm.log
                                                                         hWorkerSemaphore );
                                                  SYSTEMTIME Time;
                                                                                                                           CloseHandle(
                                                  GetLocalTime( &Time
                                                                         hDoneEvent);
);
                                                                         DeleteCriticalSection(&DelBuffCriticalSection);
                                                  wsprintf( szLogFile,
"%sdelivery-%2.2d%2.2d%2.2d-%2.2d%2.2d.log",
Reg.szPath, Time.wYear % 100, Time.wMonth, Time.wDay, Time.wHour,
Time.wMinute );
                                                                         DeleteCriticalSection(&TermCriticalSection);
                                                  txnDelilog = new
CTxnLog(szLogFile, TXN_LOG_WRITE);
                                                                                                                 if (hLibInstanceTm != NULL)
                                                                                                                           FreeLibrary(
                                                  //write event into txn
                                                                         hLibInstanceTm );
log for START
                                                                                                                 hLibInstanceTm = NULL;
txnDelilog->WriteCtrlRecToLog(TXN EVENT START,
                                                                                                                 if (hLibInstanceDb != NULL)
szMyComputerName, sizeof(szMyComputerName));
                                                                                                                           FreeLibrary(
                                                                         hLibInstanceDb );
                                                  // allocate structures for
                                                                                                                 hLibInstanceDb = NULL;
delivery buffers and thread mgmt
                                                  pDeliHandles = new
                                                                                                                 Sleep(500);
HANDLE[dwNumDeliveryThreads];
                                                                                                                 break;
```

```
default:
                                        /* nothing */;
                    }
          catch (CBaseErr *e)
                    WriteMessageToEventLog( e->ErrorText() );
                    delete e;
                    TerminateExtension(0);
                   return FALSE;
          catch (...)
                    WriteMessageToEventLog(TEXT("Unhandled
exception. DLL could not load."));
                    TerminateExtension(0);
                   return FALSE:
          return TRUE;
}
/* FUNCTION: GetExtensionVersion
* PURPOSE:
                   This function is called by the inet service when the DLL
is first loaded.
* ARGUMENTS: HSE VERSION INFO
                                                  *pVer
                                                            passed in
structure in which to place expected version number.
* RETURNS:
                              TRUE
                                        inet service expected return value.
BOOL WINAPI GetExtensionVersion(HSE_VERSION_INFO *pVer)
          pVer->dwExtensionVersion =
MAKELONG(HSE VERSION MINOR, HSE VERSION MAJOR);
          lstrcpyn(pVer->lpszExtensionDesc, "TPC-C Server.",
HSE_MAX_EXT_DLL_NAME_LEN);
          // TODO: why do we need this here instead of in the DLL attach?
          if(Reg.eTxnMon == ENCINA)
                   pCTPCC_ENCINA_post_init();
          return TRUE;
}
/* FUNCTION: TerminateExtension
* PURPOSE:
                   This function is called by the inet service when the DLL
is about to be unloaded.
                              Release all resources in anticipation of being
unloaded.
* RETURNS:
                              TRUE
                                        inet service expected return value.
BOOL WINAPI TerminateExtension( DWORD dwFlags )
          if (pDeliHandles)
                    SetEvent( hDoneEvent );
                    for(DWORD i=0; i<dwNumDeliveryThreads; i++)
                              WaitForSingleObject( pDeliHandles[i],
INFINITE);
```

```
TermDeleteAll();
          return TRUE;
/* FUNCTION: HttpExtensionProc
* PURPOSE:
                    This function is the main entry point for the TPCC DLL.
The internet service
                              calls this function passing in the http string.
* ARGUMENTS:
                   EXTENSION CONTROL BLOCK
                                                            *pECB
structure pointer to passed in internet
                                        service information.
* RETURNS:
                              DWORD HSE STATUS SUCCESS
                              connection can be dropped if error
HSE STATUS SUCCESS AND KEEP CONN
                                                  keep connect valid
comment sent
* COMMENTS:
                    None
DWORD WINAPI HttpExtensionProc(EXTENSION CONTROL BLOCK
*pECB)
          int
                                        iCmd, FormId, TermId, iSyncId;
          char
                              szBuffer[4096];
                                        lpbSize:
          int
          static char szHeader[] = "200 Ok";
          DWORD
                              dwSize = 6;
                                                            // initial
value is strlen(szHeader)
          char
                              szHeader1[4096];
#ifdef ICECAP
          StartCAP();
#endif
          try
                    //process http query
                    ProcessQueryString(pECB, &iCmd, &FormId, &TermId,
&iSyncId);
                   if (TermId != 0)
                              if ( TermId < 0 || TermId >=
Term.iNumEntries || Term.pClientData[TermId].iNextFree != -1 )
                                        // debugging...
                                        char szTmp[128];
                                        wsprintf( szTmp, "Invalid term ID;
TermId = \%d", TermId);
                                        WriteMessageToEventLog( szTmp
                                        throw new CWEBCLNT ERR(
ERR_INVALID_TERMID );
                              //must have a valid syncid here since termid is
valid
```

```
if (iSyncId !=
                                                                                              case 6:
Term.pClientData[TermId].iSyncId)
                                                                                                        // stock-level selected from menu; display
                                        throw new CWEBCLNT ERR(
                                                                          stock-level input form
ERR_INVALID_SYNC_CONNECTION );
                                                                                                        MakeStockLevelForm(TermId, NULL,
                                                                          INPUT FORM, szBuffer);
                                                                                                        break;
                              Term.pClientData[TermId].iTickCount =
                                                                                              case 7:
GetTickCount();
                                                                                                        // ExitCmd
                                                                                                        TermDelete(TermId);
                                                                                                        WelcomeForm(pECB, szBuffer);
                                                                                                        break;
                    switch(iCmd)
                                                                                              case 8:
                                                                                                        SubmitCmd(pECB, szBuffer);
                    case 0:
                                                                                                        break:
                              WelcomeForm(pECB, szBuffer);
                                                                                              case 9:
                              break;
                                                                                                        // menu
                    case 1:
                                                                                                        MakeMainMenuForm(TermId,
                                                                          Term.pClientData[TermId].iSyncId, szBuffer);
                              switch(FormId)
                                                                                                        break:
                                        case WELCOME FORM:
                                                                                              case 10:
                                        case MAIN_MENU_FORM:
                                                                                                        // CMD=Clear
                                                  break;
                                                                                                        // resets all connections; should only be used
                                        case NEW_ORDER_FORM:
                                                                          when no other connections are active
                                                                                                        TermDeleteAll();
ProcessNewOrderForm(pECB, TermId, szBuffer);
                                                                                                        TermInit();
                                                                                                        WelcomeForm(pECB, szBuffer);
                                                  break:
                                        case PAYMENT FORM:
                                                                                                        break;
                                                                                                       // CMD=Stats
                                                                                              case 11:
ProcessPaymentForm(pECB, TermId, szBuffer);
                                                                                                        StatsCmd(pECB, szBuffer);
                                                  break:
                                                                                                        break:
                                        case DELIVERY FORM:
ProcessDeliveryForm(pECB, TermId, szBuffer);
                                                                                    catch (CBaseErr *e)
                                                  break;
                                        case ORDER_STATUS_FORM:
                                                                                              ErrorForm( pECB, e->ErrorType(), e->ErrorNum(),
                                                                          TermId, iSyncId, e->ErrorText(), szBuffer );
ProcessOrderStatusForm(pECB, TermId, szBuffer);
                                                                                              delete e;
                                                  break;
                                        case STOCK_LEVEL_FORM:
                                                                                    catch (...)
                                                                                    {
                                                                                              ErrorForm( pECB, ERR TYPE WEBDLL, 0, TermId,
ProcessStockLevelForm(pECB, TermId, szBuffer);
                                                                          iSyncId, "Error: Unhandled exception in Web Client.", szBuffer );
                                                  break:
                              break:
                                                                          #ifdef ICECAP
                    case 2:
                              // new-order selected from menu; display
                                                                                    StopCAP();
new-order input form
                                                                          #endif
                              MakeNewOrderForm(TermId, NULL,
INPUT_FORM, szBuffer);
                                                                                    lpbSize = strlen(szBuffer);
                              break;
                   case 3:
                                                                                    wsprintf(szHeader1,
                                                                                                         'Content-Type: text/html\r\n"
                              // payment selected from menu; display
payment input form
                                                                                                        "Content-Length: %d\r\n"
                              MakePaymentForm(TermId, NULL,
                                                                                                        "Connection: Keep-Alive\r\n\r\n", lpbSize);
INPUT FORM, szBuffer);
                                                                                    strcat( szHeader1, szBuffer );
                              break;
                                                                                   (*pECB->ServerSupportFunction)(pECB->ConnID,
                   case 4:
                                                                          HSE REQ_SEND_RESPONSE_HEADER, szHeader, (LPDWORD) &dwSize,
                              // delivery selected from menu; display
                                                                          (LPDWORD)szHeader1);
delivery input form
                              MakeDeliveryForm(TermId, NULL,
INPUT FORM, szBuffer);
                                                                                    //finish up and keep connection
                              break;
                                                                                    pECB->dwHttpStatusCode = 200;
                   case 5:
                                                                                    return HSE_STATUS_SUCCESS_AND_KEEP_CONN;
                              // order-status selected from menu; display
order-status input form
                              MakeOrderStatusForm(TermId, NULL,
INPUT_FORM, szBuffer);
                                                                          void WriteMessageToEventLog(LPTSTR lpszMsg)
                              break;
```

```
TCHAR szMsg[256];
                                                                                                          pTxn = pCTPCC ODBC new(
  HANDLE hEventSource;
                                                                           Reg.szDbServer, Reg.szDbUser, Reg.szDbPassword, szMyComputerName,
  LPTSTR lpszStrings[2];
                                                                           Reg.szDbName);
                                                                                                else if (Reg.eDB_Protocol == DBLIB)
                                                                                                          pTxn = pCTPCC_DBLIB new(
  // Use event logging to log the error.
                                                                           Reg.szDbServer, Reg.szDbUser, Reg.szDbPassword, szMyComputerName,
  hEventSource = RegisterEventSource(NULL, TEXT("TPCC.DLL"));
                                                                           Reg.szDbName);
                                                                                                pDeliveryData = pTxn->BuffAddr Delivery();
   stprintf(szMsg, TEXT("Error in TPCC.DLL: "));
  lpszStrings[0] = szMsg;
                                                                                      catch (CBaseErr *e)
  lpszStrings[1] = lpszMsg;
                                                                                                char szTmp[1024];
  if (hEventSource != NULL)
                                                                                                wsprintf( szTmp, "Error in Delivery Txn thread. Could
                                                                           not connect to database.
    ReportEvent(hEventSource, // handle of event source
                                                                                                                     "%s. Server=%s, User=%s,
      EVENTLOG_ERROR_TYPE, // event type
                                                                           Password=%s, Database=%s",
      0,
                    // event category
                                                                                                                     e->ErrorText(), Reg.szDbServer,
      0,
                    // event ID
                                                                            Reg.szDbUser, Reg.szDbPassword, Reg.szDbName);
      NULL,
                       // current user's SID
                                                                                                WriteMessageToEventLog( szTmp );
      2,
                    // strings in lpszStrings
                                                                                                delete e;
      0
                    // no bytes of raw data
                                                                                                goto ErrorExit;
      (LPCTSTR *)lpszStrings, // array of error strings
      NULL);
                       // no raw data
                                                                                      catch (...)
    (VOID) DeregisterEventSource(hEventSource);
                                                                                                WriteMessageToEventLog(TEXT("Unhandled exception
                                                                           caught in DeliveryWorkerThread."));
                                                                                                goto ErrorExit;
/* FUNCTION: DeliveryWorkerThread
                                                                                      while (TRUE)
* PURPOSE:
                    This function processes deferred delivery txns. There are
                                                                                                try
typically several
                              threads running this routine. The number of
                                                                                                          //while delivery thread running, i.e. user has
threads is determined by an entry
                                                                           not requested termination
                              read from the registry. The thread waits for
                                                                                                          while (TRUE)
work by waiting on semaphore.
                              When a delivery txn is posted, the semaphore
                                                                                                                     // need to wait for multiple objects:
is released. After processing
                                                                           program exit or worker semaphore;
                                                                                                                     handles[0] = hDoneEvent;
                              the delivery txn, information is logged to
record the txn status and execution
                                                                                                                     handles[1] = hWorkerSemaphore;
                                                                                                                     index = WaitForMultipleObjects(
                              time
*/
                                                                           2, &handles[0], FALSE, INFINITE);
                                                                                                                     if (index == WAIT OBJECT 0)
                                                                                                                               goto ErrorExit;
/*static*/ void DeliveryWorkerThread(void *ptr)
          CTPCC BASE
                                         *pTxn = NULL;
                                                                                                                     ZeroMemory(&txnDeliRec,
                                                                           sizeof(txnDeliRec));
          DELIVERY_TRANSACTION
                                                   delivery;
                                                                                                                     txnDeliRec.TxnType =
          PDELIVERY_DATA
                                                                            TXN REC TYPE TPCC DELIV DEF;
pDeliveryData;
          TXN RECORD TPCC DELIV DEF
                                                   txnDeliRec;
                                                                                                                     // make a local copy of current
                                                                           entry from delivery buffer and increment buffer index
          DWORD
                                                             index;
          HANDLE
                                                             handles[2];
                                                                           EnterCriticalSection(&DelBuffCriticalSection);
                                                                                                                     delivery =
          SYSTEMTIME
                                                                            *(pDelBuff+dwDelBuffBusyIndex);
                                         trans end;
                                                             //delivery
transaction finished time
                                                                                                                     dwDelBuffFreeCount++;
                                                                                                                     dwDelBuffBusyIndex++;
          SYSTEMTIME
                                         trans start;//delivery transaction
start time
                                                                                                                     if (dwDelBuffBusyIndex ==
                                                                           dwDelBuffSize)
                                                                                                // wrap-around if at end of buffer
          assert(txnDelilog != NULL);
                                                                                                                               dwDelBuffBusyIndex =
                                                                           0;
          try
                    if (Reg.eDB Protocol == ODBC)
                                                                           LeaveCriticalSection(&DelBuffCriticalSection);
```

```
* RETURNS:
                                          pDeliveryData->w id =
                                                                                                              BOOL
                                                                                                                        FALSE
                                                                                                                                   delivery information
delivery.w_id;
                                                                              posted successfully
                                                                                                                                              TRUE
                                          pDeliveryData->o carrier id =
delivery.o_carrier_id;
                                                                              error cannot post delivery info
                                          txnDeliRec.w id =
pDeliveryData->w id;
                                                                              BOOL PostDeliveryInfo(short w id, short o carrier id)
                                          txnDeliRec.o carrier id =
pDeliveryData->o carrier id;
                                                                                         BOOL bError;
                                          txnDeliRec.TxnStartT0 =
Get64BitTime(&delivery.queue);
                                                                                         EnterCriticalSection(&DelBuffCriticalSection);
                                                                                         if (dwDelBuffFreeCount > 0)
                                          GetLocalTime( &trans start );
                                                                                                   bError = FALSE;
                                          pTxn->Delivery();
                                          GetLocalTime( &trans_end );
                                                                                                   (pDelBuff+dwDelBuffFreeIndex)->w_id
                                                                              = w_id;
                                          //log txn
                                                                                                   (pDelBuff+dwDelBuffFreeIndex)->o carrier id
                                          txnDeliRec.TxnStatus =
                                                                              o_carrier_id;
ERR_SUCCESS;
                                          for (int i=0; i<10; i++)
                                                                              GetLocalTime(&(pDelBuff+dwDelBuffFreeIndex)->queue);
                                                     txnDeliRec.o_id[i] =
                                                                                                   dwDelBuffFreeCount--;
pDeliveryData->o id[i];
                                          txnDeliRec.DeltaT4 =
                                                                                                   dwDelBuffFreeIndex++;
(int)(Get64BitTime(&trans_end) - txnDeliRec.TxnStartT0);
                                                                                                   if (dwDelBuffFreeIndex == dwDelBuffSize)
                                          txnDeliRec.DeltaTxnExec =
                                                                                                              dwDelBuffFreeIndex = 0;
                                                                                                                                                         //
                                                                              wrap-around if at end of buffer
(int)(Get64BitTime(&trans_end) - Get64BitTime(&trans_start));
                                          if (txnDelilog != NULL)
                                                                                        else
                                                                                                   // No free buffers. Return an error, which indicates that
                                                                              the delivery buffer is full.
txnDelilog->WriteToLog(&txnDeliRec);
                                                                                                   // Most likely, the number of delivery worker threads
                                                                              needs to be increased to keep up
                     catch (CBaseErr *e)
                                                                                                   // with the txn rate.
                                                                                                   bError = TRUE;
                                                                                         LeaveCriticalSection(&DelBuffCriticalSection);
                               char szTmp[1024];
                               wsprintf( szTmp, "Error in Delivery Txn
                                                                                         if (!bError)
thread. %s", e->ErrorText());
                               WriteMessageToEventLog( szTmp );
                                                                                                   // increment worker semaphore to wake up a worker
                                                                              thread
                                                                                                   ReleaseSemaphore( hWorkerSemaphore, 1, NULL );
                               // log the error txn
                               txnDeliRec.TxnStatus = e->ErrorType();
                               if (txnDelilog != NULL)
                                                                                         return bError;
txnDelilog->WriteToLog(&txnDeliRec);
                                                                              /* FUNCTION: ProcessQueryString
                               delete e;
                                                                               * PURPOSE:
                                                                                                   This function extracts the relevent information out of the
                     catch (...)
                                                                              http command passed in from
                                                                                                              the browser.
                               // unhandled exception; shouldn't happen; not
                                                                               * COMMENTS:
much we can do ...
                                                                                                   If this is the initial connection i.e. client is at welcome
                                                                              screen then
WriteMessageToEventLog(TEXT("Unhandled exception caught in
                                                                                                                         there will not be a terminal id or
DeliveryWorkerThread."));
                                                                              current form id. If this is the case
                                                                                                                         then the pTermid and pFormid
                                                                              return values are undefined.
ErrorExit:
                                                                              void ProcessQueryString(EXTENSION CONTROL BLOCK *pECB, int
          delete pTxn;
          _endthread();
                                                                              *pCmd, int *pFormId, int *pTermId, int *pSyncId)
                                                                                         char *ptr = pECB->lpszQueryString;
/* FUNCTION: PostDeliveryInfo
                                                                                         char szBuffer[25];
* PURPOSE:
                     This function enters the delivery txn into the deferred
delivery buffer.
                                                                                         //allowable client command strings i.e. CMD=command
                                                                                         static char *szCmds[] =
```

```
"<INPUT
                    "Process", "..NewOrder..", "..Payment..", "..Delivery..",
"..Order-Status..", "..Stock-Level..",
                                                                        TYPE=\"hidden\" NAME=\"FORMID\" VALUE=\"1\">"
                    "..Exit..", "Submit", "Menu", "Clear", "Stats", ""
                                                                                                                                    "<INPUT
                                                                         TYPE=\"hidden\" NAME=\"TERMID\" VALUE=\"0\">"
          };
                                                                                                                                    "<INPUT
          *pCmd
                                       // default is the login screen
                                                                         TYPE=\"hidden\" NAME=\"SYNCID\" VALUE=\"0\">"
                                                                                                                                    "<INPUT
          *pTermId = 0;
                                                                         TYPE=\"hidden\" NAME=\"VERSION\" VALUE=\""
          // if no params (i.e., empty query string), then return login screen
                                                                        WEBCLIENT_VERSION "\">"
          if (strlen(pECB->lpszQueryString) == 0)
                   return;
                                                                                  sprintf( szTmp,
                                                                                                      "Configuration Settings: <BR><font
          // parse FORMID, TERMID, and SYNCID
                                                                        face=\"Courier New\" color=\"blue\"><PRE>"
          *pFormId = GetIntKeyValue(&ptr, "FORMID", NO_ERR,
                                                                                                                          "Txn Monitor
                                                                         <B>%s</B><BR>"
NO_ERR);
          *pTermId = GetIntKeyValue(&ptr, "TERMID", NO ERR,
                                                                                                                          "Database protocol
NO_ERR);
                                                                         = <B>%s</B><BR>"
          *pSyncId = GetIntKeyValue(&ptr, "SYNCID", NO_ERR,
                                                                                                                          "Max Connections
NO_ERR);
                                                                         = <B>%d</B><BR>"
                                                                                                                          "# of Delivery Threads
          // parse CMD
                                                                         = <B>%d</B><BR>"
          GetKeyValue(&ptr, "CMD", szBuffer, sizeof(szBuffer),
                                                                                                                          "Max Pending
ERR_COMMAND_UNDEFINED);
                                                                        Deliveries = <B>\%d</B><BR>"
                                                                                                      , szTxnMonNames[Reg.eTxnMon].
          // see which command it matches
                                                                        szDBNames[Reg.eDB_Protocol],
          for(i=0;; i++)
                                                                                                      Reg.dwMaxConnections,
                                                                        dwNumDeliveryThreads, dwDelBuffSize );
          {
                   if (szCmds[i][0] == 0)
                                                                                  strcat( szBuffer, szTmp);
                             // no more; no match; return error
                                                                                  if(Reg.eTxnMon == COM)
                             throw new CWEBCLNT ERR(
ERR COMMAND UNDEFINED);
                   if ( !strcmp(szCmds[i], szBuffer) )
                                                                                            sprintf( szTmp,
                                                                                                                "COM Single Pool
                                                                         <B>%s</B><BR>"
                              *pCmd = i+1;
                                                                                                      Reg.bCOM SinglePool? "YES": "NO");
                             break;
                                                                                            strcat( szBuffer, szTmp);
                                                                                  strcat( szBuffer, "</PRE></font>");
                                                                                  if(Reg.eTxnMon == None)
/* FUNCTION: void WelcomeForm
                                                                                            // connection options may be specified when not using a
                                                                        txn monitor
                                                                                                                "Please enter your database options
                                                                                            sprintf( szTmp,
                                                                        for this connection: <BR>"
void WelcomeForm(EXTENSION CONTROL BLOCK *pECB, char
                                                                                                                                    "<font
*szBuffer)
                                                                        face=\"Courier New\" color=\"blue\"><PRE>"
                                                                                                                                    "DB Server
          char szTmp[1024];
                                                                         = <INPUT NAME=\"db server\" SIZE=20 VALUE=\"%s\"><BR>'
                                                                                                                                    "DB User
                                                                        ID = <INPUT NAME = \"db\_user\" SIZE = 20 VALUE = \"\%s\"> <BR>"
          //welcome to tpc-c html form buffer, this is first form client sees.
          strcpy( szBuffer,
                             "<HTML><HEAD><TITLE>TPC-C Web
                                                                                                                                    "DB
Client</TITLE></HEAD><BODY>"
                                                                        Password = <INPUT NAME=\"db_passwd\" SIZE=20
                                                                         VALUE=\"%s\"><BR>"
"<B><BIG>Microsoft TPC-C Web Client (ver 4.20)</BIG></B> <BR> <BR>'
                                                                                                                                    "DB Name
                                                            '<font
                                                                         = <INPUT NAME=\"db name\" SIZE=20 VALUE=\"%s\"><BR>"
face=\"Courier New\"><PRE>"
                                                           "Compiled:
                                                                         "</PRE></font>"
" DATE ", " TIME " <BR>"
                                                                                                                 , Reg.szDbServer, Reg.szDbUser,
                                                           "Source:
                                                                        Reg.szDbPassword, Reg.szDbName);
  FILE "(" TIMESTAMP ") <BR>"
                                                                                  else
                                                                                            // if using a txn monitor, connection options are
"</PRE></font>"
                                                                        determined from registry; can't
                                                           "<FORM
                                                                                            // set per user. show options fyi
ACTION=\"tpcc.dll\" METHOD=\"GET\">"
                                                                                                                "Database options which will be
                                                                                            sprintf( szTmp,
                                                           "<INPUT
                                                                         used by the transaction monitor: <BR>"
                                                                                                                                    "<font
TYPE=\"hidden\" NAME=\"STATUSID\" VALUE=\"0\">"
                                                           "<INPUT
                                                                        face=\"Courier New\" color=\"blue\"><PRE>"
TYPE=\"hidden\" NAME=\"ERROR\" VALUE=\"0\">"
                                                                                                                                    "DB Server
                                                                               = <B>%s</B><BR>"
```

```
"DB User
                                                                                             GetKeyValue(&ptr, "db name", szDatabase,
ID
         =<B>%s</B><BR>"
                                                                         sizeof(szDatabase), NO ERR);
                                                            "DB
Password
               = <B>%s</B><BR>"
                                                            "DB Name
                                                                                   // parse warehouse ID
      = <B>%S</B><BR>"
                                                                                   int w id = GetIntKeyValue(&ptr, "w id",
                                                                         ERR_HTML_ILL_FORMED, ERR_W_ID_INVALID);
"</PRE></font>"
                                                                                   if (w_id < 1)
                                                                                             throw new CWEBCLNT ERR( ERR W ID INVALID
                                        , Reg.szDbServer, Reg.szDbUser,
Reg.szDbPassword, Reg.szDbName);
                                                                         );
         strcat( szBuffer, szTmp);
                                                                                   // parse district ID
         sprintf( szTmp,
                              "Please enter your Warehouse and District for
                                                                                   int d id = GetIntKeyValue(&ptr, "d id",
                                                                         ERR_HTML_ILL_FORMED, ERR_D_ID_INVALID);
this session: <BR>'
                                                                                   if (d_id < 1 || d_id > 10)
                                                 "<font face=\"Courier
                                                                                             throw new CWEBCLNT ERR( ERR D ID INVALID
New\" color=\"blue\"><PRE>" );
         strcat( szBuffer, szTmp);
         streat( szBuffer,
                              "Warehouse ID = <INPUT NAME=\"w_id\"
SIZE=4><BR>"
                                                            "District ID
                                                                                   iNewTerm = TermAdd();
= <INPUT NAME=\"d_id\" SIZE=2><BR>"
                                                                                   Term.pClientData[iNewTerm].w id = w id;
"</PRE></font><HR>"
                                                                                   Term.pClientData[iNewTerm].d id = d id;
                                                            "<INPUT
TYPE=\"submit\" NAME=\"CMD\" VALUE=\"Submit\">"
                                                                                   try
                                                                                   {
"</FORM></BODY></HTML>");
                                                                                             if(Reg.eTxnMon == TUXEDO)
                                                                                                       Term.pClientData[iNewTerm].pTxn =
                                                                         pCTPCC TUXEDO new();
                                                                                             else if (Reg.eTxnMon == ENCINA)
/* FUNCTION: SubmitCmd
                                                                                                       Term.pClientData[iNewTerm].pTxn =
* PURPOSE:
                   This function allocated a new terminal id in the Term
                                                                         pCTPCC_ENCINA_new();
                                                                                             else if (Reg.eTxnMon == COM)
structure array.
                                                                                                       Term.pClientData[iNewTerm].pTxn =
                                                                         pCTPCC_COM_new( Reg.bCOM_SinglePool );
                                                                                             else if (Reg.eDB_Protocol == ODBC)
                                                                                                       \overline{\text{Term.pClientData[iNewTerm].pTxn}} =
void SubmitCmd(EXTENSION_CONTROL_BLOCK *pECB, char *szBuffer)
                                                                         pCTPCC ODBC new( szServer, szUser, szPassword, szMyComputerName,
                             iNewTerm;
                                                                         szDatabase);
          int
                    *ptr = pECB->lpszQueryString;
                                                                                             else if (Reg.eDB Protocol == DBLIB)
         char
                                                                                                       Term.pClientData[iNewTerm].pTxn =
         char
                                       = \{ 0 \};
                                                                         pCTPCC_DBLIB_new( szServer, szUser, szPassword, szMyComputerName,
                   szVersion[32]
                   szServer[32]
                                        = \{ 0 \};
                                                                         szDatabase);
         char
                                       = "sa";
          char
                    szUser[32]
          char
                    szPassword[32]
                                       = \{ 0 \};
                                                                                   catch (...)
                   szDatabase[32]
          char
                                       = "tpcc";
                                                                                             TermDelete(iNewTerm);
         // validate version field; the version field ensures that the RTE is
                                                                                             throw:
                                                                                                                 // pass exception upward
synchronized with the web client
         GetKeyValue(&ptr, "VERSION", szVersion, sizeof(szVersion),
ERR VERSION_MISMATCH);
                                                                                   MakeMainMenuForm(iNewTerm,
         if ( stremp( szVersion, WEBCLIENT VERSION ) )
                                                                         Term.pClientData[iNewTerm].iSyncId, szBuffer);
                    throw new CWEBCLNT_ERR(
ERR VERSION MISMATCH);
                                                                         /* FUNCTION: StatsCmd
          if(Reg.eTxnMon == None)
                                                                          * PURPOSE:
                                                                                             This function returns to the browser the total number of
                   // parse Server name
                                                                         active terminal ids.
                   GetKeyValue(&ptr, "db server", szServer,
                                                                                                       This routine is for development/debugging
sizeof(szServer), ERR_NO_SERVER_SPECIFIED);
                                                                         purposes.
                    // parse User name
                   GetKeyValue(&ptr, "db_user", szUser, sizeof(szUser),
NO_ERR);
                                                                         void StatsCmd(EXTENSION_CONTROL_BLOCK *pECB, char *szBuffer)
                   // parse Password
                   GetKeyValue(&ptr, "db_passwd", szPassword,
sizeof(szPassword), NO ERR);
                                                                                   int i;
                   // parse Database name
                                                                                   int
                                                                                             iTotal;
```

```
EnterCriticalSection(&TermCriticalSection);
         iTotal = 0;
         for(i=0; i<Term.iNumEntries; i++)
                   if (Term.pClientData[i].iNextFree == -1)
                            iTotal++;
         LeaveCriticalSection(&TermCriticalSection);
         wsprintf( szBuffer,
                            "<HTML><HEAD><TITLE>TPC-C Web
Client Stats</TITLE></HEAD>"
                            "<BODY><B><BIG> Total Active
Connections: %d </BIG></B><BR></BODY></HTML>"
                            , iTotal );
char *CWEBCLNT ERR::ErrorText()
         static SERRORMSG errorMsgs[] =
                            ERR COMMAND UNDEFINED,
                            "Command undefined."
         },
                            ERR D ID INVALID,
                            "Invalid District ID Must be 1 to 10."
                            ERR DELIVERY CARRIER ID RANGE,
         "Delivery Carrier ID out of range must be 1 - 10."
                            ERR DELIVERY CARRIER INVALID,
         "Delivery Carrier ID invalid must be numeric 1 - 10."
         },
                            ERR_DELIVERY_MISSING_OCD_KEY,
         "Delivery missing Carrier ID key \"OCD*\"."
                            ERR_DELIVERY_THREAD_FAILED,
                   "Could not start delivery worker thread."
                            ERR GETPROCADDR FAILED,
                            "Could not map proc in DLL. GetProcAddr
error. DLL="
                            ERR_HTML_ILL_FORMED,
                   "Required key field is missing from HTML string."
                            ERR_INVALID_SYNC_CONNECTION,
         "Invalid Terminal Sync ID."
                            ERR INVALID TERMID,
                            "Invalid Terminal ID."
         },
                            ERR LOADDLL FAILED,
                            "Load of DLL failed. DLL="
},
                            ERR_MAX_CONNECTIONS_EXCEEDED,
                   "No connections available. Max Connections is probably
too low." },
                            ERR MISSING REGISTRY ENTRIES,
         "Required registry entries are missing. Rerun INSTALL to correct."
ERR_NEWORDER_CUSTOMER_INVALID,
```

```
"New Order customer id invalid data type, range = 1 to 3000."
                           ERR NEWORDER CUSTOMER KEY.
                  "New Order missing Customer key \"CID*\"."
                           ERR NEWORDER DISTRICT INVALID,
         "New Order District ID Invalid range 1 - 10."
                           ERR_NEWORDER_FORM_MISSING_DID,
                  "New Order missing District key \"DID*\"."
                           ERR NEWORDER ITEMID INVALID,
         "New Order Item Id is wrong data type, must be numeric."
                           ERR NEWORDER ITEMID RANGE,
                  "New Order Item Id is out of range. Range = 1 to
999999."
ERR_NEWORDER_ITEMID_WITHOUT_SUPPW,
                                                        "New Order
Item_Id field entered without a corresponding Supp_W."
                           ERR_NEWORDER_MISSING_IID_KEY,
         "New Order missing Item Id key \"IID*\"."
                           ERR NEWORDER MISSING QTY KEY,
                  "New Order Missing Qty key \"Qty##*\"."
ERR_NEWORDER_MISSING_SUPPW_KEY,
"New Order missing Supp_W key \"SP##*\"."
                           ERR NEWORDER NOITEMS ENTERED,
                  "New Order No order lines entered."
                           ERR NEWORDER QTY INVALID,
                  "New Order Qty invalid must be numeric range 1 - 99."
                           ERR_NEWORDER_QTY_RANGE,
                           "New Order Qty is out of range. Range = 1 to
99 "
ERR NEWORDER OTY WITHOUT SUPPW,
"New Order Qty field entered without a corresponding Supp_W."
                           ERR NEWORDER SUPPW INVALID.
                   "New Order Supp W invalid data type must be numeric."
                           ERR NO SERVER SPECIFIED.
                  "No Server name specified."
                           ERR_ORDERSTATUS_CID_AND_CLT,
         "Order Status Only Customer ID or Last Name may be entered, not
both."
                           ERR ORDERSTATUS CID INVALID,
         "Order Status Customer ID invalid, range must be numeric 1 -
3000."
                           ERR ORDERSTATUS CLT RANGE,
                  "Order Status Customer last name longer than 16
characters."
                           ERR_ORDERSTATUS_DID_INVALID,
         "Order Status District invalid, value must be numeric 1 - 10."
ERR_ORDERSTATUS_MISSING_CID_CLT,
                                                        "Order
Status Either Customer ID or Last Name must be entered."
ERR_ORDERSTATUS_MISSING_CID_KEY,
                                                       "Order
```

```
Status missing Customer key \"CID*\"."
         },
ERR ORDERSTATUS MISSING CLT KEY,
                                                        "Order
Status missing Customer Last Name key \"CLT*\"."
ERR ORDERSTATUS MISSING DID KEY,
                                                        "Order
Status missing District key \"DID*\"."
         },
                            ERR PAYMENT CDI INVALID,
                   "Payment Customer district invalid must be numeric."
                            ERR PAYMENT CID AND CLT,
                   "Payment Only Customer ID or Last Name may be
entered, not both."
                            ERR PAYMENT CUSTOMER INVALID,
                   "Payment Customer data type invalid, must be numeric."
                            ERR PAYMENT CWI INVALID,
                   "Payment Customer Warehouse invalid, must be
numeric."
                            ERR PAYMENT DISTRICT INVALID,
         "Payment District ID is invalid, must be 1 - 10."
                            ERR PAYMENT HAM INVALID,
                   "Payment Amount invalid data type must be numeric."
                            ERR PAYMENT HAM RANGE,
                            "Payment Amount out of range, 0 - 9999.99."
                                                        },
ERR PAYMENT LAST NAME TO LONG,
"Payment Customer last name longer than 16 characters."
                            ERR PAYMENT MISSING CDI KEY,
         "Payment missing Customer district key \"CDI*\"."
                            ERR_PAYMENT_MISSING_CID_CLT,
         "Payment Either Customer ID or Last Name must be entered."
                            ERR_PAYMENT_MISSING_CID_KEY,
         "Payment missing Customer Key\"CID*\"."
                            ERR PAYMENT MISSING CLT KEY,
         "Payment missing Customer Last Name key \"CLT*\"."
                   }.
                            ERR PAYMENT MISSING CWI KEY,
         "Payment missing Customer Warehouse key \"CWI*\"."
                  }.
                            ERR_PAYMENT_MISSING_DID_KEY,
         "Payment missing District Key \"DID*\"."
                            ERR PAYMENT MISSING HAM KEY,
         "Payment missing Amount key \"HAM*\"."
ERR_STOCKLEVEL_MISSING_THRESHOLD_KEY,
                                                        "Stock
Level; missing Threshold key \"TT*\"."
ERR STOCKLEVEL THRESHOLD INVALID,
                                                        "Stock
Level; Threshold value must be in the range = 1 - 99."
ERR_STOCKLEVEL_THRESHOLD_RANGE,
"Stock Level Threshold out of range, range must be 1 - 99."
},
```

```
ERR VERSION MISMATCH,
                     "Invalid version field. RTE and Web Client are probably
out of sync." },
                               ERR_W_ID_INVALID,
                               "Invalid Warehouse ID."
          },
                                                               }
          };
          char szTmp[256];
          int i = 0;
          while (TRUE)
                     if(errorMsgs[i].szMsg[0] == 0)
                               strcpy( szTmp, "Unknown error number." );
                               break:
                    if (m Error == errorMsgs[i].iError)
                               strcpy( szTmp, errorMsgs[i].szMsg );
                               break:
                    i++*
          if (m szTextDetail)
                     strcat( szTmp, m_szTextDetail );
          if (m_SystemErr)
                     wsprintf( szTmp+strlen(szTmp), " Error=%d",
m_SystemErr);
          m_szErrorText = new char[strlen(szTmp)+1];
          strcpy( m szErrorText, szTmp );
          return m_szErrorText;
/* FUNCTION: GetKeyValue
* PURPOSE:
                    This function parses a http formatted string for specific
key values.
* ARGUMENTS:
                                                    *pQueryString
http string from client browser
                                         char
                               key value to look for
*pKey
                                          char
*pValue
                               character array into which to place key's value
                                         int
                               maximum length of key value array.
iMax
                                          WEBERROR
                                         error value to throw
err
* RETURNS:
                               nothing.
* ERROR:
                               if (the pKey value is not found) then
                                                    if (err == 0)
                                                               return
(empty string)
                                                    else
                                                               throw
CWEBCLNT_ERR(err)
```

```
* COMMENTS:
                     http keys are formatted either KEY=value& or
KEY=value\setminus 0. This DLL formats
                                          TPC-C input fields in such a
manner that the keys can be extracted in the
                                          above manner
void GetKeyValue(char **pQueryString, char *pKey, char *pValue, int iMax,
WEBERROR err)
          char *ptr;
          if (!(ptr=strstr(*pQueryString, pKey)))
                     goto ErrorExit;
          ptr += strlen(pKey);
           if (*ptr!='=')
                     goto ErrorExit;
           ptr++;
           iMax--; // one position is for terminating null
          while( *ptr && *ptr != '&' && iMax)
                     pValue++ = ptr++;
                     iMax--;
           *pValue = 0; // terminating null
          *pQueryString = ptr;
           return;
ErrorExit:
           if (err != NO_ERR)
                     throw new CWEBCLNT ERR( err );
           *pValue = 0; // return empty result string
/* FUNCTION: GetIntKeyValue
* PURPOSE:
                     This function parses a http formatted string for a specific
key value.
* ARGUMENTS:
                                                     *pQueryString
http string from client browser
                                          char
                                key value to look for
*pKey
                                          WEBERROR
NoKeyErr
                     error value to throw if key not found
                                          WEBERROR
NotIntErr
                     error value to throw if value not numeric
                                                                              */
* RETURNS:
                                integer
* ERROR:
                                if (the pKey value is not found) then
                                                     if (NoKeyErr !=
NO_ERR)
                                                               throw
CWEBCLNT ERR(err)
                                                     else
                                                               return 0
                                          else if (non-numeric char found)
then
                                                     if (NotIntErr !=
NO ERR) then
                                                                throw
CWEBCLNT_ERR(err)
                                                     else
                                                               return 0
```

```
* COMMENTS:
                    http keys are formatted either KEY=value& or
KEY=value\0. This DLL formats
                                         TPC-C input fields in such a
manner that the keys can be extracted in the
                                         above manner
int GetIntKeyValue(char **pQueryString, char *pKey, WEBERROR
NoKeyErr, WEBERROR NotIntErr)
          char *ptr0;
          char *ptr;
          if ( !(ptr=strstr(*pQueryString, pKey)) )
                    goto ErrorNoKey;
          ptr += strlen(pKey);
          if (*ptr!='=')
                    goto ErrorNoKey;
          ptr++;
                               // remember starting point
          ptr0 = ptr;
          // scan string until a terminator (null or &) or a non-digit
          while( *ptr && *ptr != '&' && isdigit(*ptr) )
                    ptr++;
          // make sure we stopped scanning for the right reason
          if ((ptr0 == ptr) || (*ptr && *ptr != '&'))
                    if (NotIntErr != NO ERR)
                               throw new CWEBCLNT_ERR( NoKeyErr );
                    return 0;
          *pQueryString = ptr;
          return atoi(ptr0);
ErrorNoKey:
          if (NoKeyErr != NO ERR)
                    throw new CWEBCLNT ERR( NoKeyErr );
          return 0;
/* FUNCTION: TermInit
* PURPOSE:
                    This function initializes the client terminal structure; it is
called when the TPCC.DLL
                               is first loaded by the inet service.
void TermInit(void)
          EnterCriticalSection(&TermCriticalSection);
          Term.iMasterSyncId = 1;
          Term.iNumEntries = Reg.dwMaxConnections+1;
          Term.pClientData
                              = NULL;
          Term.pClientData
(PCLIENTDATA)malloc(Term.iNumEntries * sizeof(CLIENTDATA));
          if (Term.pClientData == NULL)
                    LeaveCriticalSection(&TermCriticalSection);
                    throw new CWEBCLNT ERR(
ERR_MEM_ALLOC_FAILED );
```

```
ZeroMemory( Term.pClientData, Term.iNumEntries *
sizeof(CLIENTDATA));
          Term.iFreeList
                                           = Term.iNumEntries-1;
          // build free list
          // note: Term.pClientData[0].iNextFree gets set to -1, which marks it
as "in use"
                This is intentional, as the zero entry is used as an anchor and
never
                                allocated as an actual terminal.
          for(int i=0; i<Term.iNumEntries; i++)
                     Term.pClientData[i].iNextFree = i-1;
          LeaveCriticalSection(&TermCriticalSection);
/* FUNCTION: TermDeleteAll
* PURPOSE:
                     This function frees allocated resources associated with
the terminal structure.
* ARGUMENTS:
* RETURNS:
                                None
* COMMENTS:
                     This function is called only when the inet service unloads
the TPCC.DLL
*/
void TermDeleteAll(void)
          EnterCriticalSection(&TermCriticalSection);
          for(int i=1; i<Term.iNumEntries; i++)
                     if (Term.pClientData[i].iNextFree == -1)
                                delete Term.pClientData[i].pTxn;
          Term.iFreeList
                                           = 0;
          Term.iNumEntries
          if ( Term.pClientData )
                     free(Term.pClientData);
                                = NULL;
          Term.pClientData
          LeaveCriticalSection(&TermCriticalSection);
/* FUNCTION: TermAdd
* PURPOSE:
                     This function assigns a terminal id which is used to
identify a client browser.
* RETURNS:
                                int
                                                     assigned terminal id
*/
int TermAdd(void)
          DWORD i;
                                iNewTerm, iTickCount;
          if (Term.iNumEntries == 0)
                     return -1;
          EnterCriticalSection(&TermCriticalSection);
          if (Term.iFreeList != 0)
```

```
// position is available
          {
                     iNewTerm = Term.iFreeList;
                     Term.iFreeList =
Term.pClientData[iNewTerm].iNextFree;
                     Term.pClientData[iNewTerm].iNextFree = -1; //
indicates this position is in use
          else
                     // no open slots, so find the slot that hasn't been used in
the longest time and reuse it
                     for(iNewTerm=1, i=1, iTickCount=0x7FFFFFFF;
i<Reg.dwMaxConnections; i++)
                                if (iTickCount >
Term.pClientData[i].iTickCount)
                                          iTickCount =
Term.pClientData[i].iTickCount;
                                          iNewTerm = i;
                     // if oldest term is less than one minute old, it probably
means that more connections
                     // are being attempted than were specified as "Max
Connections" at install. In this case,
                     // do not bump existing connection; instead, return error
to requestor.
                     if ((GetTickCount() - iTickCount) < 60000)
                                LeaveCriticalSection(&TermCriticalSection);
                                throw new CWEBCLNT ERR(
ERR_MAX_CONNECTIONS_EXCEEDED );
          Term.pClientData[iNewTerm].iTickCount = GetTickCount();
          Term.pClientData[iNewTerm].iSyncId = Term.iMasterSyncId++;
          Term.pClientData[iNewTerm].pTxn = NULL;
          LeaveCriticalSection(&TermCriticalSection);
          return iNewTerm;
/* FUNCTION: TermDelete
* PURPOSE:
                     This function makes a terminal entry in the Term array
available for reuse.
* ARGUMENTS:
                     int
                     Terminal id of client exiting
id
void TermDelete(int id)
          if (id > 0 && id < Term.iNumEntries)
                     delete Term.pClientData[id].pTxn;
                     // put onto free list
                     EnterCriticalSection(&TermCriticalSection);
                     Term.pClientData[id].iNextFree = Term.iFreeList;
                     Term.iFreeList = id;
                     LeaveCriticalSection(&TermCriticalSection);
          }
```

```
"<INPUT TYPE=\"submit\" NAME=\"CMD\"
                                                                    VALUE=\"..Order-Status..\">
/* FUNCTION: MakeErrorForm
                                                                                       '<INPUT TYPE=\"submit\" NAME=\"CMD\"</pre>
                                                                    VALUE=\"..Stock-Level..\">"
                                                                                       "<INPUT TYPE=\"submit\" NAME=\"CMD\"
void ErrorForm(EXTENSION CONTROL BLOCK *pECB, int iType, int
                                                                    VALUE=\"..Exit..\">"
                                                                                      "</FORM></BODY></HTML>"
iErrorNum, int iTermId, int iSyncId, char *szErrorText, char *szBuffer)
                                                                                      , MAIN MENU FORM, iTermId, iSyncId);
         wsprintf(szBuffer,
                  "<HTML><HEAD><TITLE>TPC-C
Error</TITLE></HEAD><BODY>"
                                                                    /* FUNCTION: MakeStockLevelForm
                  "<FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">"
                  "<INPUT TYPE=\"hidden\" NAME=\"STATUSID\"
                                                                    * PURPOSE:
                                                                                      This function constructs the Stock Level HTML page.
VALUE=\"%d\">"
                  "<INPUT TYPE=\"hidden\" NAME=\"ERROR\"
                                                                                      The internal client buffer is created when the terminal id
                                                                    * COMMENTS:
VALUE=\"%d\">"
                                                                    is assigned and should not
                  "<INPUT TYPE=\"hidden\" NAME=\"FORMID\"
                                                                                                         be freed except when the client
VALUE=\"\%d\">"
                                                                    terminal id is no longer needed.
                  "<INPUT TYPE=\"hidden\" NAME=\"TERMID\"
VALUE=\"%d\">"
                  "<INPUT TYPE=\"hidden\" NAME=\"SYNCID\"
                                                                    void MakeStockLevelForm(int iTermId, STOCK LEVEL DATA
VALUE=\"%d\">"
                                                                    *pStockLevelData, BOOL bInput, char *szForm)
                  "<BOLD>An Error Occurred</BOLD><BR><BR>"
                                                                             int
                                                                                      c;
                  "<BR><BR><HR>"
                  "<INPUT TYPE=\"submit\" NAME=\"CMD\"
                                                                             c = wsprintf(szForm,
VALUE=\"..NewOrder..\">"
                                                                                       "<HTML><HEAD><TITLE>TPC-C Stock
                  "<INPUT TYPE=\"submit\" NAME=\"CMD\"
                                                                    Level</TITLE></HEAD><FORM ACTION=\"tpcc.dll\"
VALUE=\"..Payment..\">"
                                                                    METHOD=\"GET\">"
                                                                                      "<INPUT TYPE=\"hidden\" NAME=\"STATUSID\"
                  "<INPUT TYPE=\"submit\" NAME=\"CMD\"
                                                                    VALUE=\"0\">"
VALUE=\"..Delivery..\">"
                  "<INPUT TYPE=\"submit\" NAME=\"CMD\"
                                                                                      "<INPUT TYPE=\"hidden\" NAME=\"ERROR\"
VALUE=\"..Order-Status..\">"
                                                                    VALUE=\"0\">"
                  "<INPUT TYPE=\"submit\" NAME=\"CMD\"
                                                                                      "<INPUT TYPE=\"hidden\" NAME=\"FORMID\"
VALUE=\"..Stock-Level..\">
                                                                    VALUE=\"%d\">"
                   "<INPUT TYPE=\"submit\" NAME=\"CMD\"
                                                                                      "<INPUT TYPE=\"hidden\" NAME=\"TERMID\"
VALUE=\"..Exit..\">'
                                                                    VALUE=\"%d\">"
                  "</FORM></BODY></HTML>"
                                                                                      "<INPUT TYPE=\"hidden\" NAME=\"SYNCID\"
                  , iType, iErrorNum, MAIN_MENU_FORM, iTermId,
                                                                    VALUE=\"%d\">"
                                                                                       "<PRE><font face=\"Courier\">
iSyncId, szErrorText );
                                                                    Stock-Level<BR>"
                                                                                      "Warehouse: %4.4d District: %2.2d BR > BR > ",
                                                                                      STOCK LEVEL FORM, iTermId,
/* FUNCTION: MakeMainMenuForm
                                                                    Term.pClientData[iTermId].iSyncId,
                                                                                       Term.pClientData[iTermId].w id,
                                                                    Term.pClientData[iTermId].d_id);
void MakeMainMenuForm(int iTermId, int iSyncId, char *szForm)
         wsprintf(szForm,
                                                                             if (bInput)
                  "<HTML><HEAD><TITLE>TPC-C Main
                                                                             {
Menu</TITLE></HEAD><BODY>'
                                                                                      strcpy(szForm+c,
                  "Select Desired Transaction. <BR><HR>"
                                                                                                "Stock Level Threshold: <INPUT
                  "<FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">"
                                                                    NAME=\"TT*\" SIZE=2><BR>
                                                                                               <BR>"
                  "<INPUT TYPE=\"hidden\" NAME=\"STATUSID\"
                                                                                                "low stock: </font><BR> <BR> <BR>
VALUE=\"0\">"
                                                                    <BR> <BR> <BR> <BR> <BR> <BR> "
                  "<INPUT TYPE=\"hidden\" NAME=\"ERROR\"
                                                                                                " <BR> <BR> <BR> <BR> <BR>
VALUE=\"0\">"
                                                                    <BR></PRF><HR>"
                  "<INPUT TYPE=\"hidden\" NAME=\"FORMID\"
                                                                                                "<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"%d\">"
                                                                    VALUE=\"Process\">"
                  "<INPUT TYPE=\"hidden\" NAME=\"TERMID\"
                                                                                                "<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"%d\">"
                                                                    VALUE=\"Menu\">"
                  "<INPUT TYPE=\"hidden\" NAME=\"SYNCID\"
                                                                                                "</FORM></HTML>");
VALUE=\"%d\">"
                  "<INPUT TYPE=\"submit\" NAME=\"CMD\"
                                                                             else
VALUE=\"..NewOrder..\">"
                  "<INPUT TYPE=\"submit\" NAME=\"CMD\"
                                                                                      wsprintf(szForm+c,
VALUE=\"..Payment..\">"
                                                                                                "Stock Level Threshold: %2.2d BR > BR > "
                                                                                                "low stock: %3.3d</font> <BR> <BR> <BR>
                  "<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"..Delivery..\">'
                                                                    <BR> <BR> <BR> <BR> <BR> "
```

```
" <BR> <BR> <BR> <BR> <BR>
                                                                                                                                                  "District: <INPUT NAME=\"DID*\"
                                                                                                       SIZE=1>
<BR> <BR></PRE><HR>"
                                                                                                                                      Date: <BR>"
                                          "<INPUT TYPE=\"submit\" NAME=\"CMD\"
                                                                                                                                                  "Customer: <INPUT NAME=\"CID*\"
VALUE=\"..NewOrder..\">"
                                                                                                       SIZE=4> Name:
                                                                                                                                              Credit:
                                                                                                                                                         %Disc:<BR>"
                                          "<INPUT TYPE=\"submit\" NAME=\"CMD\"
                                                                                                                                                  "Order Number:
                                                                                                                                                                               Number of Lines:
                                                                                                                        D_tax:<BR> <BR>"
VALUE=\"..Payment..\">"
                                                                                                        W tax:
                                                                                                                                                  " Supp_W Item_Id Item Name
                                          "<INPUT TYPE=\"submit\" NAME=\"CMD\"
                                                                                                                                                                                                       Qty
VALUE=\"..Delivery..\">"
                                                                                                       Stock B/G Price Amount SR>
                                                                                                                                                  " <INPUT NAME=\"SP00*\" SIZE=4>
                                          "<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"..Order-Status..\">"
                                                                                                        <INPUT NAME=\"IID00*\" SIZE=6>
                                                                                                                                                                              <INPUT
                                                                                                       NAME=\"Qty00*\" SIZE=1><BR>"
                                          "<INPUT TYPE=\"submit\" NAME=\"CMD\"
                                                                                                                                                  " <INPUT NAME=\"SP01*\" SIZE=4>
VALUE=\" Stock-Level \">"
                                          "<INPUT TYPE=\"submit\" NAME=\"CMD\"
                                                                                                        <INPUT NAME=\"IID01*\" SIZE=6>
                                                                                                                                                                              <INPUT
                                                                                                       NAME=\"Qty01*\" SIZE=1><BR>"
VALUE=\"..Exit..\">"
                                                                                                                                                  " <INPUT NAME=\"SP02*\" SIZE=4>
                                          "</FORM></HTML>"
                                          , pStockLevelData->threshold,
                                                                                                        <INPUT NAME=\"IID02*\" SIZE=6>
                                                                                                                                                                              <INPUT
                                                                                                       NAME=\"Qty02*\" SIZE=1><BR>"
pStockLevelData->low stock);
                                                                                                                                                  " <INPUT NAME=\"SP03*\" SIZE=4>
                                                                                                        <INPUT NAME=\"IID03*\" SIZE=6>
                                                                                                                                                                              <INPUT
                                                                                                       NAME=\"Qty03*\" SIZE=1><BR>"
                                                                                                                                                  " <INPUT NAME=\"SP04*\" SIZE=4>
/* FUNCTION: MakeNewOrderForm
                                                                                                        <INPUT NAME=\"IID04*\" SIZE=6>
                                                                                                                                                                              <INPUT
* COMMENTS:
                           The internal client buffer is created when the terminal id
                                                                                                       NAME=\"Qty04*\" SIZE=1><BR>"
                                                                                                                                                  " <INPUT NAME=\"SP05*\" SIZE=4>
is assigned and should not
                                                        be freed except when the client
                                                                                                        <INPUT NAME=\"IID05*\" SIZE=6>
                                                                                                                                                                              <INPUT
                                                                                                       NAME=\"Qty05*\" SIZE=1><BR>"
terminal id is no longer needed.
                                                                                                                                                  " <INPUT NAME=\"SP06*\" SIZE=4>
                                                                                                        <INPUT NAME=\"IID06*\" SIZE=6>
                                                                                                                                                                              <INPUT
void MakeNewOrderForm(int iTermId, NEW ORDER DATA
                                                                                                       NAME=\"Qty06*\" SIZE=1><BR>"
*pNewOrderData, BOOL bInput, char *szForm)
                                                                                                                                                  " <INPUT NAME=\"SP07*\" SIZE=4>
                                                                                                        <INPUT NAME=\"IID07*\" SIZE=6>
                                                                                                                                                                              <INPUT
                                                                                                       NAME=\"Qty07*\" SIZE=1><BR>"
              int
              BOOL
                           bValid-
                                                                                                                                                  " <INPUT NAME=\"SP08*\" SIZE=4>
                            char szBR[] = " < BR > < BR 
              static
                                                                                                        <INPUT NAME=\"IID08*\" SIZE=6>
                                                                                                                                                                              <INPUT
<BR> <BR> <BR> <BR> <BR> <BR> :";
                                                                                                       NAME=\"Qty08*\" SIZE=1><BR>"
                                                                                                                                                  " <INPUT NAME=\"SP09*\" SIZE=4>
                                                                                                        <INPUT NAME=\"IID09*\" SIZE=6>
              if (!bInput)
                                                                                                                                                                              <INPUT
                                                                                                       NAME=\"Qty09*\" SIZE=1><BR>"
                           assert( pNewOrderData->exec status code == eOK ||
pNewOrderData->exec_status_code == eInvalidItem );
                                                                                                                                                  " <INPUT NAME=\"SP10*\" SIZE=4>
                                                                                                        <INPUT NAME=\"IID10*\" SIZE=6>
                                                                                                                                                                              <INPUT
              bValid = (bInput || (pNewOrderData->exec status code == eOK));
                                                                                                       NAME=\"Qty10*\" SIZE=1><BR>"
                                                                                                                                                  " <INPUT NAME=\"SP11*\" SIZE=4>
                                                                                                        <INPUT NAME=\"IID11*\" SIZE=6>
              c = wsprintf(szForm,
                                                                                                       NAME=\"Qty11*\" SIZE=1><BR>"
                            "<HTML><HEAD><TITLE>TPC-C New
                                                                                                                                                  " <INPUT NAME=\"SP12*\" SIZE=4>
Order</TITLE></HEAD><BODY>
                            "<FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">"
                                                                                                        <INPUT NAME=\"IID12*\" SIZE=6>
                                                                                                                                                                              <INPUT
                            "<INPUT TYPE=\"hidden\" NAME=\"STATUSID\"
                                                                                                       NAME=\"Qty12*\" SIZE=1><BR>"
VALUE=\"%d\">"
                                                                                                                                                  " <INPUT NAME=\"SP13*\" SIZE=4>
                            "<INPUT TYPE=\"hidden\" NAME=\"ERROR\"
                                                                                                        <INPUT NAME=\"IID13*\" SIZE=6>
                                                                                                                                                                              <INPUT
VALUE=\"0\">"
                                                                                                        NAME=\"Qty13*\" SIZE=1><BR>"
                                                                                                                                                  " <INPUT NAME=\"SP14*\" SIZE=4>
                            "<INPUT TYPE=\"hidden\" NAME=\"FORMID\"
                                                                                                        <INPUT NAME=\"IID14*\" SIZE=6>
VALUE=\"%d\">"
                                                                                                                                                                              <INPUT
                                                                                                       NAME=\"Qty14*\" SIZE=1><BR>"
                            "<INPUT TYPE=\"hidden\" NAME=\"TERMID\"
VALUE=\"%d\">"
                                                                                                                                                  "Execution Status:
                                                                                                       Total:<BR>"
                            "<INPUT TYPE=\"hidden\" NAME=\"SYNCID\"
VALUE=\"%d\">"
                                                                                                                                                  "</font></PRE><HR>"
                            "<PRE><font face=\"Courier\">
                                                                                                                                                  "<INPUT TYPE=\"submit\" NAME=\"CMD\"
New Order < BR > "
                                                                                                        VALUE=\"Process\">"
                            , bValid?0:ERR BAD ITEM ID,
                                                                                                                                                  "<INPUT TYPE=\"submit\" NAME=\"CMD\"
NEW_ORDER_FORM, iTermId, Term.pClientData[iTermId].iSyncId);
                                                                                                        VALUE=\"Menu\">"
                                                                                                                                                  "</FORM></HTML>"
              if (bInput)
                                                                                                                                                  );
                            c += wsprintf(szForm+c, "Warehouse: %4.4d",
                                                                                                                      else
Term.pClientData[iTermId].w_id );
                                                                                                                                   c += wsprintf(szForm+c, "Warehouse: %4.4d District:
                                                                                                       %2.2d
                           strcpy( szForm+c,
                                                                                                                                  Date: ",
                                                                                                                                                  pNewOrderData->w id,
```

```
pNewOrderData->d id);
                   if (bValid)
                             c += wsprintf(szForm+c,
"%2.2d-%2.2d-%4.4d %2.2d:%2.2d:%2.2d",
                                       pNewOrderData->o entry d.day,
pNewOrderData->o_entry_d.month,
                                       pNewOrderData->o entry d.year,
                                       pNewOrderData->o_entry_d.hour,
pNewOrderData->o entry d.minute,
pNewOrderData->o_entry_d.second);
                   c += wsprintf(szForm+c, "<BR>Customer: %4.4d
Name: %-16s Credit: %-2s
                             pNewOrderData->c id,
pNewOrderData->c_last, pNewOrderData->c_credit);
                   if (bValid)
                             c += sprintf(szForm+c,
                                                           "%%Disc:
%5.2f
            <BR>"
                                                           "Order
Number: %8.8d Number of Lines: %2.2d
                                         W tax: %5.2f D tax: %5.2f
<RR> <RR>'
                                                           "Supp W
Item Id Item Name
                           Qty Stock B/G Price Amount SR>",
100.0*pNewOrderData->c discount,
                                       pNewOrderData->o_id,
                                       pNewOrderData->o ol cnt,
                                       100.0 * pNewOrderData->w_tax,
                                       100.0 * pNewOrderData->d tax);
                             for(i=0; i<pNewOrderData->o ol cnt; i++)
                                       c += sprintf(szForm+c, " %4.4d
%6.6d %-24s %2.2d %3.3d %1.1s $%6.2f $%7.2f <BR>",
pNewOrderData->OL[i].ol supply w id,
pNewOrderData->OL[i].ol i id,
pNewOrderData->OL[i].ol_i_name,
pNewOrderData->OL[i].ol_quantity,
pNewOrderData->OL[i].ol_stock,
pNewOrderData->OL[i].ol_brand_generic,
pNewOrderData->OL[i].ol i price,
pNewOrderData->OL[i].ol amount);
                   else
                    {
                             c += wsprintf(szForm+c,
                                       "%Disc:<BR>"
                                       "Order Number: %8.8d Number
of Lines:
                          D_tax:<BR> <BR>'
               W_tax:
```

```
"Supp_W Item_Id Item Name
     Qty Stock B/G Price Amount SR>
                                      , pNewOrderData->o id);
                            i = 0
                   strncpy( szForm+c, szBR, (15-i)*5 );
                   c += (15-i)*5;
                   if (bValid)
                            c += sprintf(szForm+c, "Execution Status:
Transaction committed.
                               Total: $%8.2f ",
                                      pNewOrderData->total_amount);
                   else
                            c += wsprintf(szForm+c, "Execution Status:
Item number is not valid.
                               Total:");
                   strcpy(szForm+c,
                             " <BR></font></PRE><HR>"
                             "<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"..NewOrder..\">"
                            "<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"..Payment..\">"
                            "<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"..Delivery..\">"
                            "<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"..Order-Status..\">
                            "<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"..Stock-Level..\">"
                            "<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"..Exit..\">"
                            "</FORM></HTML>"
                            );
         }
/* FUNCTION: MakePaymentForm
* COMMENTS:
                   The internal client buffer is created when the terminal id
is assigned and should not
                                      be freed except when the client
terminal id is no longer needed.
void MakePaymentForm(int iTermId, PAYMENT_DATA *pPaymentData,
BOOL bInput, char *szForm)
         int c;
         c = wsprintf(szForm,
                   "<HTML><HEAD><TITLE>TPC-C
Payment</TITLE></HEAD><BODY>"
                   "<FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">"
                   "<INPUT TYPE=\"hidden\" NAME=\"STATUSID\"
VALUE=\"0\">"
                   "<INPUT TYPE=\"hidden\" NAME=\"ERROR\"
VALUE=\"0\">"
                   "<INPUT TYPE=\"hidden\" NAME=\"FORMID\"
VALUE=\"%d\">"
                   "<INPUT TYPE=\"hidden\" NAME=\"TERMID\"
VALUE=\"%d\">"
                   "<INPUT TYPE=\"hidden\" NAME=\"SYNCID\"
VALUE=\"%d\">"
                   "<PRE><font face=\"Courier\">
Payment<BR>"
                   "Date: "
```

```
, PAYMENT FORM, iTermId,
                                                                                                                                                                  , pPaymentData->c id,
Term.pClientData[iTermId].iSyncId);
                                                                                                                   pPaymentData->c w id,
                                                                                                                                                                  pPaymentData->c d id
                                                                                                                                                                  , pPaymentData->c first,
                if (!bInput)
                                                                                                                   pPaymentData->c_middle, pPaymentData->c_last
                                                                                                                                                                   , pPaymentData->c since.day,
                               c += wsprintf(szForm+c, "%2.2d-%2.2d-%4.4d
                                                                                                                   pPaymentData->c since.month, pPaymentData->c since.year
%2.2d:%2.2d:%2.2d"
                                                                                                                                                                  , pPaymentData->c street 1,
                                               pPaymentData->h date.day,
                                                                                                                   pPaymentData->c credit
                                               pPaymentData->h date.month,
                                                                                                                                                                  );
                                               pPaymentData->h date.year,
                                               pPaymentData->h_date.hour,
                                                                                                                          c += sprintf(szForm+c,
                                               pPaymentData->h_date.minute,
                                                                                                                                                                          %-20s
                                                                                                                                                                                                     %%Disc:
                                               pPaymentData->h date.second);
                                                                                                                   %5.2f<BR>",
                                                                                                                                                                  pPaymentData->c_street 2,
                }
                                                                                                                   100.0*pPaymentData->c discount);
                if (bInput)
                                                                                                                                                  c += wsprintf(szForm+c,
                               c += wsprintf(szForm+c,
                                                                                                                                                                          %-20s %-2s %5.5s-%4.4s
                                                                                                                                                                                                                     Phone:
                                               "<BR> <BR>Warehouse: %4.4d"
                                                                                                                   %6.6s-%3.3s-%3.3s-%4.4s<BR> <BR>".
                                                                      District: <INPUT
                                                                                                                                                                  pPaymentData->c city,
NAME=\"DID*\" SIZE=1><BR> <BR> <BR> <BR> <BR>"
                                                                                                                   pPaymentData->c_state, pPaymentData->c_zip, pPaymentData->c_zip+5,
                                               "Customer: <INPUT NAME=\"CID*\"
                                                                                                                                                                  pPaymentData->c phone,
SIZE=4>"
                                                                                                                   pPaymentData->c phone+6, pPaymentData->c phone+9,
                                               "Cust-Warehouse: <INPUT NAME=\"CWI*\"
                                                                                                                   pPaymentData->c phone+12);
SIZE=4> "
                                               "Cust-District: <INPUT NAME=\"CDI*\"
                                                                                                                                                  c += sprintf(szForm+c,
SIZE=1><BR>"
                                                                                                                                                                  "Amount Paid:
                                                                                                                                                                                               $%7.2f
                                                                                                                                                                                                              New
                                                                                                                   Cust-Balance: $%14.2f<BR>"
                                                                             <INPUT
                                               "Name:
NAME=\"CLT*\" SIZE=16>
                                                           Since: <BR>"
                                                                                                                                                                  "Credit Limit: $\%13.2f\le BR \right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right\right
                                                                                          Credit: <BR>"
                                                                                                                                                                  , pPaymentData->h_amount,
                                                                                          Disc:<BR>"
                                                                                                                   pPaymentData->c balance
                                                                                          Phone: <BR>
                                                                                                                                                                  , pPaymentData->c_credit_lim
<BR>"
                                               "Amount Paid:
                                                                            $<INPUT
NAME=\"HAM*\" SIZE=7>
                                              New Cust-Balance: <BR>"
                                                                                                                                                  if (pPaymentData->c_credit[0] == 'B' &&
                                                "Credit Limit:<BR> <BR>Cust-Data: <BR>
                                                                                                                   pPaymentData->c credit[1] == 'C')
<BR> <BR> <BR> </PRE></HR>"
                                                                                                                                                                  c += wsprintf(szForm+c,
                                               "<INPUT TYPE=\"submit\" NAME=\"CMD\"
                                                                                                                                                                                                  "Cust-Data:
VALUE=\"Process\"><INPUT TYPE=\"submit\" NAME=\"CMD\"
                                                                                                                   %-50.50s<BR>
                                                                                                                                                  %-50.50s<BR>
                                                                                                                                                                                  %-50.50s<BR>
VALUE=\"Menu\">"
                                                                                                                   %-50.50s<BR>".
                                               "</BODY></FORM></HTML>"
                                                                                                                                                                                                 pPaymentData->c data,
                                               , Term.pClientData[iTermId].w_id);
                                                                                                                   pPaymentData->c_data+50, pPaymentData->c_data+100,
                                                                                                                   pPaymentData->c data+150);
               }
               else
                                                                                                                                                  else
                                                                                                                                                                  strcpy(szForm+c, "Cust-Data: <BR> <BR>
                                                                                                                   <BR> <BR>");
                               c += wsprintf(szForm+c,
                                                "<BR> <BR>Warehouse: %4.4d
  District: %2.2d<BR>"
                                                                                                                                                  strcat(szForm,
                                                                                                                                                                                  " <BR></font></PRE><HR>"
                                                                           %-20s<BR>"
                                               "%-20s
                                                                                                                                                                                                                  "<INPUT
                                                                           %-20s<BR>"
                                                                                                                   TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..NewOrder..\">"
                                               "%-20s
                                               "%-20s %-2s %5.5s-%4.4s
                                                                                                                                                                                                                 "<INPUT
                                                                                          %-20s %-2s
                                                                                                                   TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Payment..\">"
%5.5s-%4.4s<BR> <BR>"
                                               "Customer: %4.4d Cust-Warehouse: %4.4d
                                                                                                                                                                                                                 "<INPUT
Cust-District: %2.2d SR>"
                                                                                                                   TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Delivery..\">"
                                                                                                                                                                                                                 "<INPUT
                                               "Name: %-16s %-2s %-16s Since:
%2.2d-%2.2d-%4.4d<BR>"
                                                                                                                   TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Order-Status..\">
                                                       %-20s
                                                                                  Credit: %-2s<BR>"
                                                                                                                                                                                                                 "<INPUT
                                                                                                                   TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Stock-Level..\">"
                                               , Term.pClientData[iTermId].w id,
                                                                                                                                                                                                                 "<INPUT
                                                                                                                   TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Exit..\">"
pPaymentData->d_id
                                               , pPaymentData->w street 1,
pPaymentData->d street 1
                                                                                                                   "</BODY></FORM></HTML>");
                                               , pPaymentData->w_street_2,
pPaymentData->d_street_2
                                               , pPaymentData->w_city,
pPaymentData->w state, pPaymentData->w zip, pPaymentData->w zip+5
                                                                                                                   /* FUNCTION: MakeOrderStatusForm
                                               , pPaymentData->d_city,
pPaymentData->d_state, pPaymentData->d_zip, pPaymentData->d_zip+5
```

```
* COMMENTS:
                   The internal client buffer is created when the terminal id
                                                                                                   "Order-Number: %8.8d Entry-Date:
                                                                      %2.2d-%2.2d-%4.4d %2.2d:%2.2d: Carrier-Number: %2.2d<BR>"
is assigned and should not
                                                                                                   "Supply-W Item-Id Qty Amount
                                      be freed except when the client
terminal id is no longer needed.
                                                                      Delivery-Date<BR>",
                                                                                                   pOrderStatusData->o id,
                                                                                                   pOrderStatusData->o entry d.day,
void\ Make Order Status Form (int\ iTermId, ORDER\_STATUS\_DATA
                                                                                                   pOrderStatusData->o_entry_d.month,
*pOrderStatusData, BOOL bInput, char *szForm)
                                                                                                   pOrderStatusData->o_entry_d.year,
                                                                                                   pOrderStatusData->o entry d.hour,
                                                                                                   pOrderStatusData->o entry d.minute,
         static char szBR[] = " <BR> <BR> <BR> <BR> <BR> <BR> <BR>
                                                                                                   pOrderStatusData->o_entry_d.second,
<BR> <BR> <BR> <BR> <BR> <BR> ";
                                                                                                   pOrderStatusData->o_carrier_id);
         c = wsprintf(szForm,
                                                                                         for(i=0; i< pOrderStatusData->o_ol_cnt; i++)
                   "<HTML><HEAD><TITLE>TPC-C
Order-Status</TITLE></HEAD><BODY>"
                                                                                                   c += sprintf(szForm+c, " %4.4d
                                                                                                                                  %6.6d
                   "<FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">"
                                                                      %2.2d $%8.2f
                                                                                        %2.2d-%2.2d-%4.4d<BR>",
                   "<INPUT TYPE=\"hidden\" NAME=\"STATUSID\"
VALUE=\"0\">"
                                                                      pOrderStatusData->OL[i].ol_supply_w_id,
                   "<INPUT TYPE=\"hidden\" NAME=\"ERROR\"
                                                                                                             pOrderStatusData->OL[i].ol i id,
VALUE=\"0\">"
                   "<INPUT TYPE=\"hidden\" NAME=\"FORMID\"
                                                                      pOrderStatusData->OL[i].ol quantity,
VALUE=\"%d\">"
                   "<INPUT TYPE=\"hidden\" NAME=\"TERMID\"
                                                                      pOrderStatusData->OL[i].ol amount,
VALUE=\"\%d\">"
                   "<INPUT TYPE=\"hidden\" NAME=\"SYNCID\"
                                                                      pOrderStatusData->OL[i].ol_delivery_d.day,
VALUE=\"%d\">"
                   "<PRE><font face=\"Courier\">
                                                                      pOrderStatusData->OL[i].ol_delivery_d.month,
Order-Status<BR>"
                   "Warehouse: %4.4d ".
                                                                      pOrderStatusData->OL[i].ol_delivery_d.year);
                   ORDER STATUS FORM, iTermId,
Term.pClientData[iTermId].iSyncId, Term.pClientData[iTermId].w_id);
                                                                                         strncpy( szForm+c, szBR, (15-i)*5 );
         if (bInput)
                                                                                         c += (15-i)*5;
                   strcpy(szForm+c,
                                                                                         strcpy(szForm+c,
                             "District: <INPUT NAME=\"DID*\"
                                                                                                   "</font></PRE><HR><INPUT
SIZE=1><BR>"
                                                                      TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..NewOrder..\">"
                            "Customer: <INPUT NAME=\"CID*\"
                                                                                                   "<INPUT TYPE=\"submit\" NAME=\"CMD\"
SIZE=4> Name:
                          <INPUT NAME=\"CLT*\" SIZE=23><BR>"
                                                                      VALUE=\"..Payment..\">"
                             "Cust-Balance: <BR> <BR>"
                                                                                                   "<INPUT TYPE=\"submit\" NAME=\"CMD\"
                             "Order-Number:
                                                                      VALUE=\"..Delivery..\">"
                                                 Entry-Date:
  Carrier-Number: <BR>"
                                                                                                   "<INPUT TYPE=\"submit\" NAME=\"CMD\"
                            "Supply-W Item-Id Qty Amount
                                                                      VALUE=\"..Order-Status..\">"
Delivery-Date<BR> <BR> <BR> <BR> "
                                                                                                   "<INPUT TYPE=\"submit\" NAME=\"CMD\"
                             " <BR> <BR> <BR> <BR> <BR>
                                                                      VALUE=\"..Stock-Level..\">"
<BR> <BR> <BR> <BR> </PRE>"
                                                                                                   "<INPUT TYPE=\"submit\" NAME=\"CMD\"
                             "<HR><INPUT TYPE=\"submit\"
                                                                      VALUE=\"..Exit..\">"
NAME=\"CMD\" VALUE=\"Process\"><INPUT TYPE=\"submit\"
                                                                                                   "</BODY></FORM></HTML>");
NAME=\"CMD\" VALUE=\"Menu\">"
                             "</BODY></FORM></HTML>" );
         }
                                                                      /* FUNCTION: MakeDeliveryForm
         else
                                                                       * COMMENTS:
                                                                                         The internal client buffer is created when the terminal id
                   c += wsprintf(szForm+c,
                            "District: %2.2d<BR>"
                                                                      is assigned and should not
                            "Customer: %4.4d Name: %-16s %-2s
                                                                                                             be freed except when the client
%-16s<BR>"
                                                                      terminal id is no longer needed.
                            pOrderStatusData->d id,
pOrderStatusData->c_id,
                            pOrderStatusData->c first,
                                                                      void MakeDeliveryForm(int iTermId, DELIVERY DATA *pDeliveryData,
pOrderStatusData->c_middle, pOrderStatusData->c_last);
                                                                      BOOL bInput, char *szForm)
                   c += sprintf(szForm+c, "Cust-Balance: $\%9.2f\le BR>
                                                                                         c;
<BR>".
                            pOrderStatusData->c balance);
                                                                                c = wsprintf(szForm,
                                                                                          "<HTML><HEAD><TITLE>TPC-C
                   c += wsprintf(szForm+c,
                                                                      Delivery</TITLE></HEAD><BODY>"
```

```
"<FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">"
                                                                                                   filling in the required input variables. it then
                   "<INPUT TYPE=\"hidden\" NAME=\"STATUSID\"
                                                                      calls the SQLNewOrder
VALUE=\"%d\">"
                                                                                                   transaction, constructs the output form and
                   "<INPUT TYPE=\"hidden\" NAME=\"ERROR\"
                                                                      writes it back to client
VALUE=\"0\">"
                                                                                                   browser
                   "<INPUT TYPE=\"hidden\" NAME=\"FORMID\"
VALUE=\"%d\">"
                                                                      void ProcessNewOrderForm(EXTENSION CONTROL BLOCK *pECB, int
                   "<INPUT TYPE=\"hidden\" NAME=\"TERMID\"
VALUE=\"%d\">"
                                                                      iTermId, char *szBuffer)
                   "<INPUT TYPE=\"hidden\" NAME=\"SYNCID\"
                                                                                PNEW_ORDER_DATA
VALUE=\"\%d\">"
                                                                                                                      pNewOrder;
                   "<PRE><font face=\"Courier\">
Delivery<BR>"
                                                                                pNewOrder =
                   "Warehouse: %4.4d<BR> <BR>",
                                                                      Term.pClientData[iTermId].pTxn->BuffAddr_NewOrder();
                   (!bInput && (pDeliveryData->exec status code !=
eOK)) ? ERR_TYPE_DELIVERY_POST: 0,
                                                                                ZeroMemory(pNewOrder, sizeof(NEW_ORDER_DATA));
                   DELIVERY FORM, iTermId,
                                                                                pNewOrder->w id = Term.pClientData[iTermId].w id;
Term.pClientData[iTermId].iSyncId, Term.pClientData[iTermId].w_id);
                                                                                GetNewOrderData(pECB->lpszQueryString, pNewOrder);
         if (bInput)
                                                                                Term.pClientData[iTermId].pTxn->NewOrder();
                   strcpy( szForm+c,
                                                                                pNewOrder =
                             "Carrier Number: <INPUT NAME=\"OCD*\"
                                                                      Term.pClientData[iTermId].pTxn->BuffAddr NewOrder();
SIZE=1><BR> <BR>'
                                                                                MakeNewOrderForm(iTermId, pNewOrder, OUTPUT FORM,
                            "Execution Status: <BR> <BR> <BR> <BR>
                                                                      szBuffer);
<BR> <BR> <BR> "
                            " <BR> <BR> <BR> <BR> <BR>
<BR> <BR> </font></PRE><HR>'
                                                                      /* FUNCTION: void ProcessPaymentForm
                             "<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"Process\">"
                                                                       * PURPOSE:
                                                                                         This function gets and validates the input data from the
                            "<INPUT TYPE=\"submit\" NAME=\"CMD\"
                                                                      payment form
VALUE=\"Menu\">"
                                                                                                   filling in the required input variables. It then
                             "</BODY></FORM></HTML>" );
                                                                      calls the SQLPayment
                                                                                                   transaction, constructs the output form and
         }
         else
                                                                      writes it back to client
                                                                                                   browser.
                   wsprintf( szForm+c,
                             "Carrier Number: %2.2d<BR> <BR>"
                                                                                         EXTENSION CONTROL BLOCK
                                                                                                                                *pECB
                                                                       * ARGUMENTS:
                            "Execution Status: %s <BR> <BR> <BR>
                                                                      passed in structure pointer from inetsry.
<BR> <BR> <BR> <BR> "
                                                                                                            int
                             " <BR> <BR> <BR> <BR> <BR>
                                                                                         iTermId
                                                                                                  client browser terminal id
<BR> <BR> </font></PRE>"
                            "<HR><INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..NewOrder..\">'
                                                                      void ProcessPaymentForm(EXTENSION CONTROL BLOCK *pECB, int
                             "<INPUT TYPE=\"submit\" NAME=\"CMD\'
                                                                      iTermId, char *szBuffer)
VALUE=\"..Payment..\">"
                            "<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"..Delivery..\">"
                                                                                PPAYMENT DATA pPayment;
                            "<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"..Order-Status..\">"
                                                                                pPayment =
                             "<INPUT TYPE=\"submit\" NAME=\"CMD\"
                                                                      Term.pClientData[iTermId].pTxn->BuffAddr_Payment();
                                                                                ZeroMemory(pPayment, sizeof(PAYMENT DATA));
VALUE=\"..Stock-Level..\">"
                            "<INPUT TYPE=\"submit\" NAME=\"CMD\"
                                                                                pPayment->w_id = Term.pClientData[iTermId].w_id;
VALUE=\"..Exit..\">"
                                                                                GetPaymentData(pECB->lpszQueryString, pPayment);
                            "</BODY></FORM></HTML>"
                                                                                Term.pClientData[iTermId].pTxn->Payment();
                            , pDeliveryData->o carrier id,
                                                                                pPayment =
                            (pDeliveryData->exec_status_code == eOK)?
"Delivery has been queued." : "Delivery Post Failed
                                                                      Term.pClientData[iTermId].pTxn->BuffAddr Payment();
                                                                                MakePaymentForm(iTermId, pPayment, OUTPUT_FORM,
                                                                      szBuffer);
/* FUNCTION: ProcessNewOrderForm
                                                                      /* FUNCTION: ProcessOrderStatusForm
* PURPOSE:
                   This function gets and validates the input data from the
                                                                       * PURPOSE:
                                                                                         This function gets and validates the input data from the
new order form
                                                                      Order Status
```

```
form filling in the required input variables. It
                                                                                   {
then calls the
                                                                                             //post delivery info
                                                                                             if ( PostDeliveryInfo(pDelivery->w_id,
                              SOLOrderStatus transaction, constructs the
output form and writes it
                                                                         pDelivery->o_carrier_id))
                              back to client browser
                                                                                                       pDelivery->exec_status_code =
                                                                         eDeliveryFailed;
* ARGUMENTS:
                   EXTENSION CONTROL BLOCK
                                                            *pECB
                                                                                             else
passed in structure pointer from inetsry.
                                                                                                       pDelivery->exec status code = eOK;
                    iTermId client browser terminal id
                                                                                   else // delivery is done synchronously if no delivery threads
                                                                         configured
                                                                                             Term.pClientData[iTermId].pTxn->Delivery();
void ProcessOrderStatusForm(EXTENSION_CONTROL_BLOCK *pECB, int
                                                                                   pDelivery =
iTermId, char *szBuffer)
                                                                         Term.pClientData[iTermId].pTxn->BuffAddr Delivery();
                                                                                   MakeDeliveryForm(iTermId, pDelivery, OUTPUT_FORM,
          PORDER STATUS DATA
                                        pOrderStatus;
                                                                         szBuffer):
          pOrderStatus =
Term.pClientData[iTermId].pTxn->BuffAddr OrderStatus();
                                                                         /* FUNCTION: ProcessStockLevelForm
          ZeroMemory(pOrderStatus, sizeof(ORDER_STATUS_DATA));
          pOrderStatus->w id = Term.pClientData[iTermId].w id;
                                                                          * PURPOSE:
                                                                                             This function gets and validates the input data from the
          GetOrderStatusData(pECB->lpszQueryString, pOrderStatus);
                                                                         Stock Level
                                                                                                       form filling in the required input variables. It
          Term.pClientData[iTermId].pTxn->OrderStatus();
                                                                         then calls the
                                                                                                       SQLStockLevel transaction, constructs the
          pOrderStatus =
                                                                         output form and writes it
Term.pClientData[iTermId].pTxn->BuffAddr_OrderStatus();
                                                                                                       back to client browser.
          MakeOrderStatusForm(iTermId, pOrderStatus, OUTPUT FORM,
                                                                          * ARGUMENTS:
szBuffer):
                                                                                             EXTENSION CONTROL BLOCK
                                                                                                                                     *pECB
                                                                         passed in structure pointer from inetsry.
/* FUNCTION: ProcessDeliveryForm
                                                                                             iTermId
                                                                                                       client browser terminal id
* PURPOSE:
                   This function gets and validates the input data from the
delivery form
                                                                         void ProcessStockLevelForm(EXTENSION_CONTROL_BLOCK *pECB, int
                              filling in the required input variables. It then
                                                                         iTermId, char *szBuffer)
calls the PostDeliveryInfo
                              Api, The client is then informed that the
                                                                                                       *ptr = pECB->lpszQueryString;
transaction has been posted.
                                                                                   char
                   EXTENSION_CONTROL_BLOCK
* ARGUMENTS:
                                                                                   PSTOCK_LEVEL_DATA
                                                            *pECB
                                                                                                                 pStockLevel;
passed in structure pointer from inetsrv.
                                        int
                                                                                   pStockLevel =
                                                                         Term.pClientData[iTermId].pTxn->BuffAddr StockLevel();
                    iTermId client browser terminal id
                                                                                   ZeroMemory( pStockLevel, sizeof(STOCK_LEVEL_DATA) );
*/
                                                                                   pStockLevel->w id = Term.pClientData[iTermId].w id;
void ProcessDeliveryForm(EXTENSION_CONTROL_BLOCK *pECB, int
                                                                                   pStockLevel->d_id = Term.pClientData[iTermId].d_id;
iTermId, char *szBuffer)
                                                                                   pStockLevel->threshold = GetIntKeyValue(&ptr, "TT*",
                                                                         ERR STOCKLEVEL MISSING THRESHOLD KEY,
          char
                    *ptr = pECB->lpszQueryString;
                                                                         ERR_STOCKLEVEL_THRESHOLD_INVALID);
                                                                                   if ( pStockLevel->threshold >= 100 \parallel pStockLevel->threshold < 0 )
          PDELIVERY DATA pDelivery;
                                                                                             throw new CWEBCLNT ERR(
          pDelivery =
                                                                         ERR_STOCKLEVEL_THRESHOLD_RANGE );
Term.pClientData[iTermId].pTxn->BuffAddr Delivery();
          ZeroMemory(pDelivery, sizeof(DELIVERY_DATA));
                                                                                   Term.pClientData[iTermId].pTxn->StockLevel();
          pDelivery->w id = Term.pClientData[iTermId].w id;
                                                                                   pStockLevel =
          pDelivery->o carrier id
                                        = GetIntKeyValue(&ptr, "OCD*",
                                                                         Term.pClientData[iTermId].pTxn->BuffAddr StockLevel();
ERR_DELIVERY_MISSING_OCD_KEY,
                                                                                   Make Stock Level Form (iTerm Id, pStock Level, OUTPUT\_FORM,
ERR_DELIVERY_CARRIER_INVALID);
                                                                         szBuffer);
          if (pDelivery->o_carrier_id > 10 || pDelivery->o_carrier_id < 1)
                    throw new CWEBCLNT_ERR(
ERR DELIVERY CARRIER ID RANGE);
                                                                         /* FUNCTION: GetNewOrderData
          if (dwNumDeliveryThreads)
```

```
* PURPOSE:
                    This function extracts and validates the new order form
data from an http command string.
                                                                                                         items++;
* ARGUMENTS:
                   LPSTR
                                                  lpszQueryString
                                                                                               else
client browser http command string
                                                                                                         // nothing entered for supply warehouse, so
                                        NEW ORDER DATA
                                                                          item id and qty must also be blank
*pNewOrderData
                              pointer to new order data structure
                                                                                                         GetKeyValue(&ptr, szIID[i], szTmp,
                                                                          sizeof(szTmp), ERR NEWORDER MISSING IID KEY);
                                                                                                         if (szTmp[0])
                                                                                                                   throw new CWEBCLNT ERR(
                                                                          ERR NEWORDER ITEMID WITHOUT SUPPW);
void GetNewOrderData(LPSTR lpszQueryString, NEW ORDER DATA
*pNewOrderData)
                                                                                                         GetKeyValue(&ptr, szQty[i], szTmp,
                                                                          sizeof(szTmp), ERR_NEWORDER_MISSING_QTY_KEY);
          char
                    szTmp[26];
          int
                                                                                                         if (szTmp[0])
          short
                    items;
                                                                                                                   throw new CWEBCLNT ERR(
                                                                          ERR NEWORDER QTY WITHOUT SUPPW);
          int
                              ol i id, ol quantity;
          char
                    *ptr = lpszQueryString;
          static char szSP[MAX OL NEW ORDER ITEMS][6] =
                                                                                     if ( items == 0 )
                    { "SP00*", "SP01*", "SP02*", "SP03*", "SP04*", "SP05*", "SP06*", "SP07*", "SP08*", "SP09*", "SP10*", "SP11*", "SP12*", "SP13*", "SP14*" };
                                                                                               throw new CWEBCLNT ERR(
                                                                          ERR NEWORDER NOITEMS ENTERED);
          static char szIID[MAX OL NEW ORDER ITEMS][7] =
                                                                                     pNewOrderData->o_ol_cnt = items;
                    {"IID00*","IID01*","IID02*","IID03*","IID04*",
"IID05*","IID06*","IID07*","IID08*","IID09*",
"IID11*","IID12*","IID13*","IID14*"};
                                                                          /* FUNCTION: GetPaymentData
          static char szQty[MAX_OL_NEW_ORDER_ITEMS][7] =
                    { "Qty00*", "Qty01*", "Qty02*", "Qty03*", "Qty04*", "Qty05*", "Qty06*", "Qty07*", "Qty08*", "Qty09*", "Qty10*", "Qty11*", "Qty12*", "Qty13*", "Qty14*" };
                                                                           * PURPOSE:
                                                                                               This function extracts and validates the payment form
                                                                          data from an http command string.
                                                                           * ARGUMENTS:
                                                                                               LPSTR
                                                                                                                             lpszQueryString
          pNewOrderData->d id = GetIntKeyValue(&ptr, "DID*",
                                                                          client browser http command string
ERR NEWORDER FORM MISSING DID,
                                                                                                                   PAYMENT DATA
ERR NEWORDER_DISTRICT_INVALID);
                                                                           *pPaymentData
                                                                                                         pointer to payment data structure
          pNewOrderData->c id = GetIntKeyValue(&ptr, "CID*",
ERR NEWORDER CUSTOMER_KEY,
ERR NEWORDER CUSTOMER INVALID);
                                                                           void GetPaymentData(LPSTR lpszQueryString, PAYMENT DATA
                                                                           *pPaymentData)
          for(i=0, items=0; i<MAX OL NEW ORDER ITEMS; i++)
                                                                                               szTmp[26];
                                                                                     char
                    GetKeyValue(&ptr, szSP[i], szTmp, sizeof(szTmp),
                                                                                               *ptr = lpszQueryString;
                                                                                     char
ERR NEWORDER MISSING SUPPW KEY);
                                                                                     BOOL
                                                                                               bCustIdBlank;
                    if(szTmp[0])
                                                                                     pPaymentData->d id = GetIntKeyValue(&ptr, "DID*",
                                                                          ERR_PAYMENT_MISSING_DID_KEY,
                              if ( !IsNumeric(szTmp) )
                                        throw new CWEBCLNT ERR(
                                                                          ERR PAYMENT DISTRICT INVALID);
ERR NEWORDER SUPPW INVALID);
                              pNewOrderData-\!\!>\!\!OL[items].ol\_supply\_w\_id
                                                                                     GetKeyValue(&ptr, "CID*", szTmp, sizeof(szTmp),
                                                                          ERR PAYMENT MISSING CID KEY);
= (short)atoi(szTmp);
                                                                                     if(szTmp[0] == 0)
                              ol i id = pNewOrderData->OL[items].ol i id
                                                                                               bCustIdBlank = TRUE;
                                        GetIntKeyValue(&ptr, szIID[i],
                                                                                               pPaymentData->c id = 0;
ERR NEWORDER MISSING IID KEY,
ERR NEWORDER ITEMID INVALID);
                                                                                    else
                              if (ol i id > 999999 || ol i id < 1)
                                                                                               // parse customer id and verify that last name was NOT
                                        throw new CWEBCLNT_ERR(
                                                                          entered
ERR NEWORDER ITEMID RANGE);
                                                                                               bCustIdBlank = FALSE;
                                                                                               if ( !IsNumeric(szTmp) )
                                                                                                         throw new CWEBCLNT ERR(
                              ol quantity =
pNewOrderData->OL[items].ol_quantity =
                                                                          ERR PAYMENT CUSTOMER INVALID);
                                        GetIntKeyValue(&ptr, szQty[i],
                                                                                               pPaymentData->c_id = atoi(szTmp);
ERR NEWORDER MISSING QTY KEY,
ERR_NEWORDER_QTY_INVALID);
                              if (ol_quantity > 99 \parallel ol_quantity < 1)
                                                                                    pPaymentData->c w id = GetIntKeyValue(&ptr, "CWI*",
                                        throw new CWEBCLNT_ERR(
                                                                          ERR_PAYMENT_MISSING_CWI_KEY,
ERR NEWORDER_QTY_RANGE );
                                                                          ERR PAYMENT CWI INVALID);
```

```
pPaymentData->c d id = GetIntKeyValue(&ptr, "CDI*".
                                                                                          if (strlen(pOrderStatusData->c last) >
ERR PAYMENT MISSING CDI KEY, ERR PAYMENT CDI INVALID);
                                                                      LAST NAME LEN)
                                                                                                    throw new CWEBCLNT ERR(
                                                                       ERR ORDERSTATUS CLT RANGE);
         if (bCustIdBlank)
                   // customer id is blank, so last name must be entered
                                                                                          strcpy(pOrderStatusData->c_last, szTmp);
                   GetKeyValue(&ptr, "CLT*", szTmp, sizeof(szTmp),
ERR PAYMENT MISSING CLT KEY);
                                                                                else
                                                                                          // parse customer id and verify that last name was NOT
                   if(szTmp[0] == 0)
                                                                                {
                             throw new CWEBCLNT ERR(
                                                                       entered
ERR PAYMENT MISSING CID CLT);
                                                                                          if ( !IsNumeric(szTmp) )
                                                                                                    throw new CWEBCLNT ERR(
                                                                       ERR ORDERSTATUS CID INVALID);
                    strupr( szTmp );
                   if (strlen(pPaymentData->c last) > LAST NAME LEN
                                                                                          pOrderStatusData->c id = atoi(szTmp);
                                                                                          GetKeyValue(&ptr, "CLT*", szTmp, sizeof(szTmp),
                             throw new CWEBCLNT ERR(
                                                                       ERR ORDERSTATUS MISSING CLT KEY);
ERR PAYMENT LAST NAME TO LONG);
                                                                                          if (szTmp[0] != 0)
                   strcpy(pPaymentData->c last, szTmp);
                                                                                                    throw new CWEBCLNT ERR(
                                                                       ERR_ORDERSTATUS_CID_AND_CLT );
         else
                   // parse customer id and verify that last name was NOT
entered
                   GetKeyValue(&ptr, "CLT*", szTmp, sizeof(szTmp),
ERR_PAYMENT_MISSING_CLT_KEY);
                                                                       /* FUNCTION: BOOL IsNumeric(char *ptr)
                   if (szTmp[0]!=0)
                             throw new CWEBCLNT ERR(
                                                                       * PURPOSE:
                                                                                          This function determines if a string is numeric. It fails if
ERR_PAYMENT_CID_AND_CLT );
                                                                       any characters other
                                                                                                    than numeric and null terminator are present.
         GetKeyValue(&ptr, "HAM*", szTmp, sizeof(szTmp),
                                                                       * ARGUMENTS:
                                                                                          char
                                                                                                                       *ptr
                                                                                                                                 pointer to
ERR_PAYMENT_MISSING_HAM_KEY);
                                                                       string to check.
         if (!IsDecimal(szTmp))
                   throw new CWEBCLNT_ERR(
                                                                       * RETURNS:
                                                                                                    BOOL
                                                                                                             FALSE
                                                                                                                       if string is not all
ERR PAYMENT HAM INVALID);
                                                                       numeric
         pPaymentData->h amount = atof(szTmp);
                                                                                                                                 TRUE
                                                                                                                                          if
         if (pPaymentData->h_amount >= 10000.00 ||
                                                                       string contains only numeric characters i.e. '0' - '9'
pPaymentData->h amount < 0)
                   throw new CWEBCLNT ERR(
ERR PAYMENT HAM RANGE);
                                                                       BOOL IsNumeric(char *ptr)
                                                                                if (*ptr == 0)
/* FUNCTION: GetOrderStatusData
                                                                                          return FALSE;
* PURPOSE:
                   This function extracts and validates the payment form
                                                                                while( *ptr && isdigit(*ptr) )
data from an http command string.
                                                                                          ptr++;
                                                                                return (!*ptr);
void GetOrderStatusData(LPSTR lpszQueryString, ORDER STATUS DATA
*pOrderStatusData)
                                                                       /* FUNCTION: BOOL IsDecimal(char *ptr)
                                                                       * PURPOSE:
                                                                                          This function determines if a string is a non-negative
         char
                   szTmp[26];
                                                                       decimal value.
         char
                   *ptr = lpszQueryString;
                                                                             It fails if any characters other than a series of numbers followed by
         pOrderStatusData->d_id = GetIntKeyValue(&ptr, "DID*",
                                                                                                    a decimal point, another series of numbers,
ERR ORDERSTATUS MISSING DID KEY,
                                                                       and a null terminator are present.
ERR ORDERSTATUS DID INVALID);
                                                                       * ARGUMENTS:
                                                                                                                       *ptr
                                                                                          char
                                                                                                                                 pointer to
         GetKeyValue(&ptr, "CID*", szTmp, sizeof(szTmp),
                                                                       string to check.
ERR_ORDERSTATUS_MISSING_CID_KEY);
         if(szTmp[0] == 0)
                                                                       * RETURNS:
                                                                                                                      if string is not a valid
                                                                                                    BOOL
                                                                                                             FALSE
                   // customer id is blank, so last name must be entered
                                                                       non-negative decimal value
                   pOrderStatusData->c id = 0;
                                                                                                                                 TRUE
                                                                                                                                          if
                   GetKeyValue(&ptr, "CLT*", szTmp, sizeof(szTmp),
                                                                       string is OK
ERR_ORDERSTATUS_MISSING_CLT_KEY);
                   if(szTmp[0] == 0)
                             throw new CWEBCLNT ERR(
                                                                       BOOL IsDecimal(char *ptr)
ERR ORDERSTATUS MISSING CID CLT);
                                                                                char *dotptr;
                   _strupr( szTmp );
                                                                                BOOL bValid;
```

```
#define TP MAX RETRIES
          if (*ptr == 0)
                                                                                      50
                    return FALSE:
          // find decimal point
                                                                            //note that the welcome form must be processed first as terminal ids assigned
          dotptr = strchr( ptr, '.' );
          if (dotptr == NULL)
                                                                            //terminal id is assigned then the forms can be processed in any order.
                                                                            #define WELCOME_FORM
                    // no decimal point, so just check for numeric
                    return IsNumeric(ptr);
                                                                                                 //beginning form no term id assigned, form id
                                                                            #define MAIN_MENU_FORM
          *dotptr = 0; // temporarily replace decimal with a terminator
                                                                                                 //term id assigned main menu form id
          if (*ptr !=0)
                                                                            #define NEW ORDER FORM
                    bValid = IsNumeric(ptr);
                                                                                                 //new order form id
          // string starts with decimal point
                                                                            #define PAYMENT_FORM
                                                                                                 //payment form id
          else if (*(dotptr+1) == 0)
                    return FALSE; // nothing but a decimal point is bad
                                                                            #define DELIVERY FORM
          else
                                                                                                 //delivery form id
                                                                            #define ORDER STATUS FORM
                    bValid = TRUE;
                                                                                                //order status id
          if (*(dotptr+1) != 0)
                                                                            #define STOCK LEVEL FORM
                    // check text after decimal point
                                                                                                //stock level form id
                    bValid &= IsNumeric(dotptr+1);
                                                                            //This macro is used to prevent the compiler error unused formal parameter
          *dotptr = '.'; // replace decimal point
                                                                            #define UNUSEDPARAM(x) (x = x)
          return bValid;
}
                                                                            //This structure defines the data necessary to keep distinct for each terminal or
                                                                            client connection.
                                                                            typedef struct CLIENTDATA
tpcc.def
                                                                                                                     iNextFree;
                                                                                      int
                                                                            //index of next free element or -1 if this entry in use.
LIBRARY TPCC.DLL
                                                                                                                      w id:
                                                                                      //warehouse id assigned at welcome form
EXPORTS
                                                                                                                     d_id:
                                                                                      //district id assigned at welcome form
          GetExtensionVersion @1
          HttpExtensionProc @2
                                                                                      int
                                                                                                                     iSyncId;
          TerminateExtension @3
                                                                            //syncronization id
                                                                                                                     iTickCount;
                                                                                      int
                                                                                      //time of last access;
tpcc.h
                                                                                      CTPCC BASE
                                                                                                           *pTxn;
          FILE:
                               TPCC.H
                                                                            } CLIENTDATA, *PCLIENTDATA;
                                         Microsoft TPC-C Kit Ver.
4.20.000
                                         Copyright Microsoft, 1999
                                                                            //This structure is used to define the operational interface for terminal id support
         All Rights Reserved
                                                                            typedef struct _TERM
                                         Version 4.10.000 audited by
                                                                                                                     iNumEntries;
                                                                                      int
Richard Gimarc, Performance Metrics, 3/17/99
                                                                                                 //total allocated terminal array entries
                                                                                      int
                                                                                                                     iFreeList;
          PURPOSE:
                               Header file for ISAPI TPCC.DLL, defines
                                                                                                //next available terminal array element or -1 if none
structures and functions used in the isapi tpcc.dll.
                                                                                                                      iMasterSyncId;
                                                                                                //syncronization id
*/
                                                                                      CLIENTDATA
                                                                                                           *pClientData;
                                                                                      //pointer to allocated client data
//VERSION RESOURCE DEFINES
                                                                            } TERM;
#define _APS_NEXT_RESOURCE_VALUE
                                                                            typedef TERM *PTERM;
#define _APS_NEXT_COMMAND_VALUE
                                                                                      //pointer to terminal structure type
#define APS NEXT CONTROL VALUE
                                                                                      enum WEBERROR
#define _APS_NEXT_SYMED_VALUE
                                                                                                 NO ERR,
                                                                                                 ERR COMMAND UNDEFINED,
```

```
ERR D ID INVALID,
                                                                                               m SystemErr = 0;
                  ERR_DELIVERY_CARRIER_ID_RANGE,
                                                                                               m szErrorText = NULL;
                  ERR_DELIVERY_CARRIER_INVALID, ERR_DELIVERY_MISSING_OCD_KEY,
                                                                                      };
                                                                                      CWEBCLNT ERR(WEBERROR Err, char
                  ERR DELIVERY THREAD FAILED,
                  ERR GETPROCADDR FAILED,
                                                                   *szTextDetail, DWORD dwSystemErr)
                  ERR HTML ILL FORMED,
                  ERR_INVALID_SYNC_CONNECTION,
                                                                                               m Error = Err;
                  ERR INVALID TERMID,
                                                                                               m szTextDetail = new
                  ERR LOADDLL FAILED,
                                                                   char[strlen(szTextDetail)+1];
                  ERR_MAX_CONNECTIONS_EXCEEDED,
                                                                                               strcpy( m_szTextDetail, szTextDetail );
                  ERR_MEM_ALLOC_FAILED,
                                                                                               m_SystemErr = dwSystemErr;
                  ERR MISSING REGISTRY ENTRIES,
                                                                                               m szErrorText = NULL;
                  ERR_NEWORDER_CUSTOMER_INVALID,
                                                                                      };
                  ERR NEWORDER CUSTOMER KEY,
                  ERR NEWORDER DISTRICT INVALID,
                                                                                      ~CWEBCLNT ERR()
                  ERR NEWORDER FORM MISSING DID,
                  ERR NEWORDER ITEMID INVALID,
                                                                                               if (m_szTextDetail != NULL)
                  ERR NEWORDER ITEMID RANGE,
                                                                                                        delete [] m szTextDetail;
                  ERR_NEWORDER_ITEMID_WITHOUT_SUPPW,
                                                                                               if (m szErrorText != NULL)
                  ERR_NEWORDER_MISSING_IID_KEY,
                                                                                                        delete [] m_szErrorText;
                  ERR_NEWORDER_MISSING_QTY_KEY,
                                                                                      };
                  ERR NEWORDER MISSING SUPPW KEY,
                  ERR NEWORDER NOITEMS ENTERED,
                                                                                      WEBERROR
                                                                                                        m Error:
                  ERR_NEWORDER_QTY_INVALID,
                                                                                      char
                                                                                                        *m_szTextDetail;
                                                                                                                          //
                  ERR_NEWORDER_QTY_RANGE,
ERR_NEWORDER_QTY_WITHOUT_SUPPW,
ERR_NEWORDER_SUPPW_INVALID,
                                                                                                        *m szErrorText;
                                                                                      char
                                                                                      DWORD
                                                                                                        m SystemErr;
                  ERR NO SERVER SPECIFIED,
                                                                                      int ErrorType() {return ERR TYPE WEBDLL;};
                  ERR_ORDERSTATUS_CID_AND_CLT,
                                                                                      int ErrorNum() {return m_Error;};
                  ERR ORDERSTATUS CID INVALID,
                                                                                      char *ErrorText():
                  ERR_ORDERSTATUS_CLT_RANGE,
                  ERR ORDERSTATUS DID INVALID,
                  ERR ORDERSTATUS MISSING CID CLT,
                  ERR_ORDERSTATUS_MISSING_CID_KEY,
                                                                   //These constants have already been defined in engstut.h, but since we do
                  ERR ORDERSTATUS MISSING CLT KEY,
                                                                   //not want to include it in the delisrv executable
                  ERR ORDERSTATUS MISSING DID KEY,
                                                                            TXN_EVENT_START
                                                                   #define
                  ERR PAYMENT CDI INVALID,
                                                                            TXN EVENT STOP
                                                                   #define
                  ERR_PAYMENT_CID_AND_CLT,
                                                                   #define TXN_EVENT_WARNING
                                                                                                                           //used to
                  ERR_PAYMENT_CUSTOMER_INVALID, ERR_PAYMENT_CWI_INVALID,
                                                                   record a warning into the log
                  ERR PAYMENT DISTRICT_INVALID,
                                                                   //function prototypes
                  ERR PAYMENT HAM INVALID,
                                                                   BOOL APIENTRY DllMain(HANDLE hModule, DWORD ul reason for call,
                  ERR_PAYMENT_HAM_RANGE,
                  ERR_PAYMENT_LAST_NAME_TO_LONG, ERR_PAYMENT_MISSING_CDI_KEY,
                                                                   LPVOID lpReserved);
                                                                   void WriteMessageToEventLog(LPTSTR lpszMsg);
                  ERR PAYMENT MISSING CID CLT,
                                                                   void ProcessQueryString(EXTENSION CONTROL BLOCK *pECB, int
                  ERR PAYMENT MISSING CID KEY,
                                                                   *pCmd, int *pFormId, int *pTermId, int *pSyncId);
                  ERR_PAYMENT_MISSING_CLT,
                                                                   void WelcomeForm(EXTENSION_CONTROL_BLOCK *pECB, char
                  ERR PAYMENT MISSING CLT KEY,
                                                                   *szBuffer);
                  ERR PAYMENT_MISSING_CWI_KEY,
                                                                   void SubmitCmd(EXTENSION CONTROL BLOCK *pECB, char *szBuffer);
                  ERR PAYMENT MISSING DID KEY,
                                                                   void BeginCmd(EXTENSION CONTROL BLOCK *pECB, int iFormId, int
                  ERR_PAYMENT_MISSING_HAM_KEY,
                                                                   iTermId);
                  \begin{array}{l} {\tt ERR\_STOCKLEVEL\_MISSING\_THRESHOLD\_KEY}, \\ {\tt ERR\_STOCKLEVEL\_THRESHOLD\_INVALID}, \end{array}
                                                                   void ProcessCmd(EXTENSION CONTROL BLOCK *pECB, int iFormId, int
                                                                   iTermId);
                  ERR_STOCKLEVEL_THRESHOLD_RANGE,
                                                                   void StatsCmd(EXTENSION_CONTROL_BLOCK *pECB, char *szBuffer);
                  ERR VERSION MISMATCH,
                                                                   void ErrorMessage(EXTENSION CONTROL BLOCK *pECB, int iError, int
                                                                   iErrorType, char *szMsg, int iTermId);
                  ERR_W_ID_INVALID
                                                                   void GetKeyValue(char **pQueryString, char *pKey, char *pValue, int iMax,
         };
                                                                   WEBERROR err);
                                                                   int GetIntKeyValue(char **pQueryString, char *pKey, WEBERROR
class CWEBCLNT ERR: public CBaseErr
                                                                   NoKeyErr, WEBERROR NotIntErr);
                                                                   void TermInit(void);
         public:
                                                                   void TermDeleteAll(void);
                  CWEBCLNT ERR(WEBERROR Err)
                                                                   int TermAdd(void);
                                                                   void TermDelete(int id);
                                                                   void ErrorForm(EXTENSION_CONTROL_BLOCK *pECB, int iType, int
                           m Error = Err:
                           m szTextDetail = NULL;
                                                                   iErrorNum, int iTermId, int iSyncId, char *szErrorText, char *szBuffer );
```

```
void MakeMainMenuForm(int iTermId, int iSyncId, char *szForm);
                                                               #ifdef DEBUG
                                                                FILEFLAGS 0x1L
void MakeStockLevelForm(int iTermId, STOCK LEVEL DATA
*pStockLevelData, BOOL bInput, char *szForm);
                                                               #else
void MakeNewOrderForm(int iTermId, NEW_ORDER_DATA
                                                                FILEFLAGS 0x0L
*pNewOrderData, BOOL bInput, char *szForm);
                                                               #endif
void MakePaymentForm(int iTermId, PAYMENT DATA *pPaymentData,
                                                                FILEOS 0x40004L
BOOL bInput, char *szForm);
                                                                FILETYPE 0x2L
void MakeOrderStatusForm(int iTermId, ORDER STATUS DATA
                                                                FILESUBTYPE 0x0L
*pOrderStatusData, BOOL bInput, char *szForm);
                                                               BEGIN
void MakeDeliveryForm(int iTermId, DELIVERY DATA *pDeliveryData,
                                                                 BLOCK "StringFileInfo"
BOOL bInput, char *szForm);
                                                                 BEGIN
void ProcessNewOrderForm(EXTENSION_CONTROL_BLOCK *pECB, int
                                                                   BLOCK "040904b0"
iTermId, char *szBuffer);
                                                                   BEGIN
void ProcessPaymentForm(EXTENSION_CONTROL_BLOCK *pECB, int
                                                                     VALUE "Comments", "TPC-C HTML DLL Server (DBLIB)\0"
iTermId, char *szBuffer);
                                                                     VALUE "CompanyName", "Microsoft\0"
                                                                     VALUE "FileDescription", "TPC-C HTML DLL Server (DBLIB)\0"
void ProcessOrderStatusForm(EXTENSION CONTROL BLOCK *pECB, int
                                                                     VALUE "FileVersion", "0, 4, 0, 0\0" VALUE "InternalName", "tpcc\0"
iTermId, char *szBuffer);
void ProcessDeliveryForm(EXTENSION_CONTROL_BLOCK *pECB, int
iTermId. char *szBuffer):
                                                                     VALUE "LegalCopyright", "Copyright © 1997\0"
                                                                     VALUE "OriginalFilename", "tpcc.dll\0"
void ProcessStockLevelForm(EXTENSION CONTROL BLOCK *pECB, int
iTermId, char *szBuffer);
                                                                     VALUE "ProductName", "Microsoft tpcc\0"
void GetNewOrderData(LPSTR lpszQueryString, NEW ORDER DATA
                                                                     VALUE "ProductVersion", "0, 4, 0, 0\0"
*pNewOrderData);
                                                                   END
void GetPaymentData(LPSTR lpszQueryString, PAYMENT DATA
                                                                 END
*pPaymentData);
                                                                 BLOCK "VarFileInfo"
void GetOrderStatusData(LPSTR lpszQueryString, ORDER STATUS DATA
                                                                 BEGIN
*pOrderStatusData):
                                                                    VALUE "Translation", 0x409, 1200
BOOL PostDeliveryInfo(short w_id, short o_carrier_id);
                                                                 END
BOOL IsNumeric(char *ptr);
                                                               END
BOOL IsDecimal(char *ptr);
void DeliveryWorkerThread(void *ptr);
                                                               #endif //! MAC
                                                               #ifdef APSTUDIO INVOKED
tpcc.rc
                                                               .......
//Microsoft Developer Studio generated resource script.
                                                               // TEXTINCLUDE
#include "resource.h"
                                                               1 TEXTINCLUDE DISCARDABLE
#define APSTUDIO_READONLY_SYMBOLS
                                                               BEGIN
"resource.h\0"
                                                               END
// Generated from the TEXTINCLUDE 2 resource.
                                                               2 TEXTINCLUDE DISCARDABLE
#include "afxres.h"
                                                               BEGIN
                                                                 "#include ""afxres.h""\r\n"
"\0"
#undef APSTUDIO READONLY SYMBOLS
                                                               END
3 TEXTINCLUDE DISCARDABLE
// English (U.S.) resources
                                                               BEGIN
                                                                  "\r\n'
#if !defined(AFX_RESOURCE_DLL) || defined(AFX_TARG_ENU)
                                                                 "\0"
#ifdef WIN32
                                                               END
LANGUAGE LANG ENGLISH, SUBLANG ENGLISH US
#pragma code_page(1252)
                                                               #endif // APSTUDIO INVOKED
#endif //_WIN32
#ifndef MAC
                                                               //
                                                               // Dialog
// Version
//
                                                               IDD DIALOG1 DIALOG DISCARDABLE 0, 0, 186, 95
VS VERSION INFO VERSIONINFO
                                                               STYLE DS MODALFRAME | WS POPUP | WS CAPTION |
FILEVERSION 0,4,0,0
                                                               WS_SYSMENU
PRODUCTVERSION 0,4,0,0
                                                               CAPTION "Dialog"
FILEFLAGSMASK 0x3fL
```

```
FONT 8, "MS Sans Serif"
                                                                  #include "..\..\common\src\trans.h"
                                                                                                               //tpckit transaction
BEGIN
                                                                  header contains definations of structures specific to TPC-C
 DEFPUSHBUTTON "OK", IDOK, 129, 7, 50, 14
                                                                  #include "..\..\common\src\error.h"
                                                                 #include "..\..\common\src\txn_base.h"
#include "tpcc_com.h"
 PUSHBUTTON "Cancel", IDCANCEL, 129, 24, 50, 14
                                                                  #include "..\..\tpcc_com_ps\src\tpcc_com_ps_i.c"
                                                                  #include "..\..\tpcc_com_all\src\tpcc_com_all_i.c"
// wrapper routine for class constructor
// DESIGNINFO
                                                                   _declspec(dllexport) CTPCC_COM* CTPCC_COM_new(BOOL
//
                                                                  bSinglePool)
#ifdef APSTUDIO INVOKED
GUIDELINES DESIGNINFO DISCARDABLE
                                                                           return new CTPCC_COM(bSinglePool);
BEGIN
 IDD_DIALOG1, DIALOG
                                                                  CTPCC COM::CTPCC COM(BOOL bSinglePool)
 BEGIN
   LEFTMARGIN, 7
   RIGHTMARGIN, 179
                                                                           HRESULT hr = NULL;
   TOPMARGIN, 7
                                                                           long lRet = 0;
                                                                          ULONG ulTmpSize = 0;
   BOTTOMMARGIN, 88
 END
END
                                                                           m pTxn
                                                                                                     = NULL;
#endif // APSTUDIO_INVOKED
                                                                          m pNewOrder
                                                                                                     = NULL;
                                                                           m_pPayment
                                                                                                     = NULL;
#endif // English (U.S.) resources
                                                                          m_pStockLevel
                                                                                            = NULL:
m pOrderStatus
                                                                                            = NULL;
                                                                           m bSinglePool
                                                                                            = bSinglePool;
                                                                           ulTmpSize = (ULONG) sizeof(COM_DATA);
#ifndef APSTUDIO INVOKED
VariantInit(&m_vTxn);
                                                                          m_vTxn.vt = VT_SAFEARRAY;
// Generated from the TEXTINCLUDE 3 resource.
                                                                           m_vTxn.parray = SafeArrayCreateVector(VT_UI1, ulTmpSize,
                                                                  ulTmpSize);
                                                                           if (!m_vTxn.parray)
throw new CCOMERR( E FAIL );
#endif // not APSTUDIO_INVOKED
                                                                           memset((void*)m\_vTxn.parray->pvData,0,ulTmpSize);
                                                                           m pTxn = (COM DATA*)m vTxn.parray->pvData;
tpcc_com.cpp
                                                                           hr = CoInitializeEx(NULL, COINIT MULTITHREADED);
                                                                          if (FAILED(hr))
        FILE:
                          TPCC COM.CPP
                                                                           {
                                    Microsoft TPC-C Kit Ver.
                                                                                   throw new CCOMERR( hr );
4.20.000
                                    Copyright Microsoft, 1999
        All Rights Reserved
                                                                          // create components
                                                                           if (m bSinglePool)
                                   not yet audited
                                                                                   hr = CoCreateInstance(CLSID TPCC, NULL,
        PURPOSE:
                          Source file for TPC-C COM+ class
                                                                  CLSCTX_SERVER, IID_ITPCC, (void **)&m_pNewOrder);
implementation.
                                                                                   if (FAILED(hr))
                 Charles Levine (clevine@microsoft.com)
        Contact:
                                                                                             throw new CCOMERR(hr);
  Change history:
                                                                                   // all txns will use same component
                 4.20.000 - first version
                                                                                   m_pPayment = m_pNewOrder;
                                                                                   m pStockLevel = m pNewOrder;
                                                                                   m_pOrderStatus = m_pNewOrder;
// needed for CoinitializeEx
```

else {

// use different components for each txn

CLSCTX_SERVER, IID_ITPCC, (void **)&m_pNewOrder); if (FAILED(hr))

hr = CoCreateInstance(CLSID NewOrder, NULL,

#define _WIN32_WINNT 0x0400

// need to declare functions for export

#define DllDecl __declspec(dllexport)

#include <windows.h>

```
throw new CCOMERR(hr);
                                                                       void CTPCC COM::Payment()
                   hr = CoCreateInstance(CLSID Payment, NULL,
CLSCTX_SERVER, IID_ITPCC, (void **)&m_pPayment);
                   if (FAILED(hr))
                                                                                 VARIANT vTxn_out;
                             throw new CCOMERR(hr);
                                                                                 HRESULT hr = m pPayment->Payment(m vTxn, &vTxn out);
                   hr = CoCreateInstance(CLSID StockLevel, NULL,
                                                                                 if (FAILED(hr))
CLSCTX_SERVER, IID_ITPCC, (void **)&m_pStockLevel);
                                                                                           throw new CCOMERR( hr );
                   if (FAILED(hr))
                                                                                 memcpy(m pTxn, (void
                             throw new CCOMERR(hr);
                                                                        *)vTxn_out.parray->pvData,vTxn_out.parray->rgsabound[0].cElements);
                                                                                 SafeArrayDestroy(vTxn_out.parray);
                   hr = CoCreateInstance(CLSID OrderStatus, NULL,
CLSCTX_SERVER, IID_ITPCC, (void **)&m_pOrderStatus);
                                                                                 if ( m_pTxn->ErrorType != ERR_SUCCESS )
                   if (FAILED(hr))
                                                                                           throw new CCOMERR( m_pTxn->ErrorType,
                             throw new CCOMERR(hr);
                                                                       m_pTxn->error );
          // call setcomplete to release each component back into pool
                                                                       void CTPCC_COM::StockLevel()
          hr = m pNewOrder->CallSetComplete();
         if (FAILED(hr))
                                                                                 VARIANT vTxn_out;
                   throw new CCOMERR(hr);
                                                                                 HRESULT hr = m_pStockLevel->StockLevel(m_vTxn,
                                                                       &vTxn_out);
         if (!m bSinglePool)
                                                                                 if (FAILED(hr))
                   hr = m_pPayment->CallSetComplete();
                                                                                           throw new CCOMERR( hr );
                   if (FAILED(hr))
                                                                                 memcpy(m pTxn, (void
                                                                        *)vTxn_out.parray->pvData,vTxn_out.parray->rgsabound[0].cElements);
                             throw new CCOMERR(hr);
                                                                                 SafeArrayDestroy(vTxn out.parray);
                   hr = m_pStockLevel->CallSetComplete();
                                                                                 if ( m_pTxn->ErrorType != ERR SUCCESS )
                   if (FAILED(hr))
                                                                                           throw new CCOMERR( m_pTxn->ErrorType,
                             throw new CCOMERR(hr);
                                                                       m_pTxn->error );
                   hr = m pOrderStatus->CallSetComplete();
                   if (FAILED(hr))
                             throw new CCOMERR(hr);
                                                                        void CTPCC_COM::OrderStatus()
          }
                                                                                 VARIANT vTxn out;
}
CTPCC_COM::~CTPCC_COM()
                                                                                 HRESULT hr = m_pOrderStatus->OrderStatus(m_vTxn,
                                                                       &vTxn out);
                                                                                 if (FAILED(hr))
          if (m_pTxn)
                   SafeArrayDestroy(m vTxn.parray);
                                                                                           throw new CCOMERR( hr );
                                                                                 memcpy(m_pTxn, (void
                                                                        *)vTxn out.parray->pvData,vTxn out.parray->rgsabound[0].cElements);
          ReleaseInterface(m pNewOrder);
          if (!m_bSinglePool)
                                                                                 SafeArrayDestroy(vTxn_out.parray);
                                                                                 if ( m_pTxn->ErrorType != ERR_SUCCESS )
                   ReleaseInterface(m pPayment);
                                                                                           throw new CCOMERR( m_pTxn->ErrorType,
                   ReleaseInterface(m_pStockLevel);
                   ReleaseInterface(m pOrderStatus);
                                                                       m pTxn->error);
          CoUninitialize();
                                                                        tpcc com.h
void CTPCC_COM::NewOrder()
                                                                                 FILE:
                                                                                                     TPCC COM.H
          VARIANT vTxn_out;
                                                                                                               Microsoft TPC-C Kit Ver.
                                                                       4.20.000
         HRESULT hr = m pNewOrder->NewOrder(m_vTxn, &vTxn_out);
                                                                                                              Copyright Microsoft, 1999
          if (FAILED(hr))
                                                                                 All Rights Reserved
                   throw new CCOMERR( hr );
         memcpy(m_pTxn, (void
                                                                                                              not yet audited
*)vTxn_out.parray->pvData,vTxn_out.parray->rgsabound[0].cElements);
          SafeArrayDestroy(vTxn_out.parray);
                                                                                 PURPOSE:
                                                                                                     Header file for TPC-C COM+ class
                                                                        implementation.
         if ( m_pTxn->ErrorType != ERR SUCCESS )
                   throw new CCOMERR( m_pTxn->ErrorType,
                                                                          Change history:
m_pTxn->error );
                                                                                           4.20.000 - first version
```

```
*/
                                                                                             BOOL m_bSinglePool;
                                                                                             // COM Interface pointers
#pragma once
                                                                                             ITPCC*
                                                                                                                           m_pNewOrder;
#include <stdio.h>
                                                                                             ITPCC*
                                                                                                                           m_pPayment;
#include "..\..\tpcc com ps\src\tpcc com ps.h"
                                                                                             ITPCC*
                                                                                                                           m pStockLevel;
                                                                                             ITPCC*
                                                                                                                           m pOrderStatus;
// need to declare functions for import, unless define has already been created
// by the DLL's .cpp module for export.
                                                                                             struct COM_DATA
#ifndef DllDecl
#define DllDecl __declspec( dllimport )
                                                                                                       int ErrorType;
#endif
                                                                                                       int error;
                                                                                                       union
class CCOMERR: public CBaseErr
                                                                                                                 NEW_ORDER_DATA
          private:
                                                                         NewOrder;
                    char m szErrorText[64];
                                                                                                                 PAYMENT_DATA
                                                                         Payment;
          public:
                                                                                                                 DELIVERY_DATA
                    // use this interface for genuine COM errors
                                                                         Delivery;
                    CCOMERR( HRESULT hr )
                                                                                                                 STOCK_LEVEL_DATA
                                                                         StockLevel;
                    {
                              m hr = hr;
                                                                                                                 ORDER_STATUS_DATA
                              m iErrorType = 0;
                                                                         OrderStatus;
                              m_iError = 0;
                                                                                                       } u;
                                                                                             } *m_pTxn;
                    // use this interface to impersonate a non-COM error type
                                                                                             VARIANT m_vTxn;
                    CCOMERR( int iErrorType, int iError )
                                                                                   public:
                                                                                             CTPCC_COM(BOOL bSinglePool);
                              m iErrorType = iErrorType;
                                                                                             ~CTPCC COM(void);
                              m_iError = iError;
                              m_hr = S_OK;
                                                                                             inline PNEW_ORDER_DATA
                                                                         BuffAddr NewOrder()
                                                                                                                  { return &m pTxn->u.NewOrder;
                    }
                                                                                             inline PPAYMENT DATA
                    int
                                        m hr;
                                                                         BuffAddr_Payment()
                                                                                                       { return &m_pTxn->u.Payment; };
                    int
                                        m_iErrorType;
                                                                                             inline PDELIVERY DATA
                                        m iError;
                    int
                                                                         BuffAddr_Delivery()
                                                                                                       { return &m_pTxn->u.Delivery; };
                    // A CCOMERR class can impersonate another class,
                                                                                             inline PSTOCK_LEVEL_DATA BuffAddr_StockLevel()
which happens if the error
                                                                          { return &m pTxn->u.StockLevel; };
                                                                                             inline PORDER_STATUS_DATA
                    // was not actually a COM Services error, but was simply
transmitted back via COM.
                                                                         BuffAddr OrderStatus()
                                                                                                       { return &m_pTxn->u.OrderStatus; };
                    int ErrorType()
                                                                                             void NewOrder
                                                                                                                           ();
                              if (m_iErrorType == 0)
                                                                                             void Payment
                                                                                                                           ();
                                        return ERR TYPE COM;
                                                                                             void StockLevel
                                                                                                                           ():
                              else
                                                                                             void OrderStatus
                                                                                             void Delivery
                                        return m_iErrorType;
                                                                                                                           () { throw new
                                                                         CCOMERR(E_NOTIMPL); } // not supported
                    int ErrorNum() {return m hr;}
                    char *ErrorText()
                                                                         inline void ReleaseInterface(IUnknown *pUnk)
                              if (m_hr == S OK)
                                                                                   if (pUnk)
                                        sprintf( m szErrorText, "Error:
                                                                                   {
Class %d, error # %d", m_iErrorType, m_iError);
                                                                                             pUnk->Release();
                                                                                             pUnk = NULL;
                                        sprintf( m_szErrorText, "Error:
COM HRESULT %x", m_hr );
                              return m_szErrorText;
                    }
                                                                         // wrapper routine for class constructor
                                                                         extern "C" __declspec(dllexport) CTPCC_COM* CTPCC_COM_new(BOOL);
class DllDecl CTPCC COM: public CTPCC BASE
                                                                         typedef CTPCC_COM* (TYPE_CTPCC_COM)(BOOL);
          private:
```

```
comdlg32.lib advapi32.lib shell32.lib ole32.lib oleaut32.lib uuid.lib odbc32.lib
tpcc_com_all.dsp
                                                                         odbccp32.lib /nologo /subsystem:windows /dll /machine:I386
# Microsoft Developer Studio Project File - Name="tpcc com all" - Package
                                                                         !ELSEIF "$(CFG)" == "tpcc_com_all - Win32 Debug"
# Microsoft Developer Studio Generated Build File, Format Version 6.00
                                                                         # PROP BASE Use MFC 0
# ** DO NOT EDIT **
                                                                         # PROP BASE Use Debug Libraries 1
                                                                         # PROP BASE Output Dir "Debug"
# TARGTYPE "Win32 (x86) Dynamic-Link Library" 0x0102
                                                                         # PROP BASE Intermediate Dir "Debug"
                                                                         # PROP BASE Target Dir ""
CFG=tpcc com all - Win32 Debug
                                                                         # PROP Use_MFC 0
!MESSAGE This is not a valid makefile. To build this project using NMAKE,
                                                                         # PROP Use_Debug_Libraries 1
!MESSAGE use the Export Makefile command and run
                                                                         # PROP Output Dir ".\bin"
!MESSAGE
                                                                         # PROP Intermediate_Dir ".\obj"
!MESSAGE NMAKE /f "tpcc com all.mak".
                                                                         # PROP Ignore_Export_Lib 0
!MESSAGE
                                                                         # PROP Target Dir '
!MESSAGE You can specify a configuration when running NMAKE
                                                                         # ADD BASE CPP /nologo /MTd /W3 /Gm /GX /Zi /Od /D "WIN32" /D
!MESSAGE by defining the macro CFG on the command line. For example:
                                                                          " DEBUG" /D "_WINDOWS" /YX /FD /c
                                                                         #ADD CPP /nologo /MTd /W3 /Gm /GX /ZI /Od /D "WIN32" /D "_DEBUG"
!MESSAGE\ NMAKE\ /f\ "tpcc\_com\_all.mak"\ CFG="tpcc\_com\_all\ -\ Win 32
                                                                         /D " WINDOWS" /YX /FD /c
Debug"
                                                                         # ADD BASE MTL /nologo /D "_DEBUG" /mktyplib203 /o "NUL" /win32
!MESSAGE
                                                                         \# ADD MTL /nologo /D "_DEBUG" /mktyplib203 /o "NUL" /win32 \# ADD BASE RSC /l 0x409 /d "_DEBUG"
!MESSAGE Possible choices for configuration are:
!MESSAGE
                                                                         # ADD RSC /l 0x409 /d " DEBUG"
!MESSAGE "tpcc_com_all - Win32 Release" (based on "Win32 (x86)
                                                                         BSC32=bscmake.exe
Dynamic-Link Library")
                                                                         # ADD BASE BSC32 /nologo
!MESSAGE "tpcc_com_all - Win32 Debug" (based on "Win32 (x86)
                                                                         # ADD BSC32 /nologo
Dynamic-Link Library")
                                                                         LINK32=link.exe
!MESSAGE
                                                                         # ADD BASE LINK32 kernel32.lib user32.lib gdi32.lib winspool.lib
                                                                         comdlg32.lib advapi32.lib shell32.lib ole32.lib oleaut32.lib uuid.lib odbc32.lib
# Begin Project
                                                                         odbccp32.lib /nologo /subsystem:windows /dll /debug /machine:I386
# PROP AllowPerConfigDependencies 0
                                                                         /pdbtype:sept
# PROP Scc ProjName '
                                                                         # ADD LINK32 ..\db dblib dll\bin\tpcc dblib.lib
# PROP Scc_LocalPath ""
                                                                         ..\db odbc dll\bin\tpcc odbc.lib kernel32.lib user32.lib gdi32.lib winspool.lib
CPP=cl.exe
                                                                         comdlg32.lib advapi32.lib shell32.lib ole32.lib oleaut32.lib uuid.lib odbc32.lib
MTL=midl.exe
                                                                          odbccp32.lib /nologo /subsystem:windows /dll /debug /machine:I386
RSC=rc.exe
                                                                         /pdbtype:sept
!IF "$(CFG)" == "tpcc_com_all - Win32 Release"
                                                                         !ENDIF
# PROP BASE Use_MFC 0
                                                                         # Begin Target
# PROP BASE Use Debug Libraries 0
# PROP BASE Output Dir "Release"
                                                                         # Name "tpcc com all - Win32 Release"
# PROP BASE Intermediate Dir "Release"
                                                                         # Name "tpcc_com_all - Win32 Debug"
# PROP BASE Target_Dir ""
                                                                         # Begin Group "Source"
# PROP Use MFC 0
# PROP Use_Debug_Libraries 0
                                                                         # PROP Default_Filter "*.cpp, *.c"
# PROP Output_Dir ".\bin"
                                                                         # Begin Source File
# PROP Intermediate Dir ".\obi"
# PROP Ignore Export Lib 0
                                                                         SOURCE=.\src\tpcc com all.cpp
# PROP Target Dir "
                                                                         # SUBTRACT CPP /YX
# ADD BASE CPP /nologo /MT /W3 /GX /O2 /D "WIN32" /D "NDEBUG" /D
                                                                         # End Source File
" WINDOWS" /YX /FD /c
                                                                         # Begin Source File
# ADD CPP /nologo /MT /W3 /GX /O2 /D "WIN32" /D "NDEBUG" /D
" WINDOWS" /YX /FD /c
                                                                         SOURCE=.\src\tpcc com all.def
# ADD BASE MTL /nologo /D "NDEBUG" /mktyplib203 /o "NUL" /win32
                                                                         # End Source File
# ADD MTL /nologo /D "NDEBUG" /mktyplib203 /o "NUL" /win32
                                                                         # Begin Source File
# ADD BASE RSC /I 0x409 /d "NDEBUG"
# ADD RSC /I 0x409 /d "NDEBUG"
                                                                         SOURCE=.\src\tpcc com all.idl
BSC32=bscmake.exe
# ADD BASE BSC32 /nologo
                                                                         !IF "$(CFG)" == "tpcc com all - Win32 Release"
```

PROP Ignore Default Tool 1

InputPath=.\src\tpcc_com_all.idl

BuildCmds=\

Begin Custom Build - Performing MIDL step

ADD BSC32 /nologo LINK32=link.exe

ADD BASE LINK32 kernel32.lib user32.lib gdi32.lib winspool.lib

odbccp32.lib /nologo /subsystem:windows /dll /machine:1386

ADD LINK32 ..\db dblib dll\bin\tpcc dblib.lib

comdlg32.lib advapi32.lib shell32.lib ole32.lib oleaut32.lib uuid.lib odbc32.lib

..\db_odbc_dll\bin\tpcc_odbc.lib kernel32.lib user32.lib gdi32.lib winspool.lib

```
PRIVATE
          midl /Oicf /h "tpcc_com_all.h" /iid "tpcc_com_all_i.c"
                                                                                    DllUnregisterServer
                                                                                                                  (a),5
".\src\tpcc com all.idl" /out ".\src"
".\src\tpcc_com_all.tlb": $(SOURCE) "$(INTDIR)" "$(OUTDIR)"
 $(BuildCmds)
                                                                          tpcc com ps.h
".\src\tpcc com all.h": $(SOURCE) "$(INTDIR)" "$(OUTDIR)"
 $(BuildCmds)
                                                                          #pragma warning( disable: 4049 ) /* more than 64k source lines */
".\src\tpcc com all i.c": $(SOURCE) "$(INTDIR)" "$(OUTDIR)"
 $(BuildCmds)
                                                                          /* this ALWAYS GENERATED file contains the definitions for the interfaces
# End Custom Build
!ELSEIF "$(CFG)" == "tpcc_com_all - Win32 Debug"
                                                                          /* File created by MIDL compiler version 5.03.0280 */
# PROP Ignore Default Tool 1
                                                                          /* at Sat Apr 08 16:40:10 2000
# Begin Custom Build - Performing MIDL step
InputPath=.\src\tpcc_com_all.idl
                                                                          /* Compiler settings for .\src\tpcc_com_ps.idl:
                                                                            Oicf (OptLev=i2), W1, Zp8, env=Win32 (32b run), ms ext, c ext
BuildCmds=\
                                                                            error checks: allocation ref bounds_check enum stub_data
          midl /Oicf /h "tpcc_com_all.h" /iid "tpcc_com_all_i.c"
                                                                            VC declspec() decoration level:
".\src\tpcc com all.idl" /out ".\src"
                                                                                 declspec(uuid()), declspec(selectany), declspec(novtable)
                                                                               DECLSPEC_UUID(), MIDL_INTERFACE()
".\src\tpcc com all.tlb": $(SOURCE) "$(INTDIR)" "$(OUTDIR)"
 $(BuildCmds)
                                                                          //@@MIDL FILE HEADING()
".\src\tpcc com all.h": $(SOURCE) "$(INTDIR)" "$(OUTDIR)"
 $(BuildCmds)
                                                                          /* verify that the <rpcndr.h> version is high enough to compile this file*/
                                                                          #ifndef REQUIRED RPCNDR H VERSION
".\src\tpcc_com_all_i.c": $(SOURCE) "$(INTDIR)" "$(OUTDIR)"
                                                                          #define REQUIRED_RPCNDR_H_VERSION 440
 $(BuildCmds)
                                                                          #endif
# End Custom Build
                                                                          #include "rpc.h"
!ENDIF
                                                                          #include "rpcndr.h"
# End Source File
                                                                          #ifndef RPCNDR H VERSION
# End Group
                                                                          #error this stub requires an updated version of <rpcndr.h>
# Begin Group "Header"
                                                                          #endif // __RPCNDR_H_VERSION__
# PROP Default Filter "*.h"
                                                                          #ifndef COM NO WINDOWS H
# Begin Source File
                                                                          #include "windows.h"
                                                                          #include "ole2.h"
SOURCE = . \label{eq:source} SOURCE = . \label{eq:source}
                                                                          #endif/*COM NO WINDOWS H*/
# End Source File
# Begin Source File
                                                                          #ifndef tpcc com ps h
                                                                          #define __tpcc_com_ps_h__
SOURCE=.\src\resource.h
# End Source File
                                                                          /* Forward Declarations */
# End Group
# Begin Source File
                                                                          #ifndef __ITPCC_FWD_DEFINED_
                                                                          #define __ITPCC_FWD_DEFINED_
SOURCE=.\src\tpcc com all.rc
                                                                          typedef interface ITPCC ITPCC;
# End Source File
                                                                                   /* ITPCC FWD DEFINED */
                                                                          #endif
# End Target
# End Project
                                                                          /* header files for imported files */
                                                                          #include "oaidl.h"
                                                                          #include "ocidl.h"
tpcc com ps.def
                                                                          #ifdef __cplusplus
LIBRARY
             "tpcc_com_ps"
                                                                          extern "C" {
                                                                          #endif
DESCRIPTION 'Proxy/Stub DLL'
                                                                          void RPC FAR * RPC USER MIDL user allocate(size t);
EXPORTS
                                                                          void __RPC_USER MIDL_user_free( void __RPC_FAR * );
          DllGetClassObject
                              @1
                                        PRIVATE
          DllCanUnloadNow
                                        PRIVATE
                                @2
                                                                          /* interface __MIDL_itf_tpcc_com_ps_0000 */
          GetProxyDllInfo
                              @3
                                        PRIVATE
                                                                          /* [local] */
                                                  PRIVATE
          DllRegisterServer
                                        (a)4
```

```
ITPCC __RPC_FAR * This,
                                                                              /* [in] */VARIANT txn_in,
                                                                              /* [out] */ VARIANT __RPC_FAR *txn_out);
extern RPC IF HANDLE MIDL itf tpcc com ps 0000 v0 0 c ifspec;
extern RPC IF HANDLE MIDL itf tpcc com ps 0000 v0 0 s ifspec;
                                                                            HRESULT ( stdcall RPC FAR *Delivery )(
                                                                              ITPCC __RPC_FAR * This,
#ifndef ITPCC INTERFACE DEFINED
                                                                              /* [in] */ VARIANT txn in,
#define ITPCC INTERFACE DEFINED
                                                                              /* [out] */ VARIANT RPC FAR *txn out);
/* interface ITPCC */
                                                                            HRESULT ( __stdcall __RPC_FAR *StockLevel )(
                                                                              ITPCC __RPC_FAR * This,
/* [unique][helpstring][uuid][oleautomation][object] */
                                                                              /* [in] */ VARIANT txn in,
                                                                              /* [out] */ VARIANT __RPC_FAR *txn_out);
EXTERN C const IID IID ITPCC;
                                                                            HRESULT ( __stdcall __RPC_FAR *OrderStatus )(
                                                                              ITPCC __RPC_FAR * This, /* [in] */ VARIANT txn_in,
#if defined( cplusplus) && !defined(CINTERFACE)
  MIDL INTERFACE("FEEE6AA2-84B1-11d2-BA47-00C04FBFE08B")
                                                                              /* [out] */ VARIANT __RPC_FAR *txn_out);
  ITPCC: public IUnknown
                                                                            HRESULT ( __stdcall __RPC_FAR *CallSetComplete )(
                                                                              ITPCC __RPC_FAR * This);
  public:
    virtual HRESULT __stdcall NewOrder(
      /* [in] */ VARIANT txn in,
                                                                            END INTERFACE
      /* [out] */ VARIANT __RPC_FAR *txn_out) = 0;
                                                                          } ITPCCVtbl;
    virtual HRESULT stdcall Payment(
                                                                          interface ITPCC
      /* [in] */ VARIANT txn_in,
      /* [out] */ VARIANT RPC FAR *txn out) = 0;
                                                                            CONST VTBL struct ITPCCVtbl RPC FAR *lpVtbl;
    virtual HRESULT __stdcall Delivery(
      /* [in] */ VARIANT txn_in,
      /* [out] */ VARIANT __RPC_FAR *txn_out) = 0;
                                                                        #ifdef COBJMACROS
    virtual HRESULT __stdcall StockLevel(
      /* [in] */ VARIANT txn in,
      /* [out] */ VARIANT __RPC_FAR *txn_out) = 0;
                                                                        #define ITPCC_QueryInterface(This,riid,ppvObject)
                                                                          (This)->lpVtbl -> QueryInterface(This,riid,ppvObject)
    virtual HRESULT __stdcall OrderStatus(
      /* [in] */ VARIANT txn in,
                                                                        #define ITPCC AddRef(This)
      /* [out] */ VARIANT RPC FAR *txn out) = 0;
                                                                          (This)->lpVtbl -> AddRef(This)
    virtual HRESULT stdcall CallSetComplete(void) = 0;
                                                                        #define ITPCC Release(This)
                                                                          (This)->lpVtbl -> Release(This)
  };
#else
         /* C style interface */
                                                                        #define ITPCC NewOrder(This,txn in,txn out)
                                                                          (This)->lpVtbl -> NewOrder(This,txn in,txn out)
  typedef struct ITPCCVtbl
                                                                        #define ITPCC Payment(This,txn in,txn out)
                                                                          (This)->lpVtbl -> Payment(This,txn_in,txn_out)
    BEGIN_INTERFACE
    HRESULT (STDMETHODCALLTYPE __RPC_FAR *QueryInterface)(
                                                                        #define ITPCC_Delivery(This,txn_in,txn_out)
      ITPCC __RPC_FAR * This,
                                                                          (This)->lpVtbl -> Delivery(This,txn in,txn out)
      /* [in] */ REFIID riid,
      /* [iid_is][out] */ void __RPC_FAR *__RPC_FAR *ppvObject);
                                                                        #define ITPCC_StockLevel(This,txn_in,txn_out)
                                                                          (This)->lpVtbl -> StockLevel(This,txn in,txn out)
    ULONG (STDMETHODCALLTYPE __RPC_FAR *AddRef)(
      ITPCC RPC FAR * This);
                                                                        #define ITPCC OrderStatus(This,txn in,txn out)
                                                                          (This)->lpVtbl -> OrderStatus(This,txn_in,txn_out)
    ULONG (STDMETHODCALLTYPE RPC FAR *Release)(
      ITPCC RPC FAR * This);
                                                                        #define ITPCC CallSetComplete(This)
                                                                          (This)->lpVtbl -> CallSetComplete(This)
    HRESULT (__stdcall __RPC_FAR *NewOrder )(
      ITPCC RPC FAR * This,
                                                                        #endif /* COBJMACROS */
      /* [in] */ VARIANT txn in,
      /* [out] */ VARIANT __RPC_FAR *txn_out);
                                                                        #endif
                                                                                 /* C style interface */
```

HRESULT (stdcall RPC FAR *Payment)(

```
HRESULT __stdcall ITPCC_NewOrder_Proxy(
  ITPCC RPC FAR * This,
  /* [in] */ VARIANT txn in,
  /* [out] */ VARIANT RPC FAR *txn out);
void RPC STUB ITPCC NewOrder Stub(
  IRpcStubBuffer *This,
  IRpc Channel Buffer * \_pRpc Channel Buffer,\\
  PRPC MESSAGE pRpcMessage,
  DWORD *_pdwStubPhase);
HRESULT __stdcall ITPCC_Payment_Proxy( ITPCC __RPC_FAR * This,
  /* [in] */ VARIANT txn_in,
  /* [out] */ VARIANT RPC FAR *txn out);
void __RPC_STUB ITPCC_Payment_Stub(
  IRpcStubBuffer *This,
  IRpcChannelBuffer *_pRpcChannelBuffer,
  PRPC_MESSAGE _pRpcMessage,
  DWORD * pdwStubPhase);
HRESULT __stdcall ITPCC_Delivery_Proxy(
  ITPCC \ \underline{\overline{\phantom{A}}}RPC\_FAR * This,
  /* [in] */ VARIANT txn_in,
  /* [out] */ VARIANT __RPC_FAR *txn_out);
void RPC STUB ITPCC Delivery Stub(
  IRpcStubBuffer *This,
  IRpcChannelBuffer * pRpcChannelBuffer,
  PRPC_MESSAGE _pRpcMessage,
  DWORD *_pdwStubPhase);
HRESULT stdcall ITPCC StockLevel Proxy(
  ITPCC __RPC_FAR * This,
  /* [in] */ VARIANT txn in,
  /* [out] */ VARIANT __RPC_FAR *txn_out);
void __RPC_STUB ITPCC_StockLevel_Stub(
  IRpcStubBuffer *This,
  IRpcChannelBuffer *_pRpcChannelBuffer, PRPC_MESSAGE _pRpcMessage,
  DWORD *_pdwStubPhase);
HRESULT stdcall ITPCC_OrderStatus_Proxy(
  ITPCC RPC FAR * This,
  /* [in] */ VARIANT txn_in,
  /* [out] */ VARIANT RPC FAR *txn out);
void RPC STUB ITPCC OrderStatus Stub(
  IRpcStubBuffer *This,
  IRpc Channel Buffer *\_pRpc Channel Buffer,\\
  PRPC_MESSAGE _pRpcMessage,
  DWORD * pdwStubPhase);
```

```
HRESULT stdcall ITPCC CallSetComplete Proxy(
  ITPCC RPC FAR * This);
void __RPC_STUB ITPCC_CallSetComplete_Stub(
  IRpcStubBuffer *This,
  IRpcChannelBuffer *_pRpcChannelBuffer,
  PRPC_MESSAGE _pRpcMessage,
  DWORD * pdwStubPhase);
#endif
        /* ITPCC INTERFACE DEFINED */
/* Additional Prototypes for ALL interfaces */
unsigned long
                  _RPC_USER_VARIANT_UserSize( unsigned long
  RPC FAR *, unsigned long
                             , VARIANT RPC FAR * );
unsigned char __RPC_FAR * __RPC_USER VARIANT_UserMarshal(
unsigned long RPC_FAR *, unsigned char RPC_FAR *, VARIANT
 RPC FAR * );
unsigned char RPC FAR * RPC USER
VARIANT_UserUnmarshal(unsigned long __RPC_FAR *, unsigned char
 _RPC_FAR *, VARIANT __RPC_FAR * );
 /* end of Additional Prototypes */
#ifdef cplusplus
#endif
#endif
tpcc com ps.idl
        FILE:
                          ITPCC IDL
                                   Microsoft TPC-C Kit Ver.
4.20.000
                                  Copyright Microsoft, 1999
        All Rights Reserved
```

```
not yet audited
          PURPOSE:
                                Defines the interface used by TPCC. This
interface can be implemented by C++ components.
* Change history:
                     4.20.000 - first version
// Forward declare all types defined
interface ITPCC;
import "oaidl.idl";
import "ocidl.idl";
                     object,
                     oleautomation,
                     uuid(FEEE6AA2-84B1-11d2-BA47-00C04FBFE08B),
                     helpstring("ITPCC Interface"),
                     pointer_default(unique)
          interface ITPCC: IUnknown
```

```
//@@MIDL_FILE_HEADING( )
    HRESULT stdcall NewOrder
                                                                          #if !defined( M IA64) && !defined( M AXP64)
                                                            [in]
VARIANT txn_in,
                                                                         #ifdef __cplusplus
                                                                         extern "C" {
                                                            [out]
                                                                          #endif
VARIANT *txn out
                                                            );
                                                                         #include <rpc.h>
    HRESULT _stdcall Payment
                                                                         #include <rpendr.h>
                                                            [in]
VARIANT txn in,
                                                                          #ifdef MIDL USE GUIDDEF
                                                            [out]
                                                                          #ifndef INITGUID
VARIANT *txn_out
                                                            );
                                                                          #define INITGUID
                                                                          #include <guiddef.h>
    HRESULT _stdcall Delivery
                                                                          #undef INITGUID
                                                                         #else
                                                            [in]
                                                                         #include <guiddef.h>
VARIANT txn_in,
                                                                          #endif
                                                            [out]
VARIANT *txn_out
                                                                          #define MIDL_DEFINE_GUID(type,name,l,w1,w2,b1,b2,b3,b4,b5,b6,b7,b8) \
                                                                              DEFINE GUID(name,1,w1,w2,b1,b2,b3,b4,b5,b6,b7,b8)
                                                            );
                                                                          #else // !_MIDL_USE_GUIDDEF_
    HRESULT _stdcall StockLevel
                                                            [in]
                                                                          #ifndef __IID_DEFINED
VARIANT txn in,
                                                                          #define IID DEFINED
                                                            [out]
VARIANT *txn out
                                                                          typedef struct IID
                                                            );
                                                                            unsigned long x;
    HRESULT stdcall OrderStatus
                                                                            unsigned short s1;
                                                                            unsigned short s2;
                                                            [in]
                                                                            unsigned char c[8];
VARIANT txn_in,
                                                                          } IID;
                                                            [out]
VARIANT *txn_out
                                                                          #endif // __IID_DEFINED__
                                                            );
                                                                          #ifndef CLSID DEFINED
                                                                          #define CLSID DEFINED
    HRESULT _stdcall CallSetComplete
                                                                          typedef IID CLSID;
                                                                          #endif // CLSID_DEFINED
                                                            ):
                                                                          #define MIDL_DEFINE_GUID(type,name,l,w1,w2,b1,b2,b3,b4,b5,b6,b7,b8) \
          // interface ITPCC
  }:
                                                                              const type name = \{\bar{l}, w1, w2, \{b1, b2, b3, b4, b5, b6, b7, b8\}\}
                                                                          #endif!_MIDL_USE_GUIDDEF_
tpcc com ps i.c
                                                                          MIDL DEFINE GUID(IID,
#pragma warning( disable: 4049 ) /* more than 64k source lines */
                                                                          IID ITPCC,0xFEEE6AA2,0x84B1,0x11d2,0xBA,0x47,0x00,0xC0,0x4F,0xBF,
                                                                          0xE0,0x8B);
/* this ALWAYS GENERATED file contains the IIDs and CLSIDs */
                                                                          #undef MIDL DEFINE GUID
/* link this file in with the server and any clients */
                                                                          #ifdef __cplusplus
/* File created by MIDL compiler version 5.03.0280 */
                                                                          #endif
/* at Sat Apr 08 16:40:10 2000
/* Compiler settings for .\src\tpcc_com_ps.idl:
  Oicf (OptLev=i2), W1, Zp8, env=Win32 (32b run), ms ext, c ext
                                                                          #endif /* !defined(_M_IA64) && !defined(_M_AXP64)*/
  error checks: allocation ref bounds_check enum stub_data
  VC declspec() decoration level:
       declspec(uuid()), __declspec(selectany), __declspec(novtable)
                                                                          #pragma warning( disable: 4049 ) /* more than 64k source lines */
     DECLSPEC UUID(), MIDL INTERFACE()
*/
                                                                          /* this ALWAYS GENERATED file contains the IIDs and CLSIDs */
```

```
#undef MIDL DEFINE GUID
/* link this file in with the server and any clients */
                                                                                                                          #ifdef __cplusplus
/* File created by MIDL compiler version 5.03.0280 */
/* at Sat Apr 08 16:40:10 2000
                                                                                                                          #endif
*/
/* Compiler settings for .\src\tpcc com ps.idl:
   Oicf (OptLev=i2), W1, Zp8, env=Win64 (32b run, appending), ms ext, c ext,
                                                                                                                          #endif /* defined( M IA64) || defined( M AXP64)*/
   error checks: allocation ref bounds_check enum stub_data
           declspec() decoration level:
           declspec(uuid()), declspec(selectany),
                                                                          declspec(novtable)
                                                                                                                          tpcc com ps p.c
        DECLSPEC_UUID(), MIDL_INTERFACE()
                                                                                                                          #pragma warning( disable: 4049 ) /* more than 64k source lines */
//@,@MIDL FILE HEADING( )
                                                                                                                          /* this ALWAYS GENERATED file contains the proxy stub code */
#if defined(_M_IA64) || defined(_M_AXP64)
#ifdef cplusplus
                                                                                                                           /* File created by MIDL compiler version 5.03.0280 */
extern "C" {
                                                                                                                          /* at Sat Apr 08 16:40:10 2000
#endif
                                                                                                                          /* Compiler settings for .\src\tpcc com ps.idl:
                                                                                                                             Oicf (OptLev=i2), W1, Zp8, env=Win32 (32b run), ms_ext, c_ext
#include <rpc.h>
                                                                                                                             error checks: allocation ref bounds check enum stub data
#include <rpcndr.h>
                                                                                                                                    __declspec() decoration level:
                                                                                                                                  __declspec(uuid()), __declspec(selectany), __
                                                                                                                                                                                                    declspec(novtable)
#ifdef _MIDL_USE_GUIDDEF_
#ifndef INITGUID
                                                                                                                          //@@MIDL FILE HEADING()
#define INITGUID
#include <guiddef.h>
                                                                                                                          #if !defined( M IA64) && !defined( M AXP64)
#undef INITGUID
                                                                                                                          #define USE_STUBLESS_PROXY
#else
#include <guiddef.h>
#endif
                                                                                                                          /* verify that the <rpcproxy.h> version is high enough to compile this file*/
                                                                                                                          #ifndef __REDQ_RPCPROXY_H_VERSION
#define MIDL DEFINE GUID(type,name,l,w1,w2,b1,b2,b3,b4,b5,b6,b7,b8) \
                                                                                                                          #define REQUIRED_RPCPROXY_H_VERSION 440
       DEFINE_GUID(name,1,w1,w2,b1,b2,b3,b4,b5,b6,b7,b8)
                                                                                                                          #endif
#else //! MIDL USE GUIDDEF
                                                                                                                          #include "rpcproxy.h"
#ifndef IID DEFINED
                                                                                                                          #ifndef RPCPROXY H VERSION
#define __IID_DEFINED__
                                                                                                                          #error this stub requires an updated version of <rpcproxy.h>
                                                                                                                          #endif // RPCPROXY H VERSION
typedef struct _IID
   unsigned long x;
                                                                                                                          #include "tpcc com ps.h"
   unsigned short s1;
   unsigned short s2;
                                                                                                                          #define TYPE_FORMAT_STRING_SIZE 997
   unsigned char c[8];
                                                                                                                          #define PROC_FORMAT_STRING_SIZE 193
} IID;
                                                                                                                          #define TRANSMIT AS TABLE SIZE 0
                                                                                                                          #define WIRE_MARSHAL_TABLE_SIZE 1
#endif // IID DEFINED
                                                                                                                          typedef struct _MIDL_TYPE_FORMAT_STRING
#ifndef CLSID DEFINED
#define CLSID DEFINED
                                                                                                                                              Pad:
typedef IID CLSID;
                                                                                                                             unsigned char Format[ TYPE_FORMAT_STRING_SIZE ];
#endif // CLSID DEFINED
                                                                                                                              } MIDL TYPE FORMAT STRING;
#define MIDL DEFINE GUID(type,name,l,w1,w2,b1,b2,b3,b4,b5,b6,b7,b8) \
                                                                                                                          typedef struct _MIDL_PROC_FORMAT_STRING
       const type name = \{1,w1,w2,\{b1,b2,b3,b4,b5,b6,b7,b8\}\}
                                                                                                                              short
                                                                                                                                              Pad:
#endif!_MIDL_USE_GUIDDEF_
                                                                                                                              unsigned char Format[ PROC FORMAT STRING SIZE ];
                                                                                                                              } MIDL_PROC_FORMAT_STRING;
MIDL DEFINE GUID(IID,
IID_ITPCC,0xFEEE6AA2,0x84B1,0x11d2,0xBA,0x47,0x00,0xC0,0x4F,0xBF,
0xE0.0x8B);
                                                                                                                         extern const MIDL_TYPE_FORMAT_STRING __MIDL_TypeFormatString;
```

```
(void *)-1 /* ITPCC::Delivery */,
extern const MIDL_PROC_FORMAT_STRING __MIDL_ProcFormatString;
                                                                                                                                     (void *)-1 /* ITPCC::StockLevel */,
                                                                                                                                     (void *)-1 /* ITPCC::OrderStatus */,
/* Standard interface: __MIDL_itf_tpcc_com_ps_0000, ver. 0.0,
                                                                                                                                     (void *)-1 /* ITPCC::CallSetComplete */
const\ CInterfaceStubVtbl\ \ ITPCCStubVtbl =
x00}} */
                                                                                                                                     &IID ITPCC,
/* Object interface: IUnknown, ver. 0.0,
                                                                                                                                     &ITPCC_ServerInfo,
0, /* pure interpreted */
0x46}} */
                                                                                                                                    CStdStubBuffer METHODS
                                                                                                                                extern const USER MARSHAL ROUTINE QUADRUPLE
/* Object interface: ITPCC, ver. 0.0,
                                                                                                                                UserMarshalRoutines[ WIRE MARSHAL TABLE SIZE ];
GUID = \{0xFEEE6AA2, 0x84B1, 0x11d2, \{0xBA, 0x47, 0x00, 0xC0, 0x4F, 0xBF, 0xB
E0,0x8B}} */
                                                                                                                                static const MIDL_STUB_DESC Object_StubDesc =
                                                                                                                                    Ò.
extern const MIDL STUB DESC Object StubDesc;
                                                                                                                                     NdrOleAllocate,
                                                                                                                                     NdrOleFree,
                                                                                                                                    0.
extern const MIDL_SERVER_INFO ITPCC_ServerInfo;
                                                                                                                                    0,
                                                                                                                                    0.
#pragma code seg(".orpc")
                                                                                                                                    0,
static const unsigned short ITPCC_FormatStringOffsetTable[] =
                                                                                                                                       MIDL TypeFormatString.Format,
                                                                                                                                     1, /* -error bounds_check flag */
   0,
   34,
                                                                                                                                     0x20000, /* Ndr library version */
   68,
   102
                                                                                                                                    0x5030118, /* MIDL Version 5.3.280 */
   170
                                                                                                                                    UserMarshalRoutines,
                                                                                                                                     0, /* notify & notify_flag routine table */
   };
                                                                                                                                    0x1, /* MIDL flag */
static const MIDL SERVER INFO ITPCC ServerInfo =
                                                                                                                                    0, /* Reserved3 */
                                                                                                                                    0, /* Reserved4 */
   &Object_StubDesc,
                                                                                                                                    0 /* Reserved5 */
   0,
       MIDL ProcFormatString.Format,
   &ITPCC FormatStringOffsetTable[-3],
                                                                                                                                #pragma data seg(".rdata")
                                                                                                                                static const USER MARSHAL ROUTINE QUADRUPLE
   0,
                                                                                                                                UserMarshalRoutines[ WIRE_MARSHAL_TABLE_SIZE ] =
   0,
   0
static const MIDL STUBLESS PROXY INFO ITPCC ProxyInfo =
                                                                                                                                             VARIANT UserSize
                                                                                                                                             ,VARIANT UserMarshal
                                                                                                                                             ,VARIANT UserUnmarshal
    &Object StubDesc,
       MIDL_ProcFormatString.Format,
                                                                                                                                             ,VARIANT_UserFree
   &ITPCC_FormatStringOffsetTable[-3],
   0,
                                                                                                                                        };
   0
   };
                                                                                                                                #if !defined( RPC WIN32 )
CINTERFACE_PROXY_VTABLE(9) _ITPCCProxyVtbl =
                                                                                                                                #error Invalid build platform for this stub.
                                                                                                                                #endif
   &ITPCC ProxyInfo,
   &IID ITPCC,
                                                                                                                                #if!(TARGET IS NT40 OR LATER)
   IUnknown_QueryInterface_Proxy,
                                                                                                                                #error You need a Windows NT 4.0 or later to run this stub because it uses
   IUnknown_AddRef_Proxy,
                                                                                                                                these features:
   IUnknown Release Proxy,
                                                                                                                                #error -Oif or -Oicf, [wire marshal] or [user marshal] attribute.
   (void *)-1 /* ITPCC::NewOrder */,
                                                                                                                                #error However, your C/C++ compilation flags indicate you intend to run this
   (void *)-1 /* ITPCC::Payment */,
                                                                                                                                app on earlier systems.
```

```
#error This app will die there with the RPC_X_WRONG_STUB_VERSION
                                                                                         NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
                                                                              /* 24 */
error.
                                                                              #else
#endif
                                                                                                              NdrFcShort(0x18), /* MIPS Stack
                                                                              size/offset = 24 */
                                                                              #endif
static const MIDL PROC FORMAT STRING MIDL ProcFormatString =
                                                                              #else
                                                                                                              NdrFcShort( 0x18 ), /* PPC Stack size/offset
    0,
                                                                              = 24 */
                                                                              #endif
                                                                              #else
          /* Procedure NewOrder */
                                                                                                              NdrFcShort(0x18), /* Alpha Stack
                                                                              size/offset = 24 */
                                0x33,
                                                                              #endif
FC_AUTO_HANDLE */
                                                                              /* 26 */
                                                                                         NdrFcShort( 0x3da ), /* Type Offset=986 */
                                                     /* Old Flags: object,
                                0x6c,
Oi2 */
                                                                                         /* Return value */
/* 2 */
          NdrFcLong( 0x0),
                                /* 0 */
/* 6 */
                                                                                         NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
          NdrFcShort(0x3),
                               /* 3 */
                                                                              /* 28 */
#ifndef ALPHA
                                                                              #ifndef ALPHA
#ifndef PPC
                                                                              #ifndef PPC
#if !defined(_MIPS )
                                                                              #if !defined(_MIPS_)
/* 8 */
          \overline{\text{NdrFcShort}} (0x1c), /* x86 Stack size/offset = 28 */
                                                                              /* 30 */
                                                                                        NdrFcShort(0x18), /* x86 Stack size/offset = 24 */
#else
                                                                              #else
                                NdrFcShort( 0x20 ), /* MIPS Stack
                                                                                                              NdrFcShort( 0x1c ), /* MIPS Stack
size/offset = 32 */
                                                                              size/offset = 28 */
#endif
                                                                              #endif
                                                                              #else
#else
                                                                                                              NdrFcShort( 0x1c ), /* PPC Stack size/offset
                                NdrFcShort( 0x20 ), /* PPC Stack size/offset
= 32 */
                                                                               = 28 */
#endif
                                                                              #endif
#else
                                                                              #else
                                NdrFcShort( 0x28 ), /* Alpha Stack
                                                                                                              NdrFcShort( 0x20 ), /* Alpha Stack
size/offset = 40 */
                                                                              size/offset = 32 */
#endif
                                                                              #endif
/* 10 */
          NdrFcShort( 0x0 ),
                                /* 0 */
                                                                              /* 32 */
                                                                                                              /* FC_LONG */
                                                                                         0x8.
/* 12 */
          NdrFcShort( 0x8 ),
                                                                                                                                    /* 0 */
                                                                                                              0x0,
/* 14 */
                                /* Oi2 Flags: srv must size, clt must size, has
          0x7,
return, */
                                                                                         /* Procedure Payment */
                                                     /* 3 */
                                0x3.
                                                                              /* 34 */
                                                                                                              /* FC_AUTO_HANDLE */
                                                                                         0x33
          /* Parameter txn in */
                                                                                                                                    /* Old Flags: object,
                                                                              Oi2 */
/* 16 */
          NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by val, */
                                                                              /* 36 */
                                                                                                              /* 0 */
                                                                                         NdrFcLong( 0x0 ),
\#ifndef \_ALPHA\_
                                                                              /* 40 */
                                                                                                              /* 4 */
                                                                                         NdrFcShort( 0x4 ),
#ifndef PPC
                                                                              #ifndef _ALPHA_
                                                                              #ifndef PPC
#if !defined(_MIPS_)
/* 18 */
         NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
                                                                              #if !defined( MIPS )
#else
                                                                              /* 42 */
                                                                                         NdrFcShort(0x1c), /* x86 Stack size/offset = 28 */
                                NdrFcShort( 0x8 ), /* MIPS Stack
                                                                              #else
size/offset = 8 */
                                                                                                              NdrFcShort( 0x20 ), /* MIPS Stack
                                                                              size/offset = 32 */
#endif
#else
                                                                              #endif
                                NdrFcShort( 0x8 ), /* PPC Stack size/offset
                                                                              #else
= 8 */
                                                                                                              NdrFcShort( 0x20 ), /* PPC Stack size/offset
                                                                              = 32 */
#endif
                                                                              #endif
#else
                                NdrFcShort( 0x8 ), /* Alpha Stack
                                                                              #else
size/offset = 8 */
                                                                                                              NdrFcShort( 0x28 ), /* Alpha Stack
#endif
                                                                              size/offset = 40 */
          NdrFcShort( 0x3c8 ), /* Type Offset=968 */
/* 20 */
                                                                              #endif
                                                                              /* 44 */
                                                                                         NdrFcShort( 0x0 ),
                                                                                                              /* 0 */
                                                                              /* 46 */
          /* Parameter txn_out */
                                                                                         NdrFcShort(0x8),
                                                                                                              /* 8 */
                                                                              /* 48 */
                                                                                         0x7,
                                                                                                              /* Oi2 Flags: srv must size, clt must size, has
/* 22 */ NdrFcShort( 0x4113 ),
                                          /* Flags: must size, must free, out,
                                                                              return, */
simple ref, srv alloc size=16 */
                                                                                                                                    /* 3 */
                                                                                                              0x3,
#ifndef ALPHA
#ifndef_PPC
                                                                                         /* Parameter txn_in */
#if !defined( MIPS )
```

```
/* 50 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by val, */
                                                                                         NdrFcLong( 0x0 ),
#ifndef _ALPHA_
                                                                               /* 74 */
                                                                                         NdrFcShort( 0x5 ),
                                                                                                              /* 5 */
#ifndef PPC
                                                                               #ifndef _ALPHA_
                                                                               #ifndef PPC
#if !defined(_MIPS_)
                                                                              #if !defined( MIPS )
/* 52 */
          NdrFcShort( 0x4 ),
                               /* x86 Stack size/offset = 4 */
#else
                                                                               /* 76 */
                                                                                         NdrFcShort(0x1c), /* x86 Stack size/offset = 28 */
                                NdrFcShort( 0x8 ), /* MIPS Stack
                                                                               #else
                                                                                                               NdrFcShort( 0x20 ), /* MIPS Stack
size/offset = 8 */
#endif
                                                                               size/offset = 32 */
#else
                                                                               #endif
                                NdrFcShort( 0x8 ), /* PPC Stack size/offset
                                                                              #else
= 8 * /
                                                                                                               NdrFcShort( 0x20 ), /* PPC Stack size/offset
#endif
                                                                               = 32 */
#else
                                                                               #endif
                                NdrFcShort( 0x8 ), /* Alpha Stack
                                                                               #else
size/offset = 8 */
                                                                                                               NdrFcShort(0x28), /* Alpha Stack
#endif
                                                                               size/offset = 40 */
          NdrFcShort( 0x3c8 ), /* Type Offset=968 */
/* 54 */
                                                                               #endif
                                                                               /* 78 */
                                                                                         NdrFcShort( 0x0 ).
                                                                                                              /* 0 */
          /* Parameter txn out */
                                                                               /* 80 */
                                                                                         NdrFcShort( 0x8 ),
                                                                                                               /* 8 */
                                                                               /* 82 */
                                                                                         0x7,
                                                                                                               /* Oi2 Flags: srv must size, clt must size, has
         NdrFcShort( 0x4113 ),
                                                                              return, */
                                          /* Flags: must size, must free, out,
simple ref, srv alloc size=16 */
                                                                                                               0x3,
                                                                                                                                    /* 3 */
#ifndef ALPHA
#ifndef _PPC
                                                                                         /* Parameter txn_in */
#if !defined( MIPS )
          NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
                                                                               /* 84 */
                                                                                         NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by val, */
                                                                               #ifndef _ALPHA_
#else
                                NdrFcShort( 0x18 ), /* MIPS Stack
                                                                               #ifndef PPC
size/offset = 24 */
                                                                               #if !defined(_MIPS )
#endif
                                                                               /* 86 */
                                                                                        NdrFcShort( 0x4 ),
                                                                                                              /* x86 Stack size/offset = 4 */
#else
                                                                               #else
                                NdrFcShort(0x18), /* PPC Stack size/offset
                                                                                                               NdrFcShort( 0x8 ), /* MIPS Stack
= 24 */
                                                                               size/offset = 8 */
#endif
                                                                               #endif
                                                                               #else
#else
                                NdrFcShort(0x18), /* Alpha Stack
                                                                                                               NdrFcShort( 0x8 ), /* PPC Stack size/offset
                                                                               = 8 */
size/offset = 24 */
                                                                               #endif
#endif
/* 60 */
          NdrFcShort( 0x3da ), /* Type Offset=986 */
                                                                               #else
                                                                                                               NdrFcShort( 0x8 ), /* Alpha Stack
          /* Return value */
                                                                               size/offset = 8 */
                                                                               #endif
/* 62 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
                                                                               /* 88 */
                                                                                         NdrFcShort( 0x3c8 ), /* Type Offset=968 */
#ifndef ALPHA
#ifndef_PPC
                                                                                         /* Parameter txn_out */
#if !defined( MIPS )
/* 64 */
          NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
                                                                               /* 90 */
                                                                                         NdrFcShort( 0x4113 ),
                                                                                                                         /* Flags: must size, must free, out,
                                                                               simple ref, srv alloc size=16 */
#else
                                NdrFcShort( 0x1c ), /* MIPS Stack
                                                                               #ifndef _ALPHA_
                                                                               #ifndef PPC
size/offset = 28 */
                                                                               #if !defined( MIPS )
#endif
                                                                               /* 92 */
                                                                                         NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
#else
                                NdrFcShort(0x1c), /* PPC Stack size/offset
                                                                              #else
= 28 */
                                                                                                               NdrFcShort(0x18), /* MIPS Stack
#endif
                                                                              size/offset = 24 */
                                                                               #endif
#else
                                NdrFcShort( 0x20 ), /* Alpha Stack
                                                                               #else
                                                                                                               NdrFcShort(0x18), /* PPC Stack size/offset
size/offset = 32 */
                                                                               = 24 */
#endif
/* 66 */
          0x8,
                                /* FC LONG */
                                                                               #endif
                                0x0,
                                                     /* 0 */
                                                                               #else
                                                                                                               NdrFcShort( 0x18 ), /* Alpha Stack
          /* Procedure Delivery */
                                                                               size/offset = 24 */
                                                                               #endif
/* 68 */
          0x33,
                                /* FC AUTO HANDLE */
                                                                               /* 94 */
                                                                                         NdrFcShort( 0x3da ), /* Type Offset=986 */
                                0x6c,
                                                     /* Old Flags: object,
Oi2 */
                                                                                         /* Return value */
```

```
#endif
/* 96 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
                                                                              /* 122 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */
#ifndef _ALPHA_
#ifndef PPC
                                                                                         /* Parameter txn_out */
#if !defined( MIPS )
/* 98 */
          NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
                                                                              /* 124 */ NdrFcShort( 0x4113 ),
                                                                                                                         /* Flags: must size, must free, out,
#else
                                                                              simple ref, srv alloc size=16 */
                                                                              #ifndef _ALPHA_
#ifndef _PPC_
                                NdrFcShort(0x1c), /* MIPS Stack
size/offset = 28 */
#endif
                                                                              #if !defined( MIPS )
#else
                                                                              /* 126 */ NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
                                NdrFcShort( 0x1c ), /* PPC Stack size/offset
                                                                              #else
=28 */
                                                                                                              NdrFcShort(0x18), /* MIPS Stack
#endif
                                                                              size/offset = 24 */
#else
                                                                              #endif
                                NdrFcShort( 0x20 ), /* Alpha Stack
                                                                              #else
size/offset = 32 */
                                                                                                              NdrFcShort( 0x18 ), /* PPC Stack size/offset
                                                                              = 24 */
#endif
/* 100 */ 0x8.
                                /* FC LONG */
                                                                              #endif
                                                     /* 0 */
                                                                              #else
                                0x0,
                                                                                                              NdrFcShort(0x18), /* Alpha Stack
                                                                              size/offset = 24 */
          /* Procedure StockLevel */
                                                                              #endif
/* 102 */ 0x33,
                                /* FC AUTO HANDLE */
                                                                              /* 128 */ NdrFcShort( 0x3da ), /* Type Offset=986 */
                                                     /* Old Flags: object,
                                0x6c,
                                                                                         /* Return value */
Oi2 */
/* 104 */ NdrFcLong( 0x0 ),
                                /* 0 */
/* 108 */ NdrFcShort( 0x6 ),
                                /* 6 */
                                                                              /* 130 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifndef ALPHA
                                                                              #ifndef ALPHA
#ifndef _PPC
                                                                              #ifndef _PPC
#if !defined( MIPS )
                                                                              #if !defined( MIPS )
/* 110 */ NdrFcShort( 0x1c ), /* x86 Stack size/offset = 28 */
                                                                              /* 132 */ NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
                                                                              #else
#else
                                NdrFcShort(0x20), /* MIPS Stack
                                                                                                              NdrFcShort(0x1c), /* MIPS Stack
                                                                              size/offset = 28 */
size/offset = 32 */
                                                                              #endif
#endif
#else
                                                                              #else
                                                                                                              NdrFcShort( 0x1c ), /* PPC Stack size/offset
                                NdrFcShort( 0x20 ), /* PPC Stack size/offset
= 32 */
                                                                              = 28 */
                                                                              #endif
#endif
                                                                              #else
#else
                                NdrFcShort( 0x28 ), /* Alpha Stack
                                                                                                              NdrFcShort( 0x20 ), /* Alpha Stack
size/offset = 40 */
                                                                              size/offset = 32 */
#endif
                                                                              #endif
/* 112 */ NdrFcShort( 0x0 ),
                                /* 0 */
                                                                              /* 134 */ 0x8,
                                                                                                              /* FC LONG */
/* 114 */ NdrFcShort( 0x8 ),
                                                                                                                                    /* 0 */
                                /* 8 */
                                                                                                              0x0.
/* 116 */ 0x7,
                                /* Oi2 Flags: srv must size, clt must size, has
return, */
                                                                                         /* Procedure OrderStatus */
                                                     /* 3 */
                                0x3.
                                                                              /* 136 */ 0x33,
                                                                                                              /* FC AUTO HANDLE */
          /* Parameter txn_in */
                                                                                                              0x6c,
                                                                                                                                    /* Old Flags: object,
                                                                              Oi2 */
/* 118 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by val, */
                                                                              /* 138 */ NdrFcLong( 0x0 ),
                                                                                                              /* 0 */
#ifndef _ALPHA_
                                                                              /* 142 */ NdrFcShort( 0x7 ),
                                                                                                              /* 7 */
#ifndef PPC
                                                                              #ifndef ALPHA
                                                                              #ifndef PPC
#if !defined( MIPS )
/* 120 */ NdrFcShort( 0x4 ),
                               /* x86 Stack size/offset = 4 */
                                                                              #if !defined( MIPS )
#else
                                                                              /* 144 */ NdrFcShort( 0x1c ), /* x86 Stack size/offset = 28 */
                                NdrFcShort( 0x8 ), /* MIPS Stack
                                                                              #else
size/offset = 8 */
                                                                                                              NdrFcShort( 0x20 ), /* MIPS Stack
#endif
                                                                              size/offset = 32 */
#else
                                                                              #endif
                                NdrFcShort( 0x8 ), /* PPC Stack size/offset
                                                                              #else
= 8 */
                                                                                                              NdrFcShort( 0x20 ), /* PPC Stack size/offset
                                                                              = 32 * /
#endif
#else
                                                                              #endif
                                NdrFcShort( 0x8 ),
                                                    /* Alpha Stack
                                                                              #else
size/offset = 8 */
```

```
NdrFcShort( 0x28 ), /* Alpha Stack
                                                                                                             NdrFcShort( 0x20 ), /* Alpha Stack
size/offset = 40 */
                                                                              size/offset = 32 */
#endif
                                                                              #endif
/* 146 */ NdrFcShort( 0x0 ),
                               /* 0 */
                                                                              /* 168 */ 0x8,
                                                                                                             /* FC LONG */
/* 148 */ NdrFcShort( 0x8 ),
                               /* 8 */
                                                                                                                                  /* 0 */
                                                                                                             0x0
/* 150 */ 0x7,
                               /* Oi2 Flags: srv must size, clt must size, has
return, */
                                                                                        /* Procedure CallSetComplete */
                                                     /* 3 */
                                                                              /* 170 */ 0x33,
                                                                                                             /* FC AUTO HANDLE */
          /* Parameter txn in */
                                                                                                                                  /* Old Flags: object,
                                                                                                             0x6c.
                                                                              Oi2 */
/* 152 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by val, */
                                                                              /* 172 */
                                                                                       NdrFcLong( 0x0 ),
                                                                                                             /* 0 */
#ifndef ALPHA
                                                                              /* 176 */ NdrFcShort( 0x8 ),
                                                                                                             /* 8 */
#ifndef PPC
                                                                             #ifndef _ALPHA_
#if !defined( MIPS )
                                                                              /* 178 */ NdrFcShort( 0x8 ),
                                                                                                             /* x86, MIPS, PPC Stack size/offset = 8 */
/* 154 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
                                                                              #else
                                                                                                             NdrFcShort( 0x10 ), /* Alpha Stack
#else
                               NdrFcShort( 0x8 ), /* MIPS Stack
                                                                              size/offset = 16 */
size/offset = 8 */
                                                                              #endif
#endif
                                                                              /* 180 */
                                                                                       NdrFcShort( 0x0 ),
                                                                                                             /* 0 */
#else
                                                                              /* 182 */ NdrFcShort( 0x8 ),
                                                                                                             /* 8 */
                               NdrFcShort( 0x8 ), /* PPC Stack size/offset
                                                                             /* 184 */ 0x4,
                                                                                                             /* Oi2 Flags: has return, */
= 8 * /
#endif
#else
                                                                                        /* Return value */
                               NdrFcShort( 0x8 ), /* Alpha Stack
size/offset = 8 */
                                                                              /* 186 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
                                                                             #ifndef _ALPHA_
#endif
/* 156 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */
                                                                              /* 188 */ NdrFcShort( 0x4 ), /* x86, MIPS, PPC Stack size/offset = 4 */
                                                                              #else
          /* Parameter txn out */
                                                                                                             NdrFcShort( 0x8 ), /* Alpha Stack
                                                                              size/offset = 8 */
/* 158 */ NdrFcShort( 0x4113 ),
                                          /* Flags: must size, must free, out,
                                                                             #endif
simple ref, srv alloc size=16 */
                                                                              /* 190 */ 0x8,
                                                                                                             /* FC LONG */
                                                                                                                                  /* 0 */
#ifndef _ALPHA_
                                                                                                             0x0,
#ifndef PPC
#if !defined( MIPS )
                                                                                                             0x0
/* 160 */ NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
                                                                                  }
                                                                                };
#else
                               NdrFcShort( 0x18 ), /* MIPS Stack
size/offset = 24 */
                                                                              static const MIDL TYPE FORMAT STRING MIDL TypeFormatString =
#endif
                                                                                  0,
#else
                               NdrFcShort(0x18), /* PPC Stack size/offset
= 24 */
                                                                                                             NdrFcShort( 0x0 ), /* 0 */
                                                                              /* 2 */
#endif
#else
                                                                                                             0x12, 0x0, /* FC UP */
                               NdrFcShort(0x18), /* Alpha Stack
                                                                              /* 4 */
                                                                                        NdrFcShort( 0x3b0 ), /* Offset= 944 (948) */
                                                                              /* 6 */
size/offset = 24 */
#endif
                                                                                                             0x2b,
/* 162 */ NdrFcShort( 0x3da ), /* Type Offset=986 */
                                                                             FC NON ENCAPSULATED UNION */
                                                                                                             0x9,
                                                                                                                                  /* FC ULONG */
                                                                              /* 8 */
          /* Return value */
                                                                                                             /* Corr desc: FC_USHORT */
                                                                                        0x7.
                                                                                                             0x0
                                                                                        NdrFcShort( 0xfff8 ), /* -8 */
/* 164 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
                                                                              /* 10 */
#ifndef _ALPHA_
                                                                                        NdrFcShort(0x2), /* Offset= 2 (14) */
                                                                              /* 12 */
#ifndef PPC
                                                                              /* 14 */
                                                                                        NdrFcShort( 0x10 ), /* 16 */
#if !defined(_MIPS_)
                                                                              /* 16 */
                                                                                        NdrFcShort( 0x2b ), /* 43 */
/* 166 */ NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
                                                                              /* 18 */
                                                                                        NdrFcLong(0x3),
                                                                                                            /* 3 */
                                                                              /* 22 */
#else
                                                                                        NdrFcShort( 0x8008 ),
                                                                                                                        /* Simple arm type: FC_LONG */
                               NdrFcShort(0x1c), /* MIPS Stack
                                                                              /* 24 */
                                                                                        NdrFcLong( 0x11 ), /* 17 */
                                                                              /* 28 */
size/offset = 28 */
                                                                                        NdrFcShort(0x8001),
                                                                                                                        /* Simple arm type: FC BYTE */
#endif
                                                                              /* 30 */
                                                                                        NdrFcLong(0x2),
#else
                                                                              /* 34 */
                                                                                        NdrFcShort( 0x8006 ),
                                                                                                                        /* Simple arm type: FC_SHORT */
                                                                             /* 36 */
                                                                                        NdrFcLong( 0x4 ),
                                                                                                            /* 4 */
                               NdrFcShort( 0x1c ), /* PPC Stack size/offset
= 28 */
                                                                              /* 40 */
                                                                                        NdrFcShort( 0x800a ),
                                                                                                                        /* Simple arm type: FC FLOAT */
                                                                              /* 42 */
                                                                                                             /* 5 */
#endif
                                                                                        NdrFcLong( 0x5 ),
#else
```

```
/* 46 */
                                         NdrFcShort( 0x800c),
                                                                            /* 246 */
*/
                                                                                      NdrFcLong(0x4013),
                                                                                                                     /* 16403 */
/* 48 */
                                                                            /* 250 */
          NdrFcLong( 0xb ),
                              /* 11 */
                                                                                      NdrFcShort( 0x27e ), /* Offset= 638 (888) */
/* 52 */
                                         /* Simple arm type: FC_SHORT */
                                                                            /* 252 */
          NdrFcShort( 0x8006 ),
                                                                                      NdrFcLong(0x4016),
                                                                                                                     /* 16406 */
                                                                                      NdrFcShort( 0x278 ), /* Offset= 632 (888) */
/* 54 */
          NdrFcLong( 0xa ),
                                                                            /* 256 */
/* 58 */
          NdrFcShort( 0x8008),
                                         /* Simple arm type: FC LONG */
                                                                            /* 258 */
                                                                                      NdrFcLong(0x4017),
                                                                                                                     /* 16407 */
                                                                                      NdrFcShort( 0x272 ), /* Offset= 626 (888) */
/* 60 */
          NdrFcLong( 0x6 ), /* 6 */
                                                                            /* 262 */
/* 64 */
          NdrFcShort( 0xd6 ), /* Offset= 214 (278) */
                                                                            /* 264 */
                                                                                                          /* 0 */
                                                                                      NdrFcLong(0x0),
                                                                                                          /* Offset= 0 (268) */
                              /* 7 */
/* 66 */
          NdrFcLong(0x7),
                                                                            /* 268 */
                                                                                      NdrFcShort( 0x0 ),
/* 70 */
                                                                            /* 270 */
          NdrFcShort( 0x800c),
                                         /* Simple arm type: FC DOUBLE
                                                                                      NdrFcLong(0x1),
                                                                                                           /* 1 */
*/
                                                                            /* 274 */
                                                                                                          /* Offset= 0 (274) */
                                                                                      NdrFcShort( 0x0 ),
/* 72 */
          NdrFcLong( 0x8 ), /* 8 */
                                                                            /* 276 */
                                                                                      NdrFcShort( 0xffffffff ),
                                                                                                                     /* Offset= -1 (275) */
          NdrFcShort( 0xd0 ), /* Offset= 208 (284) */
                                                                            /* 278 */
/* 76 */
          NdrFcLong( 0xd ), /* 13 */
/* 78 */
                                                                                                                                /* FC_STRUCT */
                                                                                                           0x15,
          NdrFcShort( 0xe2 ), /* Offset= 226 (308) */
/* 82 */
                                                                                                                                /* 7 */
                                                                                                           0x7
                              /* 9 */
                                                                                                           /* 8 */
/* 84 */
                                                                            /* 280 */ NdrFcShort( 0x8 ),
          NdrFcLong( 0x9 ),
                                                                                                           /* FC HYPER */
/* 88 */
          NdrFcShort( 0xee ). /* Offset= 238 (326) */
                                                                            /* 282 */
                                                                                      0xb,
/* 90 */
                                                                                                                                /* FC END */
          NdrFcLong( 0x2000 ),
                                         /* 8192 */
                                                                                                           0x5b,
          NdrFcShort( 0xfa ), /* Offset= 250 (344) */
/* 94 */
                                                                            /* 284 */
          NdrFcLong( 0x24 ), /* 36 */
/* 96 */
                                                                                                           0x12, 0x0, /* FC UP */
          NdrFcShort( 0x308 ), /* Offset= 776 (876) */
/* 100 */
                                                                            /* 286 */ NdrFcShort( 0xc ),
                                                                                                           /* Offset= 12 (298) */
/* 102 */
          NdrFcLong(0x4024),
                                         /* 16420 */
                                                                            /* 288 */
/* 106 */
          NdrFcShort( 0x302 ), /* Offset= 770 (876) */
                                                                                                           0x1b,
                                                                                                                                /* FC CARRAY */
/* 108 */
          NdrFcLong(0x4011),
                                         /* 16401 */
                                                                                                                                /* 1 */
                                                                                                           0x1.
          NdrFcShort( 0x300 ), /* Offset= 768 (880) */
/* 112 */
                                                                            /* 290 */
                                                                                      NdrFcShort(0x2),
                                                                                                           /* 2 */
/* 114 */
          NdrFcLong( 0x4002 ),
                                         /* 16386 */
                                                                            /* 292 */ 0x9,
                                                                                                           /* Corr desc: FC_ULONG */
          NdrFcShort( 0x2fe ), /* Offset= 766 (884) */
/* 118 */
                                                                                                           0x0
/* 120 */
                                                                            /* 294 */ NdrFcShort( 0xfffc ), /* -4 */
          NdrFcLong( 0x4003 ),
                                         /* 16387 */
/* 124 */
          NdrFcShort( 0x2fc ), /* Offset= 764 (888) */
                                                                            /* 296 */ 0x6,
                                                                                                           /* FC SHORT */
/* 126 */
          NdrFcLong(0x4004),
                                                                                                                                /* FC END */
                                         /* 16388 */
                                                                                                           0x5b,
/* 130 */
          NdrFcShort( 0x2fa ), /* Offset= 762 (892) */
                                                                            /* 298 */
/* 132 */
                                                                                                                                /* FC_CSTRUCT */
          NdrFcLong(0x4005),
                                         /* 16389 */
                                                                                                           0x17,
          NdrFcShort( 0x2f8 ), /* Offset= 760 (896) */
/* 136 */
                                                                                                                                /* 3 */
                                                                                                           0x3
/* 138 */
          NdrFcLong(0x400b),
                                         /* 16395 */
                                                                            /* 300 */
                                                                                      NdrFcShort( 0x8 ),
                                                                                                           /* 8 */
/* 142 */
          NdrFcShort( 0x2e6 ), /* Offset= 742 (884) */
                                                                            /* 302 */
                                                                                      NdrFcShort( 0xfffffff2 ),
                                                                                                                     /* Offset= -14 (288) */
/* 144 */
          NdrFcLong( 0x400a ),
                                         /* 16394 */
                                                                            /* 304 */
                                                                                      0x8,
                                                                                                           /* FC LONG */
          NdrFcShort( 0x2e4 ), /* Offset= 740 (888) */
/* 148 */
                                                                                                           0x8,
                                                                                                                                /* FC LONG */
/* 150 */
                                                                            /* 306 */ 0x5c,
                                                                                                           /* FC PAD */
          NdrFcLong( 0x4006),
                                         /* 16390 */
          NdrFcShort( 0x2ea ), /* Offset= 746 (900) */
/* 154 */
                                                                                                                                /* FC END */
                                                                                                           0x5b,
          NdrFcLong( 0x4007 ),
/* 156 */
                                         /* 16391 */
                                                                            /* 308 */
/* 160 */
          NdrFcShort( 0x2e0 ), /* Offset= 736 (896) */
                                                                                                           0x2f,
                                                                                                                                /* FC IP */
          NdrFcLong( 0x4008 ),
/* 162 */
                                         /* 16392 */
                                                                                                           0x5a
/* 166 */
          NdrFcShort( 0x2e2 ), /* Offset= 738 (904) */
                                                                            FC CONSTANT IID */
                                                                                                           /* 0 */
/* 168 */
                                                                            /* 310 */ NdrFcLong( 0x0 ),
          NdrFcLong( 0x400d ),
                                         /* 16397 */
/* 172 */
          NdrFcShort( 0x2e0 ), /* Offset= 736 (908) */
                                                                                      NdrFcShort( 0x0 ),
                                                                                                           /* 0 */
                                                                            /* 314 */
/* 174 */
                                                                            /* 316 */ NdrFcShort( 0x0 ),
                                                                                                           /* 0 */
                                         /* 16393 */
          NdrFcLong(0x4009),
/* 178 */
          NdrFcShort( 0x2de ), /* Offset= 734 (912) */
                                                                            /* 318 */ 0xc0,
                                                                                                           /* 192 */
/* 180 */
          NdrFcLong(0x6000),
                                         /* 24576 */
                                                                                                           0x0
                                                                                                                                /* 0 */
/* 184 */
          NdrFcShort( 0x2dc ), /* Offset= 732 (916) */
                                                                            /* 320 */ 0x0,
                                                                                                           /* 0 */
/* 186 */
          NdrFcLong( 0x400c),
                                         /* 16396 */
                                                                                                                                /* 0 */
                                                                                                           0x0,
/* 190 */
          NdrFcShort( 0x2da ), /* Offset= 730 (920) */
                                                                                                           /* 0 */
                                                                            /* 322 */ 0x0,
/* 192 */
          NdrFcLong( 0x10 ), /* 16 */
                                                                                                           0x0,
                                                                                                                                /* 0 */
/* 196 */
                                                                            /* 324 */ 0x0,
          NdrFcShort( 0x8002 ),
                                         /* Simple arm type: FC_CHAR */
                                                                                                           /* 0 */
          NdrFcLong( 0x12 ), /* 18 */
/* 198 */
                                                                                                           0x46,
                                                                                                                                /* 70 */
/* 202 */
                                                                            /* 326 */
                                         /* Simple arm type: FC SHORT */
          NdrFcShort( 0x8006),
          NdrFcLong( 0x13 ), /* 19 */
/* 204 */
                                                                                                           0x2f,
                                                                                                                                /* FC_IP */
/* 208 */
          NdrFcShort( 0x8008 ),
                                         /* Simple arm type: FC LONG */
                                                                                                           0x5a,
          NdrFcLong( 0x16 ), /* 22 */
/* 210 */
                                                                            FC CONSTANT IID */
/* 214 */
          NdrFcShort( 0x8008 ),
                                         /* Simple arm type: FC LONG */
                                                                                      NdrFcLong( 0x20400 ),
                                                                                                                     /* 132096 */
                                                                            /* 328 */
          NdrFcLong( 0x17 ), /* 23 */
/* 216 */
                                                                            /* 332 */
                                                                                                           /* 0 */
                                                                                      NdrFcShort( 0x0 ),
/* 220 */
          NdrFcShort( 0x8008 ),
                                         /* Simple arm type: FC_LONG */
                                                                            /* 334 */ NdrFcShort( 0x0 ),
                                                                                                           /* 0 */
          NdrFcLong( 0xe ), /* 14 */
/* 222 */
                                                                            /* 336 */ 0xc0,
                                                                                                           /* 192 */
/* 226 */
          NdrFcShort( 0x2be ), /* Offset= 702 (928) */
                                                                                                                                /* 0 */
                                                                                                           0x0.
/* 228 */
          NdrFcLong( 0x400e ),
                                         /* 16398 */
                                                                            /* 338 */ 0x0,
                                                                                                           /* 0 */
          NdrFcShort( 0x2c4 ), /* Offset= 708 (940) */
/* 232 */
                                                                                                                                /* 0 */
                                                                                                           0x0,
                                         /* 16400 */
                                                                                                           /* 0 */
/* 234 */
          NdrFcLong(0x4010),
                                                                            /* 340 */ 0x0,
          NdrFcShort( 0x2c2 ), /* Offset= 706 (944) */
                                                                                                                                /* 0 */
/* 238 */
                                                                                                           0x0.
/* 240 */ NdrFcLong( 0x4012 ),
                                         /* 16402 */
                                                                            /* 342 */ 0x0,
                                                                                                           /* 0 */
```

	0x46,	/ * 70 * /			0x4b,	/* FC_PP */
/* 344 */					0x5c,	/* FC_PAD */
	0x12, 0x10,	/* FC_UP	/* 456 */			
[pointer_deref] */					0x46,	/* FC_NO_REPEAT */
/* 346 */ NdrFcShort(0x2),	/* Offset= 2 (348) */				0x5c,	/* FC_PAD */
/* 348 */			/* 458 */	NdrFcShort(0x4),	/* 4 */	
	0x12, 0x0, /* FC UI	O */	/* 460 */	NdrFcShort(0x4),	/ * 4 * /	
/* 350 */ NdrFcShort(0x1fc)	, /* Offset= 508 (858)	*/	/* 462 */	0x11, 0x0, /* FC RP	*/	
/* 352 */	, ,			NdrFcShort(0xffffff		= -44 (420) */
	0x2a,	/*	/* 466 */	,	<i>"</i>	,
FC_ENCAPSULATED_UNIO					0x5b,	/* FC_END */
	0x49,	/* 73 */			,	
/* 354 */ NdrFcShort(0x18),	,				0x8,	/* FC_LONG */
/* 356 */ NdrFcShort(0xa),	/* 10 */		/* 468 */	0x8	/* FC_LONG */	, re_Earts ,
/* 358 */ NdrFcLong(0x8),	/* 8 */		,,	0.10,	0x5b,	/* FC END */
/* 362 */ NdrFcShort(0x58),		k /	/* 470 */		onso,	, re_END ,
/* 364 */ NdrFcLong(0xd),	/* 13 */	'	, ,,,,,		0x21,	/*
/* 368 */ NdrFcShort(0x78),		*/	FC BOG	US_ARRAY */	OAZI,	,
/* 370 */ NdrFcLong(0x9),	/* 9 */	,	I C_BOG	OS_ARRAT /	0x3,	/* 3 */
/* 374 */ NdrFcShort(0x94),		*/	/* 472 */	NdrFcShort(0x0),	/* 0 */	/ 3 /
/* 376 */ NdrFcLong(0xc),		-/	/* 474 */			ointer, FC_ULONG */
/* 380 */ NdrFcShort(0xbc),		* /	/ 4/4 //	0.119,		/* */
		-/	/* 47 <i>(</i> */	N.JE-Cl(00)	0x0,	/ /
/* 382 */ NdrFcLong(0x24),		sk: /		NdrFcShort(0x0),	/* 0 */ f). /* -1 */	
/* 386 */ NdrFcShort(0x114				NdrFcLong(0xfffffff		COMPLEX */
/* 388 */ NdrFcLong(0x800d			/ * 482 * /	0x4c,	/* FC_EMBEDDED	
/* 392 */ NdrFcShort(0x130		*/	(th. 40.4 sh./	N. 1 E. 01 0 000000	0x0,	/* 0 */
/* 394 */ NdrFcLong(0x10),			/* 484 */	NdrFcShort(0xffffff		= -176 (308) */
/* 398 */ NdrFcShort(0x148		*/	/ * 486 * /	0x5c,	/* FC_PAD */	
/* 400 */ NdrFcLong(0x2),					0x5b,	/* FC_END */
/* 404 */ NdrFcShort(0x160		*/	/* 488 */			
/* 406 */ NdrFcLong(0x3),					0x1a,	/*
/* 410 */ NdrFcShort(0x178		*/	FC_BOG	US_STRUCT */		
/* 412 */ NdrFcLong(0x14),	/* 20 */				0x3,	/* 3 */
/* 416 */ NdrFcShort(0x190), /* Offset= 400 (816)	*/	/* 490 */	NdrFcShort(0x8),	/* 8 */	
/* 418 */ NdrFcShort(0xffffff	fff), /* Offset	= -1 (417) */	/* 492 */	NdrFcShort(0x0),	/* 0 */	
/* 420 */			/* 494 */	NdrFcShort(0x6),	/* Offset= 6 (500) */	
	0x1b,	/* FC CARRAY */	/* 496 */	0x8,	/* FC LONG */	
	0x3,	/* 3 * /			0x36,	/* FC_POINTER */
/* 422 */ NdrFcShort(0x4),	/* 4 */		/* 498 */	0x5c,	/* FC_PAD */	_
/* 424 */ 0x19,	/* Corr desc: field p	ointer, FC ULONG */			0x5b,	/* FC_END */
,	0x0,	/* */ —	/* 500 */		,	_
/* 426 */ NdrFcShort(0x0),	/* 0 */				0x11, 0x0, /* FC RP	*/
/* 428 */			/* 502 */	NdrFcShort(0xffffff6		= -32 (470) */
, .20 ,	0x4b,	/* FC PP */	/* 504 */	1 (411 0011011)	, , ,	3= (170) 7
	0x5c,	/* FC_PAD */	7 301 7		0x21,	/*
/* 430 */	0.650,	/ 1 C_1 / LD /	FC BOG	US_ARRAY */	OAZI,	,
, 150 /	0x48,	/*	1.5_000	00_11III.11 /	0x3,	/* 3 */
FC_VARIABLE_REPEAT */	лато <u>,</u>	,	/* 506 */	NdrFcShort(0x0),	/* 0 */	, 3 1
IC_VARIABLE_REFEAT '/	0x49,	/*	/* 508 */	\ //		ointer, FC ULONG */
FC FIXED OFFSET */	UA+2,	ı	/ 508 ./	UA17,	-	/* */
/* 432 */ NdrFcShort(0x4),	/* 4 */		/* 510 */	NdrFcShort(0x0),	0x0, /* 0 */	1 1
				\ //		
/* 434 */ NdrFcShort(0x0),	/* 0 */ /* 1 */			NdrFcLong(0xfffffff	//	COMPLEY */
/* 436 */ NdrFcShort(0x1),	/* 1 */ /* 0 */		/ * 516 * /	ux4c,	/* FC_EMBEDDED	
/* 438 */ NdrFcShort(0x0),	/* 0 */		/* 510 ± /	NI4.E.C1 4/ 0 000000	0x0,	/* 0 */ - 102 (22() */
/* 440 */ NdrFcShort(0x0),	/* 0 */			NdrFcShort(0xffffff	//	= -192 (326) */
/* 442 */ 0x12, 0x0, /* FC_U		146 (200) 41	/* 520 */	Ux5c,	/* FC_PAD */	W.EG. ENT.
/* 444 */ NdrFcShort(0xffffff	toe), /* Offset	= -146 (298) */			0x5b,	/* FC_END */
/ * 446 * /			/* 522 */			
	0x5b,	/* FC_END */			0x1a,	/*
			FC_BOG	US_STRUCT */		
	0x8,	/* FC_LONG */	1		0x3,	/* 3 */
/* 448 */ 0x5c,	/* FC_PAD */			NdrFcShort(0x8),	/* 8 */	
	0x5b,	/* FC_END */		NdrFcShort(0x0),	/ * 0 * /	
/* 450 */				NdrFcShort(0x6),	/* Offset= 6 (534) */	
	0x16,	/* FC_PSTRUCT */	/* 530 */	0x8,	/* FC_LONG */	
	0x3,	/* 3 * /			0x36,	/* FC_POINTER */
/* 452 */ NdrFcShort(0x8),	/ * 8 * /		/* 532 */	0x5c,	/* FC_PAD */	
/* 454 */					0x5b,	/* FC_END */

/* 534 * /						0x5b,	/* FC END */
		0x11, 0x0, /* FC_RP	*/	/* 612 */			_
/* 536 */	NdrFcShort(0xffffffe		-32 (504) */			0x1a,	/*
/* 538 */	Train Conort Confirm	,, , , , , , , , , , , , , , , , , , , ,	32 (80.)	FC BOG	US_STRUCT */	··········	,
7 330 7		0x1b,	/* FC CARRAY */	I C_BOG	ob_birkoci /	0x3,	/* 3 */
			/* 3 */	/* 614 */	NdrEoShort(Ov.10)		7 3 7
		0x3,	/* 3 */		\ //	/* 16 */	
/* 540 */	NdrFcShort(0x4),	/* 4 */		/ * 616 * /	NdrFcShort(0x0),	/* 0 */	
/* 542 */	0x19,	/* Corr desc: field po	inter, FC_ULONG */		NdrFcShort(0xa),	/* Offset= 10 (628) */	/
		0x0,	/* */	/ * 620 * /	0x8.	/* FC LONG */	
/ * 544 * /	NdrFcShort(0x0),	/* 0 */			,	0x8,	/* FC LONG */
/* 546 */	run esmort (oxo),	, , ,		/* 622 */	0x4a	/* FC EMBEDDED	
/ 340 /		0.41	/* FC DD */	/ 022 /	0.40,		_
			/* FC_PP */			0x0,	/* 0 */
		0x5c,	/* FC_PAD */		NdrFcShort(0xffffffd		-40 (584) */
/* 548 * /				/ * 626 * /	0x36,	/* FC_POINTER */	
		0x48,	/*			0x5b,	/* FC_END */
FC VAR	IABLE_REPEAT */	,		/* 628 */		,	_
1 C_ 1711C	INTELL_REFEREN	0x49,	/*	, 020 ,		0x12, 0x0, /* FC UP	*/
EC EIVE	D OFFGET */	0.49,	/-	/* C20 */	NIE CL 4/0 CCCCC		
	D_OFFSET */				NdrFcShort(0xffffffe	4), /* Offset=	-28 (602) */
	NdrFcShort(0x4),	/* 4 */		/ * 632 * /			
/* 552 * /	NdrFcShort(0x0),	/* 0 */				0x1b,	/* FC_CARRAY */
/* 554 */	NdrFcShort(0x1),	/* 1 */				0x3,	/* 3 * /
	NdrFcShort(0x0),	/* 0 */		/* 634 */	NdrFcShort(0x4),	/* 4 [*] /	
	NdrFcShort(0x0),	/* 0 */		/* 636 */		/* Corr desc: field po	sinter EC III ONG */
				/ . 030 ./	UA17,		_
	0x12, 0x0, /* FC_UP					0x0,	/* */
/* 562 */	NdrFcShort(0x182),	/* Offset= 386 (948) *	*/	/* 638 */	NdrFcShort(0x0),	/* 0 */	
/* 564 */				/* 640 */			
		0x5b,	/* FC_END */			0x4b,	/* FC PP */
		· · · · · · · · · · · · · · · · · · ·	,			0x5c,	/* FC_PAD */
		Ov 9	/* EC ONG */	/* 642 */		oxse,	/ 1C_1/1D /
(de = < < de /	0.5	0x8,	/* FC_LONG */	/ 042 /		0.40	مادا
/* 566 */	0x5c,	/* FC_PAD */				0x48,	/*
		0x5b,	/* FC_END */	FC_VAR	IABLE_REPEAT */		
/ * 568 * /						0x49,	/*
		0x1a,	/*	FC FIXE	D OFFSET */		
FC BOG	US_STRUCT */	····,			NdrFcShort(0x4),	/* 4 */	
те_воо	ob_bircoci /	0v2	/* 3 */	/* 616 */		/* 0 */	
		0x3,		/ ' 040 '/	NULFCOHOLL UXU L	/ · U ·/	
(d. 550 d. (NATE OF TOO O		, 3 ,	/# C 10 /h /	N. 1. E. Cl (0 . 1)		
	NdrFcShort(0x8),	/ * 8 * /	, 3 ,	/* 648 */	NdrFcShort(0x1),	/ * 1 * /	
	NdrFcShort(0x8), NdrFcShort(0x0),		, 3 ,	/* 648 */	NdrFcShort(0x1),		
/ * 572 * /		/ * 8 * /	, 3 ,	/* 648 */ /* 650 */	NdrFcShort(0x1), NdrFcShort(0x0),	/ * 1 * /	
/* 572 */ /* 574 */	NdrFcShort(0x0), NdrFcShort(0x6),	/* 8 */ /* 0 */ /* Offset= 6 (580) */	, , ,	/* 648 */ /* 650 */ /* 652 */	NdrFcShort(0x1), NdrFcShort(0x0), NdrFcShort(0x0),	/* 1 */ /* 0 */ /* 0 */	
/ * 572 * /	NdrFcShort(0x0), NdrFcShort(0x6),	/* 8 */ /* 0 */ /* Offset= 6 (580) */ /* FC_LONG */		/* 648 */ /* 650 */ /* 652 */ /* 654 */	NdrFcShort(0x1), NdrFcShort(0x0), NdrFcShort(0x0), 0x12, 0x0, /* FC_UP	/* 1 */ /* 0 */ /* 0 */ */	-AA (612) */
/* 572 */ /* 574 */ /* 576 */	NdrFcShort(0x0), NdrFcShort(0x6), 0x8,	/* 8 */ /* 0 */ /* Offset= 6 (580) */ /* FC_LONG */ 0x36,	/* FC_POINTER */	/* 648 */ /* 650 */ /* 652 */ /* 654 */	NdrFcShort(0x1), NdrFcShort(0x0), NdrFcShort(0x0),	/* 1 */ /* 0 */ /* 0 */ */	-44 (612) */
/* 572 */ /* 574 */	NdrFcShort(0x0), NdrFcShort(0x6), 0x8,	/* 8 */ /* 0 */ /* Offset= 6 (580) */ /* FC_LONG */ 0x36, /* FC_PAD */	/* FC_POINTER */	/* 648 */ /* 650 */ /* 652 */ /* 654 */	NdrFcShort(0x1), NdrFcShort(0x0), NdrFcShort(0x0), 0x12, 0x0, /* FC_UP	/* 1 */ /* 0 */ /* 0 */ */ 0 */ */ 14), /* Offset=	` '
/* 572 */ /* 574 */ /* 576 */ /* 578 */	NdrFcShort(0x0), NdrFcShort(0x6), 0x8,	/* 8 */ /* 0 */ /* Offset= 6 (580) */ /* FC_LONG */ 0x36,		/* 648 */ /* 650 */ /* 652 */ /* 654 */	NdrFcShort(0x1), NdrFcShort(0x0), NdrFcShort(0x0), 0x12, 0x0, /* FC_UP	/* 1 */ /* 0 */ /* 0 */ */	-44 (612) */ /* FC_END */
/* 572 */ /* 574 */ /* 576 */	NdrFcShort(0x0), NdrFcShort(0x6), 0x8,	/* 8 */ /* 0 */ /* Offset= 6 (580) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b,	/* FC_POINTER */ /* FC_END */	/* 648 */ /* 650 */ /* 652 */ /* 654 */	NdrFcShort(0x1), NdrFcShort(0x0), NdrFcShort(0x0), 0x12, 0x0, /* FC_UP	/* 1 */ /* 0 */ /* 0 */ */ 0 */ */ 14), /* Offset=	` '
/* 572 */ /* 574 */ /* 576 */ /* 578 */	NdrFcShort(0x0), NdrFcShort(0x6), 0x8,	/* 8 */ /* 0 */ /* Offset= 6 (580) */ /* FC_LONG */ 0x36, /* FC_PAD */	/* FC_POINTER */ /* FC_END */	/* 648 */ /* 650 */ /* 652 */ /* 654 */	NdrFcShort(0x1), NdrFcShort(0x0), NdrFcShort(0x0), 0x12, 0x0, /* FC_UP	/* 1 */ /* 0 */ /* 0 */ */ 0 */ */ 14), /* Offset=	` '
/* 572 */ /* 574 */ /* 576 */ /* 578 */ /* 580 */	NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c,	/* 8 */ /* 0 */ /* Offset= 6 (580) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b, 0x11, 0x0, /* FC_RP	/* FC_POINTER */ /* FC_END */ */	/* 648 */ /* 650 */ /* 652 */ /* 654 */ /* 656 */ /* 658 */	NdrFcShort(0x1), NdrFcShort(0x0), NdrFcShort(0x0), 0x12, 0x0, /* FC_UP NdrFcShort(0xffffffd	/* 1 */ /* 0 */ /* 0 */ */ 14), /* Offset= 0x5b, 0x8,	/* FC_END */
/* 572 */ /* 574 */ /* 576 */ /* 578 */ /* 580 */ /* 582 */	NdrFcShort(0x0), NdrFcShort(0x6), 0x8,	/* 8 */ /* 0 */ /* Offset= 6 (580) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b, 0x11, 0x0, /* FC_RP	/* FC_POINTER */ /* FC_END */	/* 648 */ /* 650 */ /* 652 */ /* 654 */	NdrFcShort(0x1), NdrFcShort(0x0), NdrFcShort(0x0), 0x12, 0x0, /* FC_UP NdrFcShort(0xffffffd	/* 1 */ /* 0 */ /* 0 */ */ 14), /* Offset= 0x5b, 0x8, /* FC_PAD */	/* FC_END */ /* FC_LONG */
/* 572 */ /* 574 */ /* 576 */ /* 578 */ /* 580 */	NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c,	/* 8 */ /* 0 */ /* Offset= 6 (580) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b, 0x11, 0x0, /* FC_RP 14), /* Offset=	/* FC_POINTER */ /* FC_END */ */ -44 (538) */	/* 648 */ /* 650 */ /* 652 */ /* 654 */ /* 656 */ /* 660 */	NdrFcShort(0x1), NdrFcShort(0x0), NdrFcShort(0x0), 0x12, 0x0, /* FC_UP NdrFcShort(0xffffffd	/* 1 */ /* 0 */ /* 0 */ */ 14), /* Offset= 0x5b, 0x8,	/* FC_END */
/* 572 */ /* 574 */ /* 576 */ /* 578 */ /* 580 */ /* 582 */	NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c,	/* 8 */ /* 0 */ /* Offset= 6 (580) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b, 0x11, 0x0, /* FC_RP 14), /* Offset= 0x2f,	/* FC_POINTER */ /* FC_END */ */ -44 (538) */ /* FC_IP */	/* 648 */ /* 650 */ /* 652 */ /* 654 */ /* 656 */ /* 658 */	NdrFcShort(0x1), NdrFcShort(0x0), NdrFcShort(0x0), 0x12, 0x0, /* FC_UP NdrFcShort(0xffffffd	/* 1 */ /* 0 */ /* 0 */ */ 14), /* Offset= 0x5b, 0x8, /* FC_PAD */ 0x5b,	/* FC_END */ /* FC_LONG */ /* FC_END */
/* 572 */ /* 574 */ /* 576 */ /* 578 */ /* 580 */ /* 582 */ /* 584 */	NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c, NdrFcShort(0xfffffff	/* 8 */ /* 0 */ /* Offset= 6 (580) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b, 0x11, 0x0, /* FC_RP 14), /* Offset=	/* FC_POINTER */ /* FC_END */ */ -44 (538) */	/* 648 */ /* 650 */ /* 652 */ /* 654 */ /* 658 */ /* 660 */ /* 662 */	NdrFcShort(0x1), NdrFcShort(0x0), NdrFcShort(0x0), 0x12, 0x0, /* FC_UP NdrFcShort(0xffffffd	/* 1 */ /* 0 */ /* 0 */ */ 14), /* Offset= 0x5b, 0x8, /* FC_PAD */	/* FC_END */ /* FC_LONG */
/* 572 */ /* 574 */ /* 576 */ /* 578 */ /* 580 */ /* 582 */ /* 584 */ FC_CON	NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c, NdrFcShort(0xfffffffc	/* 8 */ /* 0 */ /* Offset= 6 (580) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b, 0x11, 0x0, /* FC_RP 14), /* Offset= 0x2f, 0x5a,	/* FC_POINTER */ /* FC_END */ */ -44 (538) */ /* FC_IP */	/* 648 */ /* 650 */ /* 652 */ /* 654 */ /* 658 */ /* 660 */ /* 662 */	NdrFcShort(0x1), NdrFcShort(0x0), NdrFcShort(0x0), 0x12, 0x0, /* FC_UP NdrFcShort(0xffffffd	/* 1 */ /* 0 */ /* 0 */ */ 14), /* Offset= 0x5b, 0x8, /* FC_PAD */ 0x5b, 0x1a,	/* FC_END */ /* FC_LONG */ /* FC_END */ /*
/* 572 */ /* 574 */ /* 576 */ /* 578 */ /* 580 */ /* 582 */ /* 584 */ FC_CON	NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c, NdrFcShort(0xfffffff	/* 8 */ /* 0 */ /* Offset= 6 (580) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b, 0x11, 0x0, /* FC_RP 14), /* Offset= 0x2f,	/* FC_POINTER */ /* FC_END */ */ -44 (538) */ /* FC_IP */	/* 648 */ /* 650 */ /* 652 */ /* 654 */ /* 658 */ /* 660 */ /* 662 */	NdrFcShort(0x1), NdrFcShort(0x0), NdrFcShort(0x0), 0x12, 0x0, /* FC_UP NdrFcShort(0xffffffd	/* 1 */ /* 0 */ /* 0 */ */ 14), /* Offset= 0x5b, 0x8, /* FC_PAD */ 0x5b,	/* FC_END */ /* FC_LONG */ /* FC_END */
/* 572 */ /* 574 */ /* 576 */ /* 578 */ /* 580 */ /* 582 */ /* 584 */ FC_CON /* 586 */	NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c, NdrFcShort(0xfffffffc	/* 8 */ /* 0 */ /* Offset= 6 (580) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b, 0x11, 0x0, /* FC_RP 14), /* Offset= 0x2f, 0x5a,	/* FC_POINTER */ /* FC_END */ */ -44 (538) */ /* FC_IP */	/* 648 */ /* 650 */ /* 652 */ /* 654 */ /* 658 */ /* 660 */ /* 662 */ FC_BOGI	NdrFcShort(0x1), NdrFcShort(0x0), NdrFcShort(0x0), 0x12, 0x0, /* FC_UP NdrFcShort(0xffffffd	/* 1 */ /* 0 */ /* 0 */ */ 14), /* Offset= 0x5b, 0x8, /* FC_PAD */ 0x5b, 0x1a,	/* FC_END */ /* FC_LONG */ /* FC_END */ /*
/* 572 */ /* 574 */ /* 576 */ /* 578 */ /* 580 */ /* 582 */ /* 584 */ FC_CON /* 586 */ /* 590 */	NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c, NdrFcShort(0xfffffffc STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0),	/* 8 */ /* 0 */ /* Offset= 6 (580) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b, 0x11, 0x0, /* FC_RP 14), /* Offset= 0x2f, 0x5a, /* 47 */ /* 0 */	/* FC_POINTER */ /* FC_END */ */ -44 (538) */ /* FC_IP */	/* 648 */ /* 650 */ /* 652 */ /* 654 */ /* 658 */ /* 660 */ FC_BOGI /* 664 */	NdrFcShort(0x1), NdrFcShort(0x0), NdrFcShort(0x0), 0x12, 0x0, /* FC_UP NdrFcShort(0xffffffd 0x5c, US_STRUCT */ NdrFcShort(0x8),	/* 1 */ /* 0 */ /* 0 */ */ 14), /* Offset= 0x5b, 0x8, /* FC_PAD */ 0x5b, 0x1a, 0x3, /* 8 */	/* FC_END */ /* FC_LONG */ /* FC_END */ /*
/* 572 */ /* 574 */ /* 576 */ /* 578 */ /* 580 */ /* 582 */ /* 584 */ FC_CON /* 586 */ /* 590 */ /* 592 */	NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c, NdrFcShort(0xfffffffc STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0),	/* 8 */ /* 0 */ /* Offset= 6 (580) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b, 0x11, 0x0, /* FC_RP 14), /* Offset= 0x2f, 0x5a, /* 47 */ /* 0 */ /* 0 */	/* FC_POINTER */ /* FC_END */ */ -44 (538) */ /* FC_IP */	/* 648 */ /* 650 */ /* 652 */ /* 654 */ /* 658 */ /* 660 */ FC_BOGI /* 664 */ /* 666 */	NdrFcShort(0x1), NdrFcShort(0x0), NdrFcShort(0x0), 0x12, 0x0, /* FC_UP NdrFcShort(0xffffffd 0x5c, US_STRUCT */ NdrFcShort(0x8), NdrFcShort(0x0),	/* 1 */ /* 0 */ /* 0 */ */ 14), /* Offset= 0x5b, 0x8, /* FC_PAD */ 0x5b, 0x1a, 0x3, /* 8 */ /* 0 */	/* FC_END */ /* FC_LONG */ /* FC_END */ /*
/* 572 */ /* 574 */ /* 576 */ /* 578 */ /* 580 */ /* 582 */ /* 584 */ FC_CON /* 586 */ /* 590 */	NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c, NdrFcShort(0xfffffffc STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0),	/* 8 */ /* 0 */ /* Offset= 6 (580) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b, 0x11, 0x0, /* FC_RP 14), /* Offset= 0x2f, 0x5a, /* 47 */ /* 0 */ /* 0 */ /* 192 */	/* FC_POINTER */ /* FC_END */ */ -44 (538) */ /* FC_IP */ /*	/* 648 */ /* 650 */ /* 652 */ /* 654 */ /* 658 */ /* 660 */ FC_BOGI /* 664 */ /* 666 */ /* 668 */	NdrFcShort(0x1), NdrFcShort(0x0), NdrFcShort(0x0), 0x12, 0x0, /* FC_UP NdrFcShort(0xffffffd 0x5c, US_STRUCT */ NdrFcShort(0x8), NdrFcShort(0x0), NdrFcShort(0x0), NdrFcShort(0x6),	/* 1 */ /* 0 */ /* 0 */ */ 14), /* Offset= 0x5b, 0x8, /* FC_PAD */ 0x5b, 0x1a, 0x3, /* 8 */ /* 0 */ /* Offset= 6 (674) */	/* FC_END */ /* FC_LONG */ /* FC_END */ /*
/* 572 */ /* 574 */ /* 576 */ /* 578 */ /* 580 */ /* 582 */ /* 584 */ FC_CON /* 586 */ /* 590 */ /* 592 */ /* 594 */	NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c, NdrFcShort(0xfffffffc STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0xc0,	/* 8 */ /* 0 */ /* Offset= 6 (580) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b, 0x11, 0x0, /* FC_RP 14), /* Offset= 0x2f, 0x5a, /* 47 */ /* 0 */ /* 192 */ 0x0,	/* FC_POINTER */ /* FC_END */ */ -44 (538) */ /* FC_IP */	/* 648 */ /* 650 */ /* 652 */ /* 654 */ /* 658 */ /* 660 */ FC_BOGI /* 664 */ /* 666 */	NdrFcShort(0x1), NdrFcShort(0x0), NdrFcShort(0x0), 0x12, 0x0, /* FC_UP NdrFcShort(0xffffffd 0x5c, US_STRUCT */ NdrFcShort(0x8), NdrFcShort(0x0), NdrFcShort(0x0), NdrFcShort(0x6),	/* 1 */ /* 0 */ /* 0 */ */ 14), /* Offset= 0x5b, 0x8, /* FC_PAD */ 0x5b, 0x1a, 0x3, /* 8 */ /* 0 */ /* Offset= 6 (674) */ /* FC_LONG */	/* FC_END */ /* FC_LONG */ /* FC_END */ /* /* 3 */
/* 572 */ /* 574 */ /* 576 */ /* 578 */ /* 580 */ /* 582 */ /* 584 */ FC_CON /* 586 */ /* 590 */ /* 592 */	NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c, NdrFcShort(0xfffffffc STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0xc0,	/* 8 */ /* 0 */ /* 0 ffset= 6 (580) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b, 0x11, 0x0, /* FC_RP 14), /* Offset= 0x2f, 0x5a, /* 47 */ /* 0 */ /* 0 */ /* 192 */ 0x0, /* 0 */	/* FC_POINTER */ /* FC_END */ */ -44 (538) */ /* FC_IP */ /*	/* 648 */ /* 650 */ /* 652 */ /* 654 */ /* 658 */ /* 660 */ FC_BOGI /* 664 */ /* 668 */ /* 668 */ /* 670 */	NdrFcShort(0x1), NdrFcShort(0x0), NdrFcShort(0x0), 0x12, 0x0, /* FC_UP NdrFcShort(0xffffffd 0x5c, US_STRUCT */ NdrFcShort(0x8), NdrFcShort(0x0), NdrFcShort(0x0), 0x8,	/* 1 */ /* 0 */ /* 0 */ */ 14), /* Offset= 0x5b, 0x8, /* FC_PAD */ 0x5b, 0x1a, 0x3, /* 8 */ /* 0 */ /* Offset= 6 (674) */ /* FC_LONG */ 0x36,	/* FC_END */ /* FC_LONG */ /* FC_END */ /*
/* 572 */ /* 574 */ /* 576 */ /* 578 */ /* 580 */ /* 582 */ /* 584 */ FC_CON /* 586 */ /* 590 */ /* 592 */ /* 594 */	NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c, NdrFcShort(0xfffffffc STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0xc0,	/* 8 */ /* 0 */ /* Offset= 6 (580) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b, 0x11, 0x0, /* FC_RP 14), /* Offset= 0x2f, 0x5a, /* 47 */ /* 0 */ /* 192 */ 0x0,	/* FC_POINTER */ /* FC_END */ */ -44 (538) */ /* FC_IP */ /*	/* 648 */ /* 650 */ /* 652 */ /* 654 */ /* 658 */ /* 660 */ FC_BOGI /* 664 */ /* 666 */ /* 668 */	NdrFcShort(0x1), NdrFcShort(0x0), NdrFcShort(0x0), 0x12, 0x0, /* FC_UP NdrFcShort(0xffffffd 0x5c, US_STRUCT */ NdrFcShort(0x8), NdrFcShort(0x0), NdrFcShort(0x0), 0x8,	/* 1 */ /* 0 */ /* 0 */ */ 14), /* Offset= 0x5b, 0x8, /* FC_PAD */ 0x5b, 0x1a, 0x3, /* 8 */ /* 0 */ /* Offset= 6 (674) */ /* FC_LONG */	/* FC_END */ /* FC_LONG */ /* FC_END */ /* /* 3 */
/* 572 */ /* 574 */ /* 576 */ /* 578 */ /* 580 */ /* 582 */ /* 584 */ FC_CON /* 586 */ /* 590 */ /* 592 */ /* 594 */	NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c, NdrFcShort(0xfffffffc STANT_IID */ NdrFcShort(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0xc0, 0x0,	/* 8 */ /* 0 */ /* 0 ffset= 6 (580) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b, 0x11, 0x0, /* FC_RP 14), /* Offset= 0x2f, 0x5a, /* 47 */ /* 0 */ /* 0 */ /* 192 */ 0x0, /* 0 */	/* FC_POINTER */ /* FC_END */ */ -44 (538) */ /* FC_IP */ /*	/* 648 */ /* 650 */ /* 652 */ /* 654 */ /* 658 */ /* 660 */ FC_BOGI /* 664 */ /* 668 */ /* 668 */ /* 670 */	NdrFcShort(0x1), NdrFcShort(0x0), NdrFcShort(0x0), 0x12, 0x0, /* FC_UP NdrFcShort(0xffffffd 0x5c, US_STRUCT */ NdrFcShort(0x8), NdrFcShort(0x0), NdrFcShort(0x0), 0x8,	/* 1 */ /* 0 */ /* 0 */ */ 14), /* Offset= 0x5b, 0x8, /* FC_PAD */ 0x5b, 0x1a, 0x3, /* 8 */ /* 0 */ /* Offset= 6 (674) */ /* FC_LONG */ 0x36, /* FC_PAD */	/* FC_END */ /* FC_LONG */ /* FC_END */ /* /* 3 */
/* 572 */ /* 574 */ /* 576 */ /* 578 */ /* 580 */ /* 582 */ /* 584 */ FC_CON /* 586 */ /* 590 */ /* 594 */ /* 596 */	NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c, NdrFcShort(0xfffffffc STANT_IID */ NdrFcShort(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0xc0, 0x0,	/* 8 */ /* 0 */ /* 0 */ /* Offset= 6 (580) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b, 0x11, 0x0, /* FC_RP 14), /* Offset= 0x2f, 0x5a, /* 47 */ /* 0 */ /* 0 */ /* 0 */ 0x0, /* 0 */ 0x0, /* 0 */	/* FC_POINTER */ /* FC_END */ */ -44 (538) */ /* FC_IP */ /* /* 0 */	/* 648 */ /* 650 */ /* 652 */ /* 654 */ /* 658 */ /* 660 */ FC_BOGI /* 664 */ /* 668 */ /* 670 */	NdrFcShort(0x1), NdrFcShort(0x0), NdrFcShort(0x0), 0x12, 0x0, /* FC_UP NdrFcShort(0xffffffd 0x5c, US_STRUCT */ NdrFcShort(0x8), NdrFcShort(0x0), NdrFcShort(0x0), 0x8,	/* 1 */ /* 0 */ /* 0 */ */ 14), /* Offset= 0x5b, 0x8, /* FC_PAD */ 0x5b, 0x1a, 0x3, /* 8 */ /* 0 */ /* Offset= 6 (674) */ /* FC_LONG */ 0x36,	/* FC_END */ /* FC_LONG */ /* FC_END */ /* /* 3 */ /* FC_POINTER */
/* 572 */ /* 574 */ /* 576 */ /* 578 */ /* 580 */ /* 584 */ FC_CON /* 586 */ /* 590 */ /* 594 */ /* 596 */	NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c, NdrFcShort(0xfffffff) STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0xc0, 0x0,	/* 8 */ /* 0 */ /* 0 */ /* Offset= 6 (580) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b, 0x11, 0x0, /* FC_RP 14), /* Offset= 0x2f, 0x5a, /* 47 */ /* 0 */ /* 0 */ /* 0 */ 0x0, /* 0 */ 0x0, /* 0 */ 0x0,	/* FC_POINTER */ /* FC_END */ */ -44 (538) */ /* FC_IP */ /*	/* 648 */ /* 650 */ /* 652 */ /* 654 */ /* 658 */ /* 660 */ FC_BOGI /* 664 */ /* 668 */ /* 668 */ /* 670 */	NdrFcShort(0x1), NdrFcShort(0x0), NdrFcShort(0x0), 0x12, 0x0, /* FC_UP NdrFcShort(0xffffffd 0x5c, US_STRUCT */ NdrFcShort(0x8), NdrFcShort(0x0), NdrFcShort(0x0), 0x8,	/* 1 */ /* 0 */ /* 0 */ */ 14), /* Offset= 0x5b, 0x8, /* FC_PAD */ 0x5b, 0x1a, 0x3, /* 8 */ /* 0 */ /* Offset= 6 (674) */ /* Offset= 6 (674) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b,	/* FC_END */ /* FC_LONG */ /* FC_END */ /* /* 3 */ /* FC_POINTER */ /* FC_END */
/* 572 */ /* 574 */ /* 576 */ /* 578 */ /* 580 */ /* 582 */ /* 584 */ FC_CON /* 586 */ /* 590 */ /* 594 */ /* 596 */	NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c, NdrFcShort(0xfffffff) STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0xc0, 0x0,	/* 8 */ /* 0 */ /* 0 */ /* Offset= 6 (580) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b, 0x11, 0x0, /* FC_RP 14), /* Offset= 0x2f, 0x5a, /* 47 */ /* 0 */ /* 0 */ /* 0 */ 0x0, /* 0 */ 0x0, /* 0 */ 0x0, /* 0 */ 0x0, /* 0 */	/* FC_POINTER */ /* FC_END */ */ -44 (538) */ /* FC_IP */ /* 0 */ /* 0 */ /* 0 */	/* 648 */ /* 650 */ /* 652 */ /* 654 */ /* 658 */ /* 660 */ FC_BOGI /* 664 */ /* 668 */ /* 670 */ /* 672 */	NdrFcShort(0x1), NdrFcShort(0x0), NdrFcShort(0x0), Ox12, 0x0, /* FC_UP NdrFcShort(0xffffffd 0x5c, US_STRUCT */ NdrFcShort(0x8), NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c,	/* 1 */ /* 0 */ /* 0 */ */ 14), /* Offset= 0x5b, 0x8, /* FC_PAD */ 0x5b, 0x1a, 0x3, /* 8 */ /* 0 */ /* Offset= 6 (674) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x36, /* FC_PAD */	/* FC_END */ /* FC_LONG */ /* FC_END */ /* /* 3 */ /* FC_POINTER */ /* FC_END */ */
/* 572 */ /* 574 */ /* 576 */ /* 578 */ /* 580 */ /* 582 */ /* 584 */ FC_CON /* 586 */ /* 590 */ /* 594 */ /* 598 */ /* 600 */	NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c, NdrFcShort(0xfffffff) STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0xc0, 0x0,	/* 8 */ /* 0 */ /* 0 */ /* Offset= 6 (580) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b, 0x11, 0x0, /* FC_RP 14), /* Offset= 0x2f, 0x5a, /* 47 */ /* 0 */ /* 0 */ /* 0 */ 0x0, /* 0 */ 0x0, /* 0 */ 0x0,	/* FC_POINTER */ /* FC_END */ */ -44 (538) */ /* FC_IP */ /* /* 0 */	/* 648 */ /* 650 */ /* 652 */ /* 654 */ /* 658 */ /* 660 */ /* 662 */ FC_BOGI /* 664 */ /* 668 */ /* 670 */ /* 674 */ /* 674 */	NdrFcShort(0x1), NdrFcShort(0x0), NdrFcShort(0x0), 0x12, 0x0, /* FC_UP NdrFcShort(0xffffffd 0x5c, US_STRUCT */ NdrFcShort(0x8), NdrFcShort(0x0), NdrFcShort(0x0), 0x8,	/* 1 */ /* 0 */ /* 0 */ */ 14), /* Offset= 0x5b, 0x8, /* FC_PAD */ 0x5b, 0x1a, 0x3, /* 8 */ /* 0 */ /* Offset= 6 (674) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x36, /* FC_PAD */	/* FC_END */ /* FC_LONG */ /* FC_END */ /* /* 3 */ /* FC_POINTER */ /* FC_END */
/* 572 */ /* 574 */ /* 576 */ /* 578 */ /* 580 */ /* 584 */ FC_CON /* 586 */ /* 590 */ /* 594 */ /* 596 */	NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c, NdrFcShort(0xfffffff) STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0xc0, 0x0,	/* 8 */ /* 0 */ /* 0 */ /* Offset= 6 (580) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b, 0x11, 0x0, /* FC_RP 4), /* Offset= 0x2f, 0x5a, /* 47 */ /* 0 */ /* 0 */ 0x0, /* 0 */ 0x0, /* 0 */ 0x0, /* 0 */ 0x46,	/* FC_POINTER */ /* FC_END */ */ -44 (538) */ /* FC_IP */ /* 0 */ /* 0 */ /* 0 */ /* 70 */	/* 648 */ /* 650 */ /* 652 */ /* 654 */ /* 658 */ /* 660 */ FC_BOGI /* 664 */ /* 668 */ /* 670 */ /* 672 */	NdrFcShort(0x1), NdrFcShort(0x0), NdrFcShort(0x0), Ox12, 0x0, /* FC_UP NdrFcShort(0xffffffd 0x5c, US_STRUCT */ NdrFcShort(0x8), NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c,	/* 1 */ /* 0 */ /* 0 */ /* 0 */ */ 14), /* Offset= 0x5b, 0x8, /* FC_PAD */ 0x5b, 0x1a, 0x3, /* 8 */ /* 0 */ /* Offset= 6 (674) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b, 0x11, 0x0, /* FC_RP 14), /* Offset=	/* FC_END */ /* FC_LONG */ /* FC_END */ /* 3 */ /* FC_POINTER */ /* FC_END */ */ -44 (632) */
/* 572 */ /* 574 */ /* 576 */ /* 578 */ /* 580 */ /* 582 */ /* 584 */ FC_CON /* 586 */ /* 590 */ /* 594 */ /* 598 */ /* 600 */	NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c, NdrFcShort(0xfffffff) STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0xc0, 0x0,	/* 8 */ /* 0 */ /* 0 */ /* Offset= 6 (580) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b, 0x11, 0x0, /* FC_RP 14), /* Offset= 0x2f, 0x5a, /* 47 */ /* 0 */ /* 0 */ /* 0 */ 0x0, /* 0 */ 0x0, /* 0 */ 0x0, /* 0 */ 0x0, /* 0 */	/* FC_POINTER */ /* FC_END */ */ -44 (538) */ /* FC_IP */ /* /* 0 */ /* 0 */ /* 70 */ /* 70 */ /* FC_CARRAY */	/* 648 */ /* 650 */ /* 652 */ /* 654 */ /* 658 */ /* 660 */ /* 662 */ FC_BOGI /* 664 */ /* 668 */ /* 670 */ /* 674 */ /* 674 */	NdrFcShort(0x1), NdrFcShort(0x0), NdrFcShort(0x0), Ox12, 0x0, /* FC_UP NdrFcShort(0xffffffd 0x5c, US_STRUCT */ NdrFcShort(0x8), NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c,	/* 1 */ /* 0 */ /* 0 */ */ 14), /* Offset= 0x5b, 0x8, /* FC_PAD */ 0x5b, 0x1a, 0x3, /* 8 */ /* 0 */ /* Offset= 6 (674) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x36, /* FC_PAD */	/* FC_END */ /* FC_LONG */ /* FC_END */ /* 3 */ /* FC_POINTER */ /* FC_END */ */ -44 (632) */ /* FC_SMFARRAY */
/* 572 */ /* 574 */ /* 576 */ /* 578 */ /* 580 */ /* 582 */ /* 584 */ FC_CON /* 586 */ /* 590 */ /* 594 */ /* 598 */ /* 600 */	NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c, NdrFcShort(0xfffffff) STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0xc0, 0x0,	/* 8 */ /* 0 */ /* 0 */ /* Offset= 6 (580) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b, 0x11, 0x0, /* FC_RP 4), /* Offset= 0x2f, 0x5a, /* 47 */ /* 0 */ /* 0 */ 0x0, /* 0 */ 0x0, /* 0 */ 0x0, /* 0 */ 0x46,	/* FC_POINTER */ /* FC_END */ */ -44 (538) */ /* FC_IP */ /* 0 */ /* 0 */ /* 0 */ /* 70 */	/* 648 */ /* 650 */ /* 652 */ /* 654 */ /* 658 */ /* 660 */ /* 662 */ FC_BOGI /* 664 */ /* 668 */ /* 670 */ /* 674 */ /* 674 */	NdrFcShort(0x1), NdrFcShort(0x0), NdrFcShort(0x0), Ox12, 0x0, /* FC_UP NdrFcShort(0xffffffd 0x5c, US_STRUCT */ NdrFcShort(0x8), NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c,	/* 1 */ /* 0 */ /* 0 */ /* 0 */ */ 14), /* Offset= 0x5b, 0x8, /* FC_PAD */ 0x5b, 0x1a, 0x3, /* 8 */ /* 0 */ /* Offset= 6 (674) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b, 0x11, 0x0, /* FC_RP 14), /* Offset=	/* FC_END */ /* FC_LONG */ /* FC_END */ /* 3 */ /* FC_POINTER */ /* FC_END */ */ -44 (632) */
/* 572 */ /* 574 */ /* 576 */ /* 578 */ /* 580 */ /* 582 */ /* 584 */ FC_CON /* 586 */ /* 590 */ /* 592 */ /* 594 */ /* 600 */ /* 602 */	NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c, NdrFcShort(0xfffffffc STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0xc0, 0x0, 0x0,	/* 8 */ /* 0 */ /* 0 */ /* Offset= 6 (580) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b, 0x11, 0x0, /* FC_RP 14), /* Offset= 0x2f, 0x5a, /* 47 */ /* 0 */ /* 0 */ /* 0 */ 0x0, /* 0 */ 0x0, /* 0 */ 0x46, 0x1b, 0x0,	/* FC_POINTER */ /* FC_END */ */ -44 (538) */ /* FC_IP */ /* /* 0 */ /* 0 */ /* 70 */ /* 70 */ /* FC_CARRAY */	/* 648 */ /* 650 */ /* 652 */ /* 654 */ /* 658 */ /* 660 */ /* 662 */ FC_BOGI /* 664 */ /* 668 */ /* 672 */ /* 674 */ /* 676 */ /* 678 */	NdrFcShort(0x1), NdrFcShort(0x0), NdrFcShort(0x0), 0x12, 0x0, /* FC_UP NdrFcShort(0xffffffd 0x5c, US_STRUCT */ NdrFcShort(0x8), NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c,	/* 1 */ /* 0 */ /* 0 */ /* 0 */ */ 14), /* Offset= 0x5b, 0x8, /* FC_PAD */ 0x5b, 0x1a, 0x3, /* 8 */ /* 0 */ /* Offset= 6 (674) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b, 0x11, 0x0, /* FC_RP 14), /* Offset= 0x1d,	/* FC_END */ /* FC_LONG */ /* FC_END */ /* 3 */ /* FC_POINTER */ /* FC_END */ */ -44 (632) */ /* FC_SMFARRAY */
/* 572 */ /* 574 */ /* 576 */ /* 578 */ /* 580 */ /* 582 */ /* 584 */ FC_CON /* 586 */ /* 590 */ /* 592 */ /* 594 */ /* 600 */ /* 602 */	NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c, NdrFcShort(0xfffffffc STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0xc0, 0x0, 0x0, NdrFcShort(0x1),	/* 8 */ /* 0 */ /* 0 */ /* Offset= 6 (580) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b, 0x11, 0x0, /* FC_RP 14), /* Offset= 0x2f, 0x5a, /* 47 */ /* 0 */ /* 0 */ 0x0, /* 0 */ 0x0, /* 0 */ 0x46, 0x1b, 0x0, /* 1 */	/* FC_POINTER */ /* FC_END */ */ -44 (538) */ /* FC_IP */ /* 0 */ /* 0 */ /* 70 */ /* 70 */ /* FC_CARRAY */ /* 0 */	/* 648 */ /* 650 */ /* 652 */ /* 654 */ /* 658 */ /* 660 */ /* 662 */ FC_BOGI /* 664 */ /* 668 */ /* 670 */ /* 672 */ /* 674 */ /* 678 */ /* 680 */	NdrFcShort(0x1), NdrFcShort(0x0), NdrFcShort(0x0), 0x12, 0x0, /* FC_UP NdrFcShort(0xffffffd 0x5c, US_STRUCT */ NdrFcShort(0x8), NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c, NdrFcShort(0x6), 0x8, 0x5c,	/* 1 */ /* 0 */ /* 0 */ /* 0 */ */ 14), /* Offset= 0x5b, 0x8, /* FC_PAD */ 0x5b, 0x1a, 0x3, /* 8 */ /* 0 */ /* Offset= 6 (674) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b, 0x11, 0x0, /* FC_RP 14), /* Offset= 0x1d, 0x0, /* 8 */	/* FC_END */ /* FC_LONG */ /* FC_END */ /* 3 */ /* FC_POINTER */ /* FC_END */ */ -44 (632) */ /* FC_SMFARRAY */
/* 572 */ /* 574 */ /* 576 */ /* 578 */ /* 580 */ /* 582 */ /* 584 */ FC_CON /* 586 */ /* 590 */ /* 592 */ /* 594 */ /* 600 */ /* 602 */	NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c, NdrFcShort(0xfffffffc STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0xc0, 0x0, 0x0, NdrFcShort(0x1),	/* 8 */ /* 0 */ /* 0 */ /* Offset= 6 (580) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b, 0x11, 0x0, /* FC_RP 14), /* Offset= 0x2f, 0x5a, /* 47 */ /* 0 */ /* 0 */ /* 0 */ 0x0, /* 0 */ 0x0, /* 0 */ 0x46, 0x1b, 0x0, /* 1 */ /* Corr desc: field po	/* FC_POINTER */ /* FC_END */ */ -44 (538) */ /* FC_IP */ /* /* 0 */ /* 0 */ /* 70 */ /* FC_CARRAY */ /* 0 */ inter, FC_ULONG */	/* 648 */ /* 650 */ /* 652 */ /* 654 */ /* 658 */ /* 660 */ /* 662 */ FC_BOGI /* 664 */ /* 668 */ /* 672 */ /* 674 */ /* 676 */ /* 678 */	NdrFcShort(0x1), NdrFcShort(0x0), NdrFcShort(0x0), 0x12, 0x0, /* FC_UP NdrFcShort(0xffffffd 0x5c, US_STRUCT */ NdrFcShort(0x8), NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c, NdrFcShort(0x6), 0x8, 0x5c,	/* 1 */ /* 0 */ /* 0 */ /* 0 */ */ 14), /* Offset= 0x5b, 0x8, /* FC_PAD */ 0x5b, 0x1a, 0x3, /* 8 */ /* 0 */ /* Offset= 6 (674) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b, 0x11, 0x0, /* FC_RP 14), /* Offset= 0x1d, 0x0, /* 8 */ /* FC_CHAR */	/* FC_END */ /* FC_LONG */ /* FC_END */ /* 3 */ /* FC_POINTER */ /* FC_END */ */ -44 (632) */ /* FC_SMFARRAY */ /* 0 */
/* 572 */ /* 574 */ /* 576 */ /* 578 */ /* 580 */ /* 582 */ /* 584 */ FC_CON /* 586 */ /* 590 */ /* 592 */ /* 594 */ /* 600 */ /* 602 */ /* 604 */ /* 606 */	NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c, NdrFcShort(0xfffffff) STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0xc0, 0x0, 0x0, 0x0, NdrFcShort(0x1), 0x19,	/* 8 */ /* 0 */ /* 0 */ /* Offset= 6 (580) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b, 0x11, 0x0, /* FC_RP 14), /* Offset= 0x2f, 0x5a, /* 47 */ /* 0 */ /* 0 */ 0x0, /* 0 */ 0x0, /* 0 */ 0x0, /* 1 */ /* Corr desc: field po 0x0,	/* FC_POINTER */ /* FC_END */ */ -44 (538) */ /* FC_IP */ /* 0 */ /* 0 */ /* 70 */ /* 70 */ /* FC_CARRAY */ /* 0 */	/* 648 */ /* 650 */ /* 652 */ /* 654 */ /* 656 */ /* 660 */ FC_BOGI /* 664 */ /* 668 */ /* 670 */ /* 674 */ /* 676 */ /* 678 */ /* 680 */ /* 680 */	NdrFcShort(0x1), NdrFcShort(0x0), NdrFcShort(0x0), 0x12, 0x0, /* FC_UP NdrFcShort(0xffffffd 0x5c, US_STRUCT */ NdrFcShort(0x8), NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c, NdrFcShort(0x6), 0x8, 0x5c,	/* 1 */ /* 0 */ /* 0 */ /* 0 */ */ 14), /* Offset= 0x5b, 0x8, /* FC_PAD */ 0x5b, 0x1a, 0x3, /* 8 */ /* 0 */ /* Offset= 6 (674) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b, 0x11, 0x0, /* FC_RP 14), /* Offset= 0x1d, 0x0, /* 8 */	/* FC_END */ /* FC_LONG */ /* FC_END */ /* 3 */ /* FC_POINTER */ /* FC_END */ */ -44 (632) */ /* FC_SMFARRAY */
/* 572 */ /* 574 */ /* 576 */ /* 578 */ /* 580 */ /* 582 */ /* 584 */ FC_CON /* 586 */ /* 590 */ /* 592 */ /* 594 */ /* 600 */ /* 602 */ /* 604 */ /* 608 */	NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c, NdrFcShort(0xfffffff) STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), 0xc0, 0x0, 0x0, 0x0, 0x0, NdrFcShort(0x1), 0x19, NdrFcShort(0x4),	/* 8 */ /* 0 */ /* 0 */ /* Offset= 6 (580) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b, 0x11, 0x0, /* FC_RP 14), /* Offset= 0x2f, 0x5a, /* 47 */ /* 0 */ /* 0 */ 0x0, /* 0 */ 0x0, /* 0 */ 0x0, /* 1 */ /* Corr desc: field po 0x0, /* 4 */	/* FC_POINTER */ /* FC_END */ */ -44 (538) */ /* FC_IP */ /* /* 0 */ /* 0 */ /* 70 */ /* FC_CARRAY */ /* 0 */ inter, FC_ULONG */	/* 648 */ /* 650 */ /* 652 */ /* 654 */ /* 658 */ /* 660 */ /* 662 */ FC_BOGI /* 664 */ /* 668 */ /* 670 */ /* 672 */ /* 674 */ /* 678 */ /* 680 */	NdrFcShort(0x1), NdrFcShort(0x0), NdrFcShort(0x0), 0x12, 0x0, /* FC_UP NdrFcShort(0xfffffffd 0x5c, US_STRUCT */ NdrFcShort(0x8), NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c, NdrFcShort(0x6), 0x8, 0x5c,	/* 1 */ /* 0 */ /* 0 */ /* 0 */ */ 14), /* Offset= 0x5b, 0x8, /* FC_PAD */ 0x5b, 0x1a, 0x3, /* 8 */ /* 0 */ /* Offset= 6 (674) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b, 0x11, 0x0, /* FC_RP 14), /* Offset= 0x1d, 0x0, /* 8 */ /* FC_CHAR */ 0x5b,	/* FC_END */ /* FC_LONG */ /* FC_END */ /* 3 */ /* FC_POINTER */ /* FC_END */ */ -44 (632) */ /* FC_SMFARRAY */ /* O */ /* FC_END */
/* 572 */ /* 574 */ /* 576 */ /* 578 */ /* 580 */ /* 582 */ /* 584 */ FC_CON /* 586 */ /* 590 */ /* 592 */ /* 594 */ /* 600 */ /* 602 */ /* 604 */ /* 606 */	NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c, NdrFcShort(0xfffffff) STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), 0xc0, 0x0, 0x0, 0x0, 0x0, NdrFcShort(0x1), 0x19, NdrFcShort(0x4),	/* 8 */ /* 0 */ /* 0 */ /* Offset= 6 (580) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b, 0x11, 0x0, /* FC_RP 14), /* Offset= 0x2f, 0x5a, /* 47 */ /* 0 */ /* 0 */ 0x0, /* 0 */ 0x0, /* 0 */ 0x0, /* 1 */ /* Corr desc: field po 0x0,	/* FC_POINTER */ /* FC_END */ */ -44 (538) */ /* FC_IP */ /* /* 0 */ /* 0 */ /* 70 */ /* FC_CARRAY */ /* 0 */ inter, FC_ULONG */	/* 648 */ /* 650 */ /* 652 */ /* 654 */ /* 656 */ /* 660 */ FC_BOGI /* 664 */ /* 668 */ /* 670 */ /* 674 */ /* 676 */ /* 678 */ /* 680 */ /* 680 */	NdrFcShort(0x1), NdrFcShort(0x0), NdrFcShort(0x0), 0x12, 0x0, /* FC_UP NdrFcShort(0xfffffffd 0x5c, US_STRUCT */ NdrFcShort(0x8), NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x5c, NdrFcShort(0x6), 0x8, 0x5c,	/* 1 */ /* 0 */ /* 0 */ /* 0 */ */ 14), /* Offset= 0x5b, 0x8, /* FC_PAD */ 0x5b, 0x1a, 0x3, /* 8 */ /* 0 */ /* Offset= 6 (674) */ /* FC_LONG */ 0x36, /* FC_PAD */ 0x5b, 0x11, 0x0, /* FC_RP 14), /* Offset= 0x1d, 0x0, /* 8 */ /* FC_CHAR */	/* FC_END */ /* FC_LONG */ /* FC_END */ /* 3 */ /* FC_POINTER */ /* FC_END */ */ -44 (632) */ /* FC_SMFARRAY */ /* 0 */

		0.2	(d) (2) d) (A 2	(d) 0 d) (
	/* (0(*/ NI4-E-Cl(010)	0x3,	/* 3 */	/* 750 */	N.J., ECl(00)	0x3,	/* 3 */
Marth Mart	/* 686 */ NdrrcSnort(UX10)				Narresnort (0x8),	/* 8 */	
	/ 088 / 0x8,		/* FC_SHORT */	/ 100 1/		0x4h	/* FC PP */
Proc	/* 690 */ 0x6		/ TC_SHORT /				
FC_EMEDIED COMPLEX	, 0,0 , 0,10,		/*	/* 762 */		0.100,	, 10_1115 ,
	FC EMBEDDED COMPLEX					0x46,	/* FC NO REPEAT */
1-5 (6.06 F 1-1						0x5c,	
Part		NdrFcShort(0xffffff	f1), /* Offset=				
Page	-15 (678) */						
	14 (0(4)	0x5b,	/* FC_END */				24 (746) */
FC DOCUS STRUCT	/* 696 */	Ov 1 o	/*		Narresnort(0x111111	es), /* Offset=	= -24 (/46) */
Marticolour	FC ROGUS STRUCT */	oxia,	7 -	1. 112.1		0x5h	/* FC_FND */
	re_bodes_sincer /	0x3.	/* 3 */			onso,	/ TC_END /
"700 + \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \	/* 698 */ NdrFcShort(0x18)					0x8,	/* FC LONG */
				/* 774 */	0x8,	/* FC_LONG */	
Mark	/* 702 */ NdrFcShort(0xa),		*/			0x5b,	/* FC_END */
	/* 704 */ 0x8,	_		/* 776 */			
	/* 70 <i>C</i> */ 0 4						
	/* /06 */ 0x4c,	_		/* 770 */	NdrEaChart(Ov4)		/* 3 */
	/* 708 */ NdrFcShort(0xffff						ointer FC III ONG */
			24 (004) 7	7 700 7	0.117,		
	, , 10 , 0.120,		/* FC END */	/* 782 */	NdrFcShort(0x0),		, ,
	/* 712 */	,	_	/* 784 */	0x8,		
** ** ** ** ** ** ** *						0x5b,	/* FC_END */
Marteshort (Data)		f0c), /* Offset=	= -244 (470) */	/* 786 */			
*** *** *** *** *** *** *** *	/* 716 */	0.11	(* F.C. C.1 P.D. 137 */				
** 18* 18* ** 18* 18* ** 18* ** 18* ** 18* 18* ** 18* 18* 18* 18* 18* 18* 18* 18* 18* 18* 18* 18* 18* 18*				/* 700 */	NdrEaChart (Ov. 9)		/* 3 */
** 720 * /* 0 k19,	/* 718 */ NdrEcShort(0v1)		/ · 0 ·/		Natresnori (0x8),	/* 8 */	
No.			ointer. FC_ULONG */	1 100 1		0x4b.	/* FC_PP */
**722 * ** NdrFcShort(0x0)	, ,20 , 0.113,						
	/* 722 */ NdrFcShort(0x0),	/* 0 */		/* 792 */		,	_
* 726 *	/* 724 */ 0x1,						
Nation N							
NdFeShort(0x8)		0x5b,	/* FC_END */				/* FC_PAD */
*** 728 ** NdrFcShort(0x8),	/* 726 */	,				/* 4 */	/* FC_PAD */
***N30 **	/* 726 */	0x16,	/* FC_PSTRUCT */	/* 796 */	NdrFcShort(0x4),	/* 4 */ /* 4 */	/* FC_PAD */
		0x16, 0x3,	/* FC_PSTRUCT */	/* 796 */ /* 798 */	NdrFcShort(0x4), 0x12, 0x0, /* FC_UF	/* 4 */ /* 4 */ • */	_
	/* 728 */ NdrFcShort(0x8),	0x16, 0x3,	/* FC_PSTRUCT */	/* 796 */ /* 798 */ /* 800 */	NdrFcShort(0x4), 0x12, 0x0, /* FC_UF	/* 4 */ /* 4 */ • */	_
Number N	/* 728 */ NdrFcShort(0x8),	0x16, 0x3, /* 8 */	/* FC_PSTRUCT */ /* 3 */	/* 796 */ /* 798 */ /* 800 */	NdrFcShort(0x4), 0x12, 0x0, /* FC_UF	/* 4 */ /* 4 */ o */ e8), /* Offset=	= -24 (776) */
0x5c,	/* 728 */ NdrFcShort(0x8),	0x16, 0x3, /* 8 */ 0x4b,	/* FC_PSTRUCT */ /* 3 */ /* FC_PP */	/* 796 */ /* 798 */ /* 800 */	NdrFcShort(0x4), 0x12, 0x0, /* FC_UF	/* 4 */ /* 4 */ o */ e8), /* Offset=	= -24 (776) */ /* FC_END */
/* 734 */ NdrFcShort(0x4),	/* 728 */ NdrFcShort(0x8), /* 730 */	0x16, 0x3, /* 8 */ 0x4b, 0x5c,	/* FC_PSTRUCT */ /* 3 */ /* FC_PP */ /* FC_PAD */	/* 796 */ /* 798 */ /* 800 */ /* 802 */	NdrFcShort(0x4), 0x12, 0x0, /* FC_UF NdrFcShort(0xffffff	/* 4 */ /* 4 */ o */ e8), /* Offset= 0x5b, 0x8,	= -24 (776) */ /* FC_END */
/* 736 */ NdrFcShort(0x4),	/* 728 */ NdrFcShort(0x8), /* 730 */	0x16, 0x3, /* 8 */ 0x4b, 0x5c, 0x46,	/* FC_PSTRUCT */ /* 3 */ /* FC_PP */ /* FC_PAD */ /* FC_NO_REPEAT */	/* 796 */ /* 798 */ /* 800 */ /* 802 */	NdrFcShort(0x4), 0x12, 0x0, /* FC_UF NdrFcShort(0xffffff	/* 4 */ /* 4 */ o */ e8), /* Offset= 0x5b, 0x8, /* FC_LONG */	= -24 (776) */ /* FC_END */ /* FC_LONG */
/* 738 */ 0x12, 0x0, /* FC_UP */	/* 728 */ NdrFcShort(0x8), /* 730 */ /* 732 */	0x16, 0x3, /* 8 */ 0x4b, 0x5c, 0x46, 0x5c,	/* FC_PSTRUCT */ /* 3 */ /* FC_PP */ /* FC_PAD */ /* FC_NO_REPEAT */	/* 796 */ /* 798 */ /* 800 */ /* 802 */ /* 804 */	NdrFcShort(0x4), 0x12, 0x0, /* FC_UF NdrFcShort(0xffffff	/* 4 */ /* 4 */ o */ e8), /* Offset= 0x5b, 0x8, /* FC_LONG */	= -24 (776) */ /* FC_END */ /* FC_LONG */
/* 740 */	/* 728 */ NdrFcShort(0x8), /* 730 */ /* 732 */ /* 734 */ NdrFcShort(0x4),	0x16, 0x3, /* 8 */ 0x4b, 0x5c, 0x46, 0x5c, /* 4 */	/* FC_PSTRUCT */ /* 3 */ /* FC_PP */ /* FC_PAD */ /* FC_NO_REPEAT */	/* 796 */ /* 798 */ /* 800 */ /* 802 */ /* 804 */	NdrFcShort(0x4), 0x12, 0x0, /* FC_UF NdrFcShort(0xffffff	/* 4 */ /* 4 */ o */ e8), /* Offset= 0x5b, 0x8, /* FC_LONG */ 0x5b,	= -24 (776) */ /* FC_END */ /* FC_LONG */ /* FC_END */
/* 742 */	/* 728 */ NdrFcShort(0x8), /* 730 */ /* 732 */ /* 734 */ NdrFcShort(0x4), /* 736 */ NdrFcShort(0x4),	0x16, 0x3, /* 8 */ 0x4b, 0x5c, 0x46, 0x5c, /* 4 */ /* 4 */	/* FC_PSTRUCT */ /* 3 */ /* FC_PP */ /* FC_PAD */ /* FC_NO_REPEAT */	/* 796 */ /* 798 */ /* 800 */ /* 802 */ /* 804 */	NdrFcShort(0x4), 0x12, 0x0, /* FC_UF NdrFcShort(0xffffff	/* 4 */ /* 4 */ > */ e8), /* Offset= 0x5b, 0x8, /* FC_LONG */ 0x5b, 0x1b,	= -24 (776) */ /* FC_END */ /* FC_LONG */ /* FC_END */ /* FC_CARRAY */
*** ***	/* 728 */ NdrFcShort(0x8), /* 730 */ /* 732 */ /* 734 */ NdrFcShort(0x4), /* 736 */ NdrFcShort(0x4), /* 738 */ 0x12, 0x0, /* FC_U	0x16, 0x3, /* 8 */ 0x4b, 0x5c, 0x46, 0x5c, /* 4 */ /* 4 */ P */	/* FC_PSTRUCT */ /* 3 */ /* FC_PP */ /* FC_PAD */ /* FC_NO_REPEAT */ /* FC_PAD */	/* 796 */ /* 798 */ /* 800 */ /* 802 */ /* 804 */ /* 806 */	NdrFcShort(0x4), 0x12, 0x0, /* FC_UF NdrFcShort(0xffffff 0x8,	/* 4 */ /* 4 */ o */ e8), /* Offset= 0x5b, 0x8, /* FC_LONG */ 0x5b, 0x1b, 0x7,	= -24 (776) */ /* FC_END */ /* FC_LONG */ /* FC_END */ /* FC_CARRAY */
	/* 728 */ NdrFcShort(0x8), /* 730 */ /* 732 */ /* 734 */ NdrFcShort(0x4), /* 736 */ NdrFcShort(0x4), /* 738 */ 0x12, 0x0, /* FC_U /* 740 */ NdrFcShort(0xffff)	0x16, 0x3, /* 8 */ 0x4b, 0x5c, 0x46, 0x5c, /* 4 */ /* 4 */ P */	/* FC_PSTRUCT */ /* 3 */ /* FC_PP */ /* FC_PAD */ /* FC_NO_REPEAT */ /* FC_PAD */	/* 796 */ /* 798 */ /* 800 */ /* 802 */ /* 804 */ /* 806 */	NdrFcShort(0x4), 0x12, 0x0, /* FC_UF NdrFcShort(0xffffff 0x8, NdrFcShort(0x8),	/* 4 */ /* 4 */ o */ e8), /* Offset= 0x5b, 0x8, /* FC_LONG */ 0x5b, 0x1b, 0x7, /* 8 */	= -24 (776) */ /* FC_END */ /* FC_LONG */ /* FC_END */ /* FC_CARRAY */ /* 7 */
/* 744 */ 0x8,	/* 728 */ NdrFcShort(0x8), /* 730 */ /* 732 */ /* 734 */ NdrFcShort(0x4), /* 736 */ NdrFcShort(0x4), /* 738 */ 0x12, 0x0, /* FC_U /* 740 */ NdrFcShort(0xffff)	0x16, 0x3, /* 8 */ 0x4b, 0x5c, 0x46, 0x5c, /* 4 */ /* 4 */ P */ fte8), /* Offset=	/* FC_PSTRUCT */ /* 3 */ /* FC_PP */ /* FC_PAD */ /* FC_NO_REPEAT */ /* FC_PAD */ = -24 (716) */	/* 796 */ /* 798 */ /* 800 */ /* 802 */ /* 804 */ /* 806 */ /* 808 */ /* 810 */	NdrFcShort(0x4), 0x12, 0x0, /* FC_UF NdrFcShort(0xffffff 0x8, NdrFcShort(0x8), 0x19,	/* 4 */ /* 4 */ o */ e8), /* Offset= 0x5b, 0x8, /* FC_LONG */ 0x5b, 0x1b, 0x7, /* 8 */ /* Corr desc: field pox0,	= -24 (776) */ /* FC_END */ /* FC_LONG */ /* FC_END */ /* FC_CARRAY */ /* 7 */ ointer, FC_ULONG */
0x5b,	/* 728 */ NdrFcShort(0x8), /* 730 */ /* 732 */ /* 734 */ NdrFcShort(0x4), /* 736 */ NdrFcShort(0x4), /* 738 */ 0x12, 0x0, /* FC_U /* 740 */ NdrFcShort(0xffff)	0x16, 0x3, /* 8 */ 0x4b, 0x5c, 0x46, 0x5c, /* 4 */ /* 4 */ P */ ffe8), /* Offset= 0x5b,	/* FC_PSTRUCT */ /* 3 */ /* FC_PP */ /* FC_PAD */ /* FC_NO_REPEAT */ /* FC_PAD */ = -24 (716) */ /* FC_END */	/* 796 */ /* 798 */ /* 800 */ /* 802 */ /* 804 */ /* 806 */ /* 808 */ /* 810 */	NdrFcShort(0x4), 0x12, 0x0, /* FC_UF NdrFcShort(0xfffffff 0x8, NdrFcShort(0x8), 0x19, NdrFcShort(0x0),	/* 4 */ /* 4 */ 0 */ e8), /* Offset= 0x5b, 0x8, /* FC_LONG */ 0x5b, 0x1b, 0x7, /* 8 */ /* Corr desc: field pox0, /* 0 */	= -24 (776) */ /* FC_END */ /* FC_LONG */ /* FC_END */ /* FC_CARRAY */ /* 7 */ ointer, FC_ULONG */
/* 746 */	/* 728 */ NdrFcShort(0x8), /* 730 */ /* 732 */ /* 734 */ NdrFcShort(0x4), /* 736 */ NdrFcShort(0x4), /* 738 */ 0x12, 0x0, /* FC_U /* 740 */ NdrFcShort(0xffff)	0x16, 0x3, /* 8 */ 0x4b, 0x5c, 0x46, 0x5c, /* 4 */ /* 4 */ P */ fe8), /* Offset= 0x5b, 0x8,	/* FC_PSTRUCT */ /* 3 */ /* FC_PP */ /* FC_PAD */ /* FC_NO_REPEAT */ /* FC_PAD */ = -24 (716) */ /* FC_END */	/* 796 */ /* 798 */ /* 800 */ /* 802 */ /* 804 */ /* 806 */ /* 808 */ /* 810 */	NdrFcShort(0x4), 0x12, 0x0, /* FC_UF NdrFcShort(0xfffffff 0x8, NdrFcShort(0x8), 0x19, NdrFcShort(0x0),	/* 4 */ /* 4 */ 0 */ e8), /* Offset= 0x5b, 0x8, /* FC_LONG */ 0x5b, 0x1b, 0x7, /* 8 */ /* Corr desc: field pox0, /* 0 */ /* FC_HYPER */	= -24 (776) */ /* FC_END */ /* FC_LONG */ /* FC_END */ /* FC_CARRAY */ /* 7 */ pointer, FC_ULONG */ /* */
0x1b, /* FC_CARRAY */ 0x1, /* 1 */ /* 748 */ NdrFcShort(0x2), /* 2 */ /* 750 */ 0x19, /* Corr desc: field pointer, FC_ULONG */ 0x0, /* */ /* 752 */ NdrFcShort(0x0), /* 0 */ /* 754 */ 0x6, /* FC_SHORT */ 0x5b, /* FC_END */ /* 824 */ NdrFcShort(0x4), /* 4 */ 0x3, /* 3 */ NdrFcShort(0x8), /* 8 */ /* 820 */ 0x4b, /* FC_PP */ 0x5c, /* FC_PAD */ /* 822 */ 0x46, /* FC_NO_REPEAT */ 0x5c, /* FC_PAD */ /* 824 */ NdrFcShort(0x4), /* 4 */	/* 728 */ NdrFcShort(0x8), /* 730 */ /* 732 */ /* 734 */ NdrFcShort(0x4), /* 736 */ NdrFcShort(0x4), /* 738 */ 0x12, 0x0, /* FC_U /* 740 */ NdrFcShort(0xffff)	0x16, 0x3, /* 8 */ 0x4b, 0x5c, 0x46, 0x5c, /* 4 */ /* 4 */ P */ fe8), /* Offset= 0x5b, 0x8, /* FC_LONG */	/* FC_PSTRUCT */ /* 3 */ /* FC_PP */ /* FC_PAD */ /* FC_NO_REPEAT */ /* FC_PAD */ = -24 (716) */ /* FC_END */ /* FC_LONG */	/* 796 */ /* 798 */ /* 800 */ /* 802 */ /* 804 */ /* 806 */ /* 810 */ /* 812 */ /* 814 */	NdrFcShort(0x4), 0x12, 0x0, /* FC_UF NdrFcShort(0xfffffff 0x8, NdrFcShort(0x8), 0x19, NdrFcShort(0x0),	/* 4 */ /* 4 */ 0 */ e8), /* Offset= 0x5b, 0x8, /* FC_LONG */ 0x5b, 0x1b, 0x7, /* 8 */ /* Corr desc: field pox0, /* 0 */ /* FC_HYPER */	= -24 (776) */ /* FC_END */ /* FC_LONG */ /* FC_END */ /* FC_CARRAY */ /* 7 */ pointer, FC_ULONG */ /* */
NdrFcShort(0x2), * 2 * * 818 * NdrFcShort(0x8), * 8 * * 820 * * 820 * * 822 * * 822 * * 824 * NdrFcShort(0x4), * 8 * * 8 * * 8 * * 8 * * 8 * *	/* 728 */ NdrFcShort(0x8), /* 730 */ /* 732 */ /* 734 */ NdrFcShort(0x4), /* 736 */ NdrFcShort(0x4), /* 738 */ 0x12, 0x0, /* FC_U /* 740 */ NdrFcShort(0xfffff /* 742 */ /* 744 */ 0x8,	0x16, 0x3, /* 8 */ 0x4b, 0x5c, 0x46, 0x5c, /* 4 */ /* 4 */ P */ fe8), /* Offset= 0x5b, 0x8, /* FC_LONG */	/* FC_PSTRUCT */ /* 3 */ /* FC_PP */ /* FC_PAD */ /* FC_NO_REPEAT */ /* FC_PAD */ = -24 (716) */ /* FC_END */ /* FC_LONG */	/* 796 */ /* 798 */ /* 800 */ /* 802 */ /* 804 */ /* 806 */ /* 810 */ /* 812 */ /* 814 */	NdrFcShort(0x4), 0x12, 0x0, /* FC_UF NdrFcShort(0xfffffff 0x8, NdrFcShort(0x8), 0x19, NdrFcShort(0x0),	/* 4 */ /* 4 */ o */ e8), /* Offset= 0x5b, 0x8, /* FC_LONG */ 0x5b, 0x1b, 0x7, /* 8 */ /* Corr desc: field pox0, /* 0 */ /* FC_HYPER */ 0x5b,	= -24 (776) */ /* FC_END */ /* FC_LONG */ /* FC_END */ /* FC_CARRAY */ /* 7 */ cointer, FC_ULONG */ /* */ /* FC_END */
/* 748 */ NdrFcShort(0x2),	/* 728 */ NdrFcShort(0x8), /* 730 */ /* 732 */ /* 734 */ NdrFcShort(0x4), /* 736 */ NdrFcShort(0x4), /* 738 */ 0x12, 0x0, /* FC_U /* 740 */ NdrFcShort(0xfffff /* 742 */ /* 744 */ 0x8,	0x16, 0x3, /* 8 */ 0x4b, 0x5c, 0x46, 0x5c, /* 4 */ /* 4 */ P */ fe8), /* Offset= 0x5b, 0x8, /* FC_LONG */ 0x5b,	/* FC_PSTRUCT */ /* 3 */ /* FC_PP */ /* FC_PAD */ /* FC_NO_REPEAT */ /* FC_PAD */ = -24 (716) */ /* FC_END */ /* FC_LONG */ /* FC_END */	/* 796 */ /* 798 */ /* 800 */ /* 802 */ /* 804 */ /* 806 */ /* 810 */ /* 812 */ /* 814 */	NdrFcShort(0x4), 0x12, 0x0, /* FC_UF NdrFcShort(0xfffffff 0x8, NdrFcShort(0x8), 0x19, NdrFcShort(0x0),	/* 4 */ /* 4 */ o */ e8), /* Offset= 0x5b, 0x8, /* FC_LONG */ 0x5b, 0x1b, 0x7, /* 8 */ /* Corr desc: field pox0, /* 0 */ /* FC_HYPER */ 0x5b,	= -24 (776) */ /* FC_END */ /* FC_LONG */ /* FC_END */ /* FC_CARRAY */ /* 7 */ pointer, FC_ULONG */ /* */ /* FC_END */ /* FC_END */
/* 750 */ 0x19,	/* 728 */ NdrFcShort(0x8), /* 730 */ /* 732 */ /* 734 */ NdrFcShort(0x4), /* 736 */ NdrFcShort(0x4), /* 738 */ 0x12, 0x0, /* FC_U /* 740 */ NdrFcShort(0xfffff /* 742 */ /* 744 */ 0x8,	0x16, 0x3, /* 8 */ 0x4b, 0x5c, 0x46, 0x5c, /* 4 */ /* 4 */ P */ fe8), /* Offset= 0x5b, 0x8, /* FC_LONG */ 0x5b, 0x1b,	/* FC_PSTRUCT */ /* 3 */ /* FC_PP */ /* FC_PAD */ /* FC_NO_REPEAT */ /* FC_PAD */ = -24 (716) */ /* FC_END */ /* FC_LONG */ /* FC_END */ /* FC_END */	/* 796 */ /* 798 */ /* 800 */ /* 802 */ /* 804 */ /* 806 */ /* 810 */ /* 812 */ /* 814 */ /* 816 */	NdrFcShort(0x4), 0x12, 0x0, /* FC_UF NdrFcShort(0xfffffff 0x8, NdrFcShort(0x8), 0x19, NdrFcShort(0x0), 0xb,	/* 4 */ /* 4 */ 0 */ e8), /* Offset= 0x5b, 0x8, /* FC_LONG */ 0x5b, 0x1b, 0x7, /* 8 */ /* Corr desc: field pox0, /* 0x6, /* FC_HYPER */ 0x5b,	= -24 (776) */ /* FC_END */ /* FC_LONG */ /* FC_END */ /* FC_CARRAY */ /* 7 */ pointer, FC_ULONG */ /* */ /* FC_END */ /* FC_END */
/* 752 */ NdrFcShort(0x0),	/* 728 */ /* 730 */ /* 730 */ /* 732 */ /* 734 */ /* 736 */ NdrFcShort(0x4), /* 738 */ /* 740 */ /* 742 */ /* 744 */ 0x8, /* 746 */ /* 748 */ NdrFcShort(0x2),	0x16, 0x3, /* 8 */ 0x4b, 0x5c, 0x46, 0x5c, /* 4 */ /* 4 */ P */ fe8), /* Offset= 0x5b, 0x8, /* FC_LONG */ 0x5b, 0x1b, 0x1,	/* FC_PSTRUCT */ /* 3 */ /* FC_PP */ /* FC_PAD */ /* FC_NO_REPEAT */ /* FC_PAD */ = -24 (716) */ /* FC_END */ /* FC_LONG */ /* FC_END */ /* FC_END */	/* 796 */ /* 798 */ /* 800 */ /* 804 */ /* 806 */ /* 808 */ /* 812 */ /* 814 */ /* 816 */ /* 818 */	NdrFcShort(0x4), 0x12, 0x0, /* FC_UF NdrFcShort(0xfffffff 0x8, NdrFcShort(0x8), 0x19, NdrFcShort(0x0), 0xb,	/* 4 */ /* 4 */ 0 */ e8), /* Offset= 0x5b, 0x8, /* FC_LONG */ 0x5b, 0x1b, 0x7, /* 8 */ /* Corr desc: field pox0, /* 0x6, /* FC_HYPER */ 0x5b,	= -24 (776) */ /* FC_END */ /* FC_LONG */ /* FC_END */ /* FC_CARRAY */ /* 7 */ pointer, FC_ULONG */ /* */ /* FC_END */ /* FC_END */
/* 754 */ 0x6,	/* 728 */ /* 730 */ /* 730 */ /* 732 */ /* 734 */ /* 736 */ NdrFcShort(0x4), /* 738 */ /* 740 */ /* 742 */ /* 744 */ 0x8, /* 746 */ /* 748 */ NdrFcShort(0x2),	0x16, 0x3, /* 8 */ 0x4b, 0x5c, 0x46, 0x5c, /* 4 */ /* 4 */ P */ fe8), /* Offset= 0x5b, 0x8, /* FC_LONG */ 0x5b, 0x1b, 0x1, /* 2 */	/* FC_PSTRUCT */ /* 3 */ /* FC_PP */ /* FC_PAD */ /* FC_NO_REPEAT */ /* FC_PAD */ = -24 (716) */ /* FC_END */ /* FC_END */ /* FC_END */ /* FC_END */ /* FC_CARRAY */ /* 1 */ ointer, FC_ULONG */	/* 796 */ /* 798 */ /* 800 */ /* 804 */ /* 806 */ /* 808 */ /* 812 */ /* 814 */ /* 816 */ /* 818 */	NdrFcShort(0x4), 0x12, 0x0, /* FC_UF NdrFcShort(0xfffffff 0x8, NdrFcShort(0x8), 0x19, NdrFcShort(0x0), 0xb,	/* 4 */ /* 4 */ /* 4 */ 0 */ e8), /* Offset= 0x5b, 0x8, /* FC_LONG */ 0x5b, 0x1b, 0x7, /* 8 */ /* Corr desc: field pox0, /* 0 */ /* FC_HYPER */ 0x5b, 0x16, 0x3, /* 8 */	= -24 (776) */ /* FC_END */ /* FC_LONG */ /* FC_END */ /* FC_CARRAY */ /* 7 */ cointer, FC_ULONG */ /* FC_END */ /* FC_PSTRUCT */ /* 3 */ /* FC_PP */
0x5b, /* FC_END */ /* 756 */ 0x5c, /* FC_PAD */ /* 824 */ NdrFcShort(0x4), /* 4 */	/* 728 */ /* 730 */ /* 730 */ /* 732 */ /* 734 */ /* 736 */ NdrFcShort(0x4), /* 738 */ /* 740 */ /* 742 */ /* 744 */ 0x8, /* 746 */ /* 748 */ /* 750 */ Ox19,	0x16, 0x3, /* 8 */ 0x4b, 0x5c, 0x46, 0x5c, /* 4 */ /* 4 */ P */ fe8), /* Offset= 0x5b, 0x8, /* FC_LONG */ 0x5b, 0x1b, 0x1, /* 2 */ /* Corr desc: field pox0,	/* FC_PSTRUCT */ /* 3 */ /* FC_PP */ /* FC_PAD */ /* FC_NO_REPEAT */ /* FC_PAD */ = -24 (716) */ /* FC_END */ /* FC_END */ /* FC_END */ /* FC_END */ /* FC_CARRAY */ /* 1 */ ointer, FC_ULONG */	/* 796 */ /* 798 */ /* 800 */ /* 804 */ /* 806 */ /* 808 */ /* 812 */ /* 816 */ /* 818 */ /* 820 */	NdrFcShort(0x4), 0x12, 0x0, /* FC_UF NdrFcShort(0xfffffff 0x8, NdrFcShort(0x8), 0x19, NdrFcShort(0x0), 0xb,	/* 4 */ /* 4 */ /* 4 */ 0 */ e8), /* Offset= 0x5b, 0x8, /* FC_LONG */ 0x5b, 0x1b, 0x7, /* 8 */ /* Corr desc: field pox0, /* 0 */ /* FC_HYPER */ 0x5b, 0x16, 0x3, /* 8 */ 0x4b,	= -24 (776) */ /* FC_END */ /* FC_LONG */ /* FC_END */ /* FC_CARRAY */ /* 7 */ cointer, FC_ULONG */ /* FC_END */ /* FC_PSTRUCT */ /* 3 */ /* FC_PP */
/* 756 */ /* 824 */ NdrFcShort(0x4), /* 4 */	/* 728 */ /* 730 */ /* 730 */ /* 732 */ /* 734 */ /* 736 */ /* 738 */ /* 740 */ /* 742 */ /* 744 */ /* 744 */ /* 748 */ /* 750 */ /* 752 */ NdrFcShort(0x8), NdrFcShort(0x4), 0x12, 0x0, /* FC_U NdrFcShort(0xfffff) /* 744 */ 0x8, /* 746 */ /* 748 */ /* 750 */ 0x19, /* 752 */ NdrFcShort(0x0),	0x16, 0x3, /* 8 */ 0x4b, 0x5c, 0x46, 0x5c, /* 4 */ P */ fe8), /* Offset= 0x5b, 0x8, /* FC_LONG */ 0x5b, 0x1b, 0x1, /* 2 */ /* Corr desc: field pox0, /* 0 */	/* FC_PSTRUCT */ /* 3 */ /* FC_PP */ /* FC_PAD */ /* FC_NO_REPEAT */ /* FC_PAD */ = -24 (716) */ /* FC_END */ /* FC_END */ /* FC_END */ /* FC_END */ /* FC_CARRAY */ /* 1 */ ointer, FC_ULONG */	/* 796 */ /* 798 */ /* 800 */ /* 804 */ /* 806 */ /* 808 */ /* 812 */ /* 816 */ /* 818 */ /* 820 */	NdrFcShort(0x4), 0x12, 0x0, /* FC_UF NdrFcShort(0xfffffff 0x8, NdrFcShort(0x8), 0x19, NdrFcShort(0x0), 0xb,	/* 4 */ /* 4 */ /* 4 */ 0 */ e8), /* Offset= 0x5b, 0x8, /* FC_LONG */ 0x5b, 0x1b, 0x7, /* 8 */ /* Corr desc: field pools, /* 0 */ /* FC_HYPER */ 0x5b, 0x16, 0x3, /* 8 */ 0x4b, 0x5c,	= -24 (776) */ /* FC_END */ /* FC_LONG */ /* FC_END */ /* FC_CARRAY */ ointer, FC_ULONG */ /* FC_END */ /* FC_PSTRUCT */ /* FC_PAD */
	/* 728 */ /* 730 */ /* 730 */ /* 732 */ /* 734 */ /* 736 */ NdrFcShort(0x4), /* 738 */ /* 740 */ /* 742 */ /* 744 */ /* 744 */ /* 748 */ /* 750 */ /* 752 */ NdrFcShort(0x8), /* 752 */ NdrFcShort(0x9), /* 752 */ NdrFcShort(0x0), /* 752 */ NdrFcShort(0x0),	0x16, 0x3, /* 8 */ 0x4b, 0x5c, 0x46, 0x5c, /* 4 */ P */ fe8), /* Offset= 0x5b, 0x8, /* FC_LONG */ 0x5b, 0x1b, 0x1, /* 2 */ /* Corr desc: field p 0x0, /* 0 */ /* FC_SHORT */	/* FC_PSTRUCT */ /* 3 */ /* FC_PP */ /* FC_PAD */ /* FC_NO_REPEAT */ /* FC_PAD */ = -24 (716) */ /* FC_END */ /* FC_END */ /* FC_END */ /* FC_CARRAY */ /* 1 */ ointer, FC_ULONG */ /* */	/* 796 */ /* 798 */ /* 800 */ /* 804 */ /* 806 */ /* 808 */ /* 812 */ /* 816 */ /* 818 */ /* 820 */	NdrFcShort(0x4), 0x12, 0x0, /* FC_UF NdrFcShort(0xfffffff 0x8, NdrFcShort(0x8), 0x19, NdrFcShort(0x0), 0xb,	/* 4 */ /* 4 */ 0 */ e8), /* Offset= 0x5b, 0x8, /* FC_LONG */ 0x5b, 0x1b, 0x7, /* 8 */ /* Corr desc: field pox0, /* 0 */ /* FC_HYPER */ 0x5b, 0x16, 0x3, /* 8 */ 0x4b, 0x5c, 0x46,	= -24 (776) */ /* FC_END */ /* FC_LONG */ /* FC_END */ /* FC_CARRAY */ /* 7 */ cointer, FC_ULONG */ /* FC_PSTRUCT */ /* FC_PAD */ /* FC_PAD */ /* FC_NO_REPEAT */
, , , , , , , , , , , , , , , , , , , ,	/* 728 */ /* 730 */ /* 730 */ /* 732 */ /* 734 */ /* 736 */ /* 736 */ /* 738 */ /* 740 */ /* 742 */ /* 744 */ /* 744 */ /* 748 */ /* 750 */ /* 750 */ /* 754 */ /* 754 */ /* 754 */ /* 754 */ /* 0x6,	0x16, 0x3, /* 8 */ 0x4b, 0x5c, 0x46, 0x5c, /* 4 */ P */ fe8), /* Offset= 0x5b, 0x8, /* FC_LONG */ 0x5b, 0x1b, 0x1, /* 2 */ /* Corr desc: field p 0x0, /* 0 */ /* FC_SHORT */	/* FC_PSTRUCT */ /* 3 */ /* FC_PP */ /* FC_PAD */ /* FC_NO_REPEAT */ /* FC_PAD */ = -24 (716) */ /* FC_END */ /* FC_END */ /* FC_END */ /* FC_CARRAY */ /* 1 */ ointer, FC_ULONG */ /* */	/* 796 */ /* 798 */ /* 800 */ /* 804 */ /* 806 */ /* 808 */ /* 812 */ /* 814 */ /* 816 */ /* 822 */	NdrFcShort(0x4), 0x12, 0x0, /* FC_UF NdrFcShort(0xfffffff 0x8, NdrFcShort(0x8), 0x19, NdrFcShort(0x0), 0xb,	/* 4 */ /* 4 */ 0 */ e8), /* Offset= 0x5b, 0x8, /* FC_LONG */ 0x5b, 0x1b, 0x7, /* 8 */ /* Corr desc: field pox0, /* 0 */ /* FC_HYPER */ 0x5b, 0x16, 0x3, /* 8 */ 0x4b, 0x5c, 0x46, 0x5c,	= -24 (776) */ /* FC_END */ /* FC_LONG */ /* FC_END */ /* FC_CARRAY */ /* 7 */ cointer, FC_ULONG */ /* FC_PSTRUCT */ /* FC_PAD */ /* FC_PAD */ /* FC_NO_REPEAT */
	/* 728 */ /* 730 */ /* 730 */ /* 732 */ /* 734 */ /* 736 */ /* 736 */ /* 738 */ /* 740 */ /* 742 */ /* 744 */ /* 744 */ /* 748 */ /* 750 */ /* 750 */ /* 754 */ /* 754 */ /* 754 */ /* 754 */ /* 0x6,	0x16, 0x3, /* 8 */ 0x4b, 0x5c, 0x46, 0x5c, /* 4 */ /* 4 */ P */ fie8), /* Offset= 0x5b, 0x8, /* FC_LONG */ 0x5b, 0x1b, 0x1, /* 2 */ /* Corr desc: field poons, 0x0, /* 0 */ /* FC_SHORT */ 0x5b,	/* FC_PSTRUCT */ /* 3 */ /* FC_PP */ /* FC_PAD */ /* FC_NO_REPEAT */ /* FC_PAD */ = -24 (716) */ /* FC_END */ /* FC_END */ /* FC_CARRAY */ /* 1 */ ointer, FC_ULONG */ /* FC_END */ /* FC_END */	/* 796 */ /* 798 */ /* 800 */ /* 804 */ /* 806 */ /* 808 */ /* 810 */ /* 812 */ /* 816 */ /* 822 */ /* 824 */	NdrFcShort(0x4), 0x12, 0x0, /* FC_UF NdrFcShort(0xfffffff 0x8, NdrFcShort(0x8), 0x19, NdrFcShort(0x0), 0xb,	/* 4 */ /* 4 */ 0 */ e8), /* Offset= 0x5b, 0x8, /* FC_LONG */ 0x5b, 0x1b, 0x7, /* 8 */ /* Corr desc: field pox0, /* 0 */ /* FC_HYPER */ 0x5b, 0x16, 0x3, /* 8 */ 0x4b, 0x5c, 0x46, 0x5c, /* 4 */	= -24 (776) */ /* FC_END */ /* FC_LONG */ /* FC_END */ /* FC_CARRAY */ /* 7 */ cointer, FC_ULONG */ /* FC_PSTRUCT */ /* FC_PAD */ /* FC_PAD */ /* FC_NO_REPEAT */

	0x12, 0x0, /* FC_UP					0x5c,		/* FC_PAD */
	NdrFcShort(0xfffffffe	e8), /* Offset=	-24 (806) */	/ * 900 * /				
/* 832 */						0x12, 0x0,		
		0x5b,	/* FC_END */		NdrFcShort(0xfffffd9	90),	/* Offset=	-624 (278) */
				/* 904 */				
			/* FC_LONG */			0x12, 0x10),	/* FC_UP
/ * 834 * /	0x8,	/* FC_LONG */		[pointer_d				
		0x5b,	/* FC_END */		NdrFcShort(0xfffffd9	92),	/* Offset=	-622 (284) */
/* 836 */				/* 908 */				
			/* FC_STRUCT */			0x12, 0x10),	/* FC_UP
		0x3,	/* 3 */	[pointer_d				
	NdrFcShort(0x8),	/* 8 */			NdrFcShort(0xfffffda	a6),	/* Offset=	-602 (308) */
/* 840 */	0x8,	/* FC_LONG */		/* 912 * /				
			/* FC_LONG */	l		0x12, 0x10),	/* FC_UP
/* 842 */	0x5c,	/* FC_PAD */		[pointer_d				
		0x5b,	/* FC_END */		NdrFcShort(0xfffffdb	54),	/* Offset=	-588 (326) */
/* 844 */		0.41	WEG GIRRING	/ * 916 * /		0 10 0 10		W FG IIP
			/* FC_CARRAY */	l		0x12, 0x10),	/* FC_UP
(* O 4 6 * /	NIE CL ((O.O.)		/* 3 */	[pointer_d		2)	/# O.CC .	574 (244) #/
		/* 8 */	IODT */		NdrFcShort(0xfffffdd	32),	/* Offset=	-5 /4 (344) */
/* 848 */	UX/,	/* Corr desc: FC_USI	HUKI */ _/*_*/	/* 920 */		0 12 0 10		/* EC IID
/* OFO */	N.J., C (0 C. 10)	*****	/ * */			0x12, 0x10),	/* FC_UP
	NdrFcShort(0xffd8)	*	COMPLEX */	[pointer_d		/* O.CC 4	2 (024) */	
/* 852 */	0x4c,	/* FC_EMBEDDED_			NdrFcShort(0x2),	/* Offset=	2 (924) */	
/ * 0.5.4. * /	NIECI (O CCCCC	0x0,		/* 924 */		0 12 0 0	/+ FC IID	* /
	NdrFcShort(0xffffffe		-18 (836) */	/* 02(*/	N.JECl+(01()	0x12, 0x0,		
/* 856 */	uxsc,	/* FC_PAD */	/* EC_END */	/* 926 */ /* 928 */	NdrFcShort(0x16),	/* Offset=	22 (948) */	
/* 0.50 */		0x5b,	/* FC_END */	/* 928 */		015		/* EC_CTD11CT */
/* 858 */		01-	/*			0x15,		/* FC_STRUCT */ /* 7 */
EC DOC	US_STRUCT */	0x1a,	/.	/* 020 */	N.JECl+(010)	0x7,		/. / ./
rc_bog	US_STRUCT '/	02	/* 3 */	/* 930 */ /* 932 */	NdrFcShort(0x10),		NDT * /	
/* 060 */	NdrFcShort(0x28),		/ . 3 ./	/ 932 /	UXO,	/* FC_SHO 0x1,	JK1 ·/	/* FC_BYTE */
	NdrFcShort(0xffffffe		10 (011) */	/* 934 */	0v 1	/* FC_BY	ГС */	/ · rc_bile ·/
	NdrFcShort(0x0),		-10 (044) /	/ 934 /	UXI,	0x38,		/* FC ALIGNM4 */
/ 604 /						0.750,		/ I C ALIGINIT /
/* 866 */	0x6	/* FC_SHORT */		/* 936 */	0x8	/* FC LON		_
/* 866 */	0x6,	/* FC_SHORT */ 0x6	/* FC_SHORT */	/* 936 */	0x8,	/* FC_LON	NG */	_
	•	0x6,	/* FC_SHORT */			0x39,	NG */	/* FC_ALIGNM8 */
/* 866 */ /* 868 */	•	0x6, /* FC_ALIGNM4 */		/* 936 */ /* 938 */		_	NG */ PER */	/* FC_ALIGNM8 */
	0x38,	0x6, /* FC_ALIGNM4 */	/* FC_SHORT */ /* FC_LONG */			0x39, /* FC_HYI	NG */ PER */	_
/* 868 */	0x38,	0x6, /* FC_ALIGNM4 */ 0x8, /* FC_LONG */		/* 938 */ /* 940 */	0xb,	0x39, /* FC_HYI 0x5b, 0x12, 0x0,	NG */ PER */ /* FC_UP	/* FC_ALIGNM8 */ /* FC_END */ */
/* 868 */ /* 870 */ FC EMB	0x38, 0x8, EDDED COMPLEX*	0x6, /* FC_ALIGNM4 */ 0x8, /* FC_LONG */ 0x4c,	/* FC_LONG */	/* 938 */ /* 940 */		0x39, /* FC_HYI 0x5b, 0x12, 0x0,	NG */ PER */ /* FC_UP	/* FC_ALIGNM8 */ /* FC_END */ */
/* 868 */ /* 870 */	0x38, 0x8, EDDED COMPLEX*	0x6, /* FC_ALIGNM4 */ 0x8, /* FC_LONG */ 0x4c, :/ /* 0 */	/* FC_LONG */ /*	/* 938 */ /* 940 */	0xb,	0x39, /* FC_HYI 0x5b, 0x12, 0x0, 2),	NG */ PER */ /* FC_UP /* Offset=	/* FC_ALIGNM8 */ /* FC_END */ */ -14 (928) */
/* 868 */ /* 870 */ FC EMB	0x38, 0x8, EDDED COMPLEX*	0x6, /* FC_ALIGNM4 */ 0x8, /* FC_LONG */ 0x4c, :/ /* 0 */	/* FC_LONG */	/* 938 */ /* 940 */ /* 942 */ /* 944 */	0xb, NdrFcShort(0xfffffff	0x39, /* FC_HYI 0x5b, 0x12, 0x0, 2), 0x12, 0x8,	NG */ PER */ /* FC_UP /* Offset= /* FC_UP	/* FC_ALIGNM8 */ /* FC_END */ */
/* 868 */ /* 870 */ FC EMB	0x38, 0x8, EDDED_COMPLEX * 0x0,	0x6, /* FC_ALIGNM4 */ 0x8, /* FC_LONG */ 0x4c, 5/ /* 0 */ NdrFcShort(0xfffffdf	/* FC_LONG */ /* 77), /* Offset=	/* 938 */ /* 940 */ /* 942 */	0xb, NdrFcShort(0xfffffff	0x39, /* FC_HYI 0x5b, 0x12, 0x0, 2), 0x12, 0x8, /* FC_CHA	NG */ PER */ /* FC_UP /* Offset= /* FC_UP AR */	/* FC_ALIGNM8 */ /* FC_END */ */ -14 (928) */ [simple_pointer] */
/* 868 */ /* 870 */ FC_EMB /* 872 */ -521 (352	0x38, 0x8, EDDED_COMPLEX * 0x0,	0x6, /* FC_ALIGNM4 */ 0x8, /* FC_LONG */ 0x4c, 5/ /* 0 */ NdrFcShort(0xfffffdf	/* FC_LONG */ /*	/* 938 */ /* 940 */ /* 942 */ /* 944 */ /* 946 */	0xb, NdrFcShort(0xfffffff	0x39, /* FC_HYI 0x5b, 0x12, 0x0, 2), 0x12, 0x8,	NG */ PER */ /* FC_UP /* Offset= /* FC_UP AR */	/* FC_ALIGNM8 */ /* FC_END */ */ -14 (928) */
/* 868 */ /* 870 */ FC_EMB /* 872 */	0x38, 0x8, EDDED_COMPLEX * 0x0,	0x6, /* FC_ALIGNM4 */ 0x8, /* FC_LONG */ 0x4c, */ /* 0 */ NdrFcShort(0xfffffdf	/* FC_LONG */ /* 77), /* Offset= /* FC_END */	/* 938 */ /* 940 */ /* 942 */ /* 944 */	0xb, NdrFcShort(0xfffffff	0x39, /* FC_HYI 0x5b, 0x12, 0x0, 2), 0x12, 0x8, /* FC_CH/ 0x5c,	NG */ PER */ /* FC_UP /* Offset= /* FC_UP AR */	/* FC_ALIGNM8 */ /* FC_END */ */ -14 (928) */ [simple_pointer] */ /* FC_PAD */
/* 868 */ /* 870 */ FC_EMB /* 872 */ -521 (352 /* 876 */	0x38, 0x8, EDDED_COMPLEX * 0x0,	0x6, /* FC_ALIGNM4 */ 0x8, /* FC_LONG */ 0x4c, */ /* 0 */ NdrFcShort(0xfffffdf 0x5b, 0x12, 0x0, /* FC_UP	/* FC_LONG */ /* 77), /* Offset= /* FC_END */ */	/* 938 */ /* 940 */ /* 942 */ /* 944 */ /* 946 */ /* 948 */	0xb, NdrFcShort(0xffffffff 0x2,	0x39, /* FC_HYI 0x5b, 0x12, 0x0, 2), 0x12, 0x8, /* FC_CHA	NG */ PER */ /* FC_UP /* Offset= /* FC_UP AR */	/* FC_ALIGNM8 */ /* FC_END */ */ -14 (928) */ [simple_pointer] */
/* 868 */ /* 870 */ FC_EMB /* 872 */ -521 (352 /* 876 */	0x38, 0x8, EDDED_COMPLEX * 0x0,	0x6, /* FC_ALIGNM4 */ 0x8, /* FC_LONG */ 0x4c, */ /* 0 */ NdrFcShort(0xfffffdf 0x5b, 0x12, 0x0, /* FC_UP	/* FC_LONG */ /* 77), /* Offset= /* FC_END */	/* 938 */ /* 940 */ /* 942 */ /* 944 */ /* 946 */ /* 948 */	0xb, NdrFcShort(0xfffffff	0x39, /* FC_HYI 0x5b, /* FC_CHYI 0x5b, /* 0x12, 0x0, 2), /* FC_CHZ 0x5c, /* FC_CHZ 0x5c, /* 0x1a,	NG */ PER */ /* FC_UP /* Offset= /* FC_UP AR */	/* FC_ALIGNM8 */ /* FC_END */ */ -14 (928) */ [simple_pointer] */ /* FC_PAD */ /*
/* 868 */ /* 870 */ FC_EMB /* 872 */ -521 (352 /* 876 */	0x38, 0x8, EDDED_COMPLEX * 0x0,	0x6, /* FC_ALIGNM4 */ 0x8, /* FC_LONG */ 0x4c, :// /* 0 */ NdrFcShort(0xfffffdf 0x5b, 0x12, 0x0, /* FC_UP f6), /* Offset=	/* FC_LONG */ /* 77), /* Offset= /* FC_END */ */ -266 (612) */	/* 938 */ /* 940 */ /* 942 */ /* 944 */ /* 946 */ FC_BOGI	0xb, NdrFcShort(0xfffffff 0x2, US_STRUCT */	0x39, /* FC_HYI 0x5b, /* FC_HYI 0x5b, 0x12, 0x0, 2), 0x12, 0x8, /* FC_CH/ 0x5c, 0x1a, 0x7,	NG */ PER */ /* FC_UP /* Offset= /* FC_UP AR */	/* FC_ALIGNM8 */ /* FC_END */ */ -14 (928) */ [simple_pointer] */ /* FC_PAD */
/* 868 */ /* 870 */ FC_EMB /* 872 */ -521 (352 /* 876 */ /* 880 */	0x38, 0x8, EDDED_COMPLEX * 0x0,) */ NdrFcShort(0xffffffet	0x6, /* FC_ALIGNM4 */ 0x8, /* FC_LONG */ 0x4c, /* 0 */ NdrFcShort(0xfffffdf 0x5b, 0x12, 0x0, /* FC_UP 16), /* Offset= 0x12, 0x8, /* FC_UP	/* FC_LONG */ /* 77), /* Offset= /* FC_END */ */ -266 (612) */	/* 938 */ /* 940 */ /* 942 */ /* 944 */ /* 946 */ FC_BOGU /* 950 */	0xb, NdrFcShort(0xffffffff 0x2, US_STRUCT */ NdrFcShort(0x20),	0x39, /* FC_HYI 0x5b, 0x12, 0x0, 2), 0x12, 0x8, /* FC_CHA 0x5c, 0x1a, 0x7, /* 32 */	NG */ PER */ /* FC_UP /* Offset= /* FC_UP AR */	/* FC_ALIGNM8 */ /* FC_END */ */ -14 (928) */ [simple_pointer] */ /* FC_PAD */ /*
/* 868 */ /* 870 */ FC_EMB /* 872 */ -521 (352 /* 876 */	0x38, 0x8, EDDED_COMPLEX * 0x0,) */ NdrFcShort(0xffffffet	0x6, /* FC_ALIGNM4 */ 0x8, /* FC_LONG */ 0x4c, */ /* 0 */ NdrFcShort(0xfffffdf 0x5b, 0x12, 0x0, /* FC_UP 16), /* Offset= 0x12, 0x8, /* FC_UP /* FC_BYTE */	/* FC_LONG */ /* 77), /* Offset= /* FC_END */ */ -266 (612) */ [simple_pointer] */	/* 938 */ /* 940 */ /* 942 */ /* 944 */ /* 946 */ FC_BOGU /* 950 */ /* 952 */	0xb, NdrFcShort(0xffffffff 0x2, US_STRUCT */ NdrFcShort(0x20), NdrFcShort(0x0),	0x39, /* FC_HYI 0x5b, 0x12, 0x0, 2), 0x12, 0x8, /* FC_CH/ 0x5c, 0x1a, 0x7, /* 32 */ /* 0 */	PER */ /* FC_UP /* Offset= /* FC_UP AR */	/* FC_ALIGNM8 */ /* FC_END */ */ -14 (928) */ [simple_pointer] */ /* FC_PAD */ /*
/* 868 */ /* 870 */ FC_EMB /* 872 */ -521 (352 /* 876 */ /* 888 */ /* 882 */	0x38, 0x8, EDDED_COMPLEX * 0x0,) */ NdrFcShort(0xffffffet	0x6, /* FC_ALIGNM4 */ 0x8, /* FC_LONG */ 0x4c, /* 0 */ NdrFcShort(0xfffffdf 0x5b, 0x12, 0x0, /* FC_UP 16), /* Offset= 0x12, 0x8, /* FC_UP	/* FC_LONG */ /* 77), /* Offset= /* FC_END */ */ -266 (612) */	/* 938 */ /* 940 */ /* 942 */ /* 944 */ /* 946 */ FC_BOGU /* 950 */ /* 952 */ /* 954 */	0xb, NdrFcShort(0xffffffff 0x2, US_STRUCT */ NdrFcShort(0x20), NdrFcShort(0x0), NdrFcShort(0x0),	0x39, /* FC_HYI 0x5b, 0x12, 0x0, 2), 0x12, 0x8, /* FC_CH/ 0x5c, 0x1a, 0x7, /* 32 */ /* 0 */ /* Offset=	NG */ PER */ /* FC_UP /* Offset= /* FC_UP AR */	/* FC_ALIGNM8 */ /* FC_END */ */ -14 (928) */ [simple_pointer] */ /* FC_PAD */ /*
/* 868 */ /* 870 */ FC_EMB /* 872 */ -521 (352 /* 876 */ /* 880 */	0x38, 0x8, EDDED_COMPLEX * 0x0,) */ NdrFcShort(0xffffffet	0x6, /* FC_ALIGNM4 */ 0x8, /* FC_LONG */ 0x4c, */ /* 0 */ NdrFcShort(0xfffffdf 0x5b, 0x12, 0x0, /* FC_UP 66), /* Offset= 0x12, 0x8, /* FC_UP /* FC_BYTE */ 0x5c,	/* FC_LONG */ /* 77), /* Offset= /* FC_END */ */ -266 (612) */ [simple_pointer] */ /* FC_PAD */	/* 938 */ /* 940 */ /* 942 */ /* 944 */ /* 946 */ FC_BOGU /* 950 */ /* 952 */	0xb, NdrFcShort(0xffffffff 0x2, US_STRUCT */ NdrFcShort(0x20), NdrFcShort(0x0), NdrFcShort(0x0),	0x39, /* FC_HYI 0x5b, 0x12, 0x0, 2), 0x12, 0x8, /* FC_CH/ 0x5c, 0x1a, 0x7, /* 32 */ /* 0 */ /* Offset= /* FC_LON	NG */ PER */ /* FC_UP /* Offset= /* FC_UP AR */ 0 (954) */ NG */	/* FC_ALIGNM8 */ /* FC_END */ */ -14 (928) */ [simple_pointer] */ /* FC_PAD */ /* /* 7 */
/* 868 */ /* 870 */ FC_EMB /* 872 */ -521 (352 /* 876 */ /* 880 */ /* 882 */ /* 884 */	0x38, 0x8, EDDED_COMPLEX * 0x0,) */ NdrFcShort(0xfffffet	0x6, /* FC_ALIGNM4 */ 0x8, /* FC_LONG */ 0x4c, */ /* 0 */ NdrFcShort(0xfffffdf 0x5b, 0x12, 0x0, /* FC_UP 76), /* Offset= 0x12, 0x8, /* FC_UP /* FC_BYTE */ 0x5c, 0x12, 0x8, /* FC_UP	/* FC_LONG */ /* 77), /* Offset= /* FC_END */ */ -266 (612) */ [simple_pointer] */ /* FC_PAD */	/* 938 */ /* 940 */ /* 942 */ /* 944 */ /* 946 */ FC_BOGU /* 950 */ /* 952 */ /* 954 */	0xb, NdrFcShort(0xffffffff 0x2, US_STRUCT */ NdrFcShort(0x20), NdrFcShort(0x0), NdrFcShort(0x0), 0x8,	0x39, /* FC_HYI 0x5b, 0x12, 0x0, 2), 0x12, 0x8, /* FC_CH/ 0x5c, 0x1a, 0x7, /* 32 */ /* 0 */ /* Offset= /* FC_LON 0x8,	NG */ PER */ /* FC_UP /* Offset= /* FC_UP AR */ 0 (954) */ NG */	/* FC_ALIGNM8 */ /* FC_END */ */ -14 (928) */ [simple_pointer] */ /* FC_PAD */ /*
/* 868 */ /* 870 */ FC_EMB /* 872 */ -521 (352 /* 876 */ /* 888 */ /* 882 */	0x38, 0x8, EDDED_COMPLEX * 0x0,) */ NdrFcShort(0xfffffet	0x6, /* FC_ALIGNM4 */ 0x8, /* FC_LONG */ 0x4c, */ /* 0 */ NdrFcShort(0xfffffdf 0x5b, 0x12, 0x0, /* FC_UP /* Offset= 0x12, 0x8, /* FC_UP /* FC_BYTE */ 0x5c, 0x12, 0x8, /* FC_UP /* FC_BYTE */	/* FC_LONG */ /* 77), /* Offset= /* FC_END */ */ -266 (612) */ [simple_pointer] */ /* FC_PAD */ [simple_pointer] */	/* 938 */ /* 940 */ /* 942 */ /* 944 */ /* 946 */ FC_BOGU /* 950 */ /* 952 */ /* 954 */	0xb, NdrFcShort(0xffffffff 0x2, US_STRUCT */ NdrFcShort(0x20), NdrFcShort(0x0), NdrFcShort(0x0),	0x39, /* FC_HYI 0x5b, 0x12, 0x0, 2), 0x12, 0x8, /* FC_CH/ 0x5c, 0x1a, 0x7, /* 32 */ /* 0 */ /* Offset= /* FC_LON 0x8, /* FC_SHO	NG */ PER */ /* FC_UP /* Offset= /* FC_UP AR */ 0 (954) */ NG */ DRT */	/* FC_ALIGNM8 */ /* FC_END */ */ -14 (928) */ [simple_pointer] */ /* FC_PAD */ /* /* 7 */ /* FC_LONG */
/* 868 */ /* 870 */ FC_EMB /* 872 */ -521 (352 /* 876 */ /* 880 */ /* 884 */ /* 886 */	0x38, 0x8, EDDED_COMPLEX * 0x0,) */ NdrFcShort(0xfffffet	0x6, /* FC_ALIGNM4 */ 0x8, /* FC_LONG */ 0x4c, */ /* 0 */ NdrFcShort(0xfffffdf 0x5b, 0x12, 0x0, /* FC_UP 76), /* Offset= 0x12, 0x8, /* FC_UP /* FC_BYTE */ 0x5c, 0x12, 0x8, /* FC_UP	/* FC_LONG */ /* 77), /* Offset= /* FC_END */ */ -266 (612) */ [simple_pointer] */ /* FC_PAD */	/* 938 */ /* 940 */ /* 942 */ /* 944 */ /* 946 */ FC_BOGI /* 950 */ /* 952 */ /* 954 */ /* 958 */	0xb, NdrFcShort(0xffffffff 0x2, US_STRUCT */ NdrFcShort(0x20), NdrFcShort(0x0), NdrFcShort(0x0), 0x8, 0x6,	0x39, /* FC_HYI 0x5b, /* FC_HYI 0x5b, /* FC_CHYI 0x5b, /* FC_CHYI 0x5c, /* FC_CHYI 0x5c, /* 32 */ /* 0 */ /* 0 */ /* 0 */ /* FC_LON 0x8, /* FC_SHC 0x6, /* FC_SHC 0x6,	PER */ /* FC_UP /* Offset= /* FC_UP AR */ 0 (954) */ NG */ ORT */	/* FC_ALIGNM8 */ /* FC_END */ */ -14 (928) */ [simple_pointer] */ /* FC_PAD */ /* /* 7 */
/* 868 */ /* 870 */ FC_EMB /* 872 */ -521 (352 /* 876 */ /* 880 */ /* 882 */ /* 884 */	0x38, 0x8, EDDED_COMPLEX * 0x0,) */ NdrFcShort(0xfffffet	0x6, /* FC_ALIGNM4 */ 0x8, /* FC_LONG */ 0x4c, */ /* 0 */ NdrFcShort(0xfffffdf 0x5b, 0x12, 0x0, /* FC_UP /6), /* Offset= 0x12, 0x8, /* FC_UP /* FC_BYTE */ 0x5c, 0x12, 0x8, /* FC_UP /* FC_SHORT */ 0x5c,	/* FC_LONG */ /* F7), /* Offset= /* FC_END */ */ -266 (612) */ [simple_pointer] */ /* FC_PAD */ [simple_pointer] */ /* FC_PAD */	/* 938 */ /* 940 */ /* 942 */ /* 944 */ /* 946 */ FC_BOGU /* 950 */ /* 952 */ /* 954 */	0xb, NdrFcShort(0xffffffff 0x2, US_STRUCT */ NdrFcShort(0x20), NdrFcShort(0x0), NdrFcShort(0x0), 0x8,	0x39, /* FC_HYI 0x5b, 0x12, 0x0, 2), 0x12, 0x8, /* FC_CH/ 0x5c, 0x1a, 0x7, /* 32 */ /* 0 */ /* Offset= /* FC_LON 0x8, /* FC_SHO 0x6, /* FC_SHO	PER */ /* FC_UP /* Offset= /* FC_UP AR */ 0 (954) */ NG */ ORT */	/* FC_ALIGNM8 */ /* FC_END */ */ -14 (928) */ [simple_pointer] */ /* FC_PAD */ /* /* 7 */ /* FC_LONG */ /* FC_SHORT */
/* 868 */ /* 870 */ FC_EMB /* 872 */ -521 (352 /* 876 */ /* 888 */ /* 884 */ /* 886 */ /* 888 */	0x38, 0x8, EDDED_COMPLEX* 0x0,)*/ NdrFcShort(0xfffffet 0x1, 0x6,	0x6, /* FC_ALIGNM4 */ 0x8, /* FC_LONG */ 0x4c, */ /* 0 */ NdrFcShort(0xfffffdf 0x5b, 0x12, 0x0, /* FC_UP //6), /* Offset= 0x12, 0x8, /* FC_UP /* FC_BYTE */ 0x5c, 0x12, 0x8, /* FC_UP /* FC_SHORT */ 0x5c, 0x12, 0x8, /* FC_UP	/* FC_LONG */ /* F7), /* Offset= /* FC_END */ */ -266 (612) */ [simple_pointer] */ /* FC_PAD */ [simple_pointer] */ /* FC_PAD */	/* 938 */ /* 940 */ /* 942 */ /* 944 */ /* 946 */ /* 948 */ FC_BOGU /* 950 */ /* 954 */ /* 956 */ /* 958 */ /* 960 */	0xb, NdrFcShort(0xffffffff 0x2, US_STRUCT */ NdrFcShort(0x20), NdrFcShort(0x0), NdrFcShort(0x0), 0x8, 0x6, 0x6,	0x39, /* FC_HYI 0x5b, /* FC_HYI 0x5b, /* FC_CH/ 0x5c, /* FC_CH/ 0x5c, /* 32 */ /* 0*/ /* Offset= /* FC_LOn 0x8, /* FC_SHC 0x6,	PER */ /* FC_UP /* Offset= /* FC_UP AR */ 0 (954) */ DRT */ DRT */	/* FC_ALIGNM8 */ /* FC_END */ */ -14 (928) */ [simple_pointer] */ /* FC_PAD */ /* /* 7 */ /* FC_LONG */ /* FC_SHORT */
/* 868 */ /* 870 */ FC_EMB /* 872 */ -521 (352 /* 876 */ /* 880 */ /* 884 */ /* 886 */	0x38, 0x8, EDDED_COMPLEX* 0x0,)*/ NdrFcShort(0xfffffet 0x1, 0x6,	0x6, /* FC_ALIGNM4 */ 0x8, /* FC_LONG */ 0x4c, */ /* 0 */ NdrFcShort(0xfffffdf 0x5b, 0x12, 0x0, /* FC_UP /* Offset= 0x12, 0x8, /* FC_UP /* FC_BYTE */ 0x5c, 0x12, 0x8, /* FC_UP /* FC_SHORT */ 0x5c, 0x12, 0x8, /* FC_UP /* FC_SHORT */ 0x5c,	/* FC_LONG */ /* FT), /* Offset= /* FC_END */ */ -266 (612) */ [simple_pointer] */ /* FC_PAD */ [simple_pointer] */ /* FC_PAD */ [simple_pointer] */	/* 938 */ /* 940 */ /* 942 */ /* 944 */ /* 946 */ FC_BOGI /* 950 */ /* 952 */ /* 956 */ /* 958 */	0xb, NdrFcShort(0xffffffff 0x2, US_STRUCT */ NdrFcShort(0x20), NdrFcShort(0x0), NdrFcShort(0x0), 0x8, 0x6,	0x39, /* FC_HYI 0x5b, /* FC_HYI 0x5b, /* FC_CH/ 0x5c, /* FC_CH/ 0x5c, /* 32 */ /* 0*/ /* Offset= /* FC_LOn 0x8, /* FC_SHC 0x6, /* FC_SHC 0x6, /* FC_EMI	PER */ /* FC_UP /* Offset= /* FC_UP AR */ 0 (954) */ DRT */ DRT */ BEDDED_	/* FC_ALIGNM8 */ /* FC_END */ */ -14 (928) */ [simple_pointer] */ /* FC_PAD */ /* /* 7 */ /* FC_LONG */ /* FC_SHORT */ COMPLEX */
/* 868 */ /* 870 */ FC_EMB /* 872 */ -521 (352 /* 876 */ /* 888 */ /* 884 */ /* 888 */ /* 8890 */	0x38, 0x8, EDDED_COMPLEX* 0x0,)*/ NdrFcShort(0xfffffet 0x1, 0x6,	0x6, /* FC_ALIGNM4 */ 0x8, /* FC_LONG */ 0x4c, */ /* 0 */ NdrFcShort(0xfffffdf 0x5b, 0x12, 0x0, /* FC_UP /* Offset= 0x12, 0x8, /* FC_UP /* FC_BYTE */ 0x5c, 0x12, 0x8, /* FC_UP /* FC_SHORT */ 0x5c, 0x12, 0x8, /* FC_UP /* FC_SHORT */ 0x5c,	/* FC_LONG */ /* F7), /* Offset= /* FC_END */ */ -266 (612) */ [simple_pointer] */ /* FC_PAD */ [simple_pointer] */ /* FC_PAD */	/* 938 */ /* 940 */ /* 942 */ /* 944 */ /* 946 */ /* 948 */ FC_BOGI /* 950 */ /* 952 */ /* 956 */ /* 958 */ /* 960 */ /* 962 */	0xb, NdrFcShort(0xffffffff 0x2, US_STRUCT */ NdrFcShort(0x20), NdrFcShort(0x0), NdrFcShort(0x0), 0x8, 0x6, 0x6, 0x4c,	0x39, /* FC_HYI 0x5b, 0x12, 0x0, 2), 0x12, 0x8, /* FC_CH/ 0x5c, 0x1a, 0x7, /* 32 */ /* Offset= /* FC_LON 0x8, /* FC_SHO 0x6, /* FC_EMI 0x0,	NG */ PER */ /* FC_UP /* Offset= /* FC_UP AR */ 0 (954) */ ORT */ DRT */ BEDDED_	/* FC_ALIGNM8 */ /* FC_END */ */ -14 (928) */ [simple_pointer] */ /* FC_PAD */ /* /* 7 */ /* FC_LONG */ /* FC_SHORT */ COMPLEX */ /* 0 */
/* 868 */ /* 870 */ FC_EMB /* 872 */ -521 (352 /* 876 */ /* 888 */ /* 884 */ /* 886 */ /* 888 */	0x38, 0x8, EDDED_COMPLEX* 0x0,)*/ NdrFcShort(0xfffffet 0x1, 0x6,	0x6, /* FC_ALIGNM4 */ 0x8, /* FC_LONG */ 0x4c, /* 0 */ NdrFcShort(0xfffffdf 0x5b, 0x12, 0x0, /* FC_UP /6), /* Offset= 0x12, 0x8, /* FC_UP /* FC_BYTE */ 0x5c, 0x12, 0x8, /* FC_UP /* FC_SHORT */ 0x5c, 0x12, 0x8, /* FC_UP /* FC_SHORT */ 0x5c,	/* FC_LONG */ /* /* 7), /* Offset= /* FC_END */ */ -266 (612) */ [simple_pointer] */ /* FC_PAD */ [simple_pointer] */ /* FC_PAD */ [simple_pointer] */ /* FC_PAD */	/* 938 */ /* 940 */ /* 942 */ /* 944 */ /* 946 */ /* 948 */ FC_BOGI /* 950 */ /* 952 */ /* 954 */ /* 960 */ /* 962 */	0xb, NdrFcShort(0xffffffff 0x2, US_STRUCT */ NdrFcShort(0x20), NdrFcShort(0x0), NdrFcShort(0x0), 0x8, 0x6, 0x6, 0x4c, NdrFcShort(0xfffffc4	0x39, /* FC_HYI 0x5b, 0x12, 0x0, 2), 0x12, 0x8, /* FC_CH/ 0x5c, 0x1a, 0x7, /* 32 */ /* 0 */ /* Offset= /* FC_LON 0x8, /* FC_SHO 0x6, /* FC_EMI 0x0, 12),	NG */ PER */ /* FC_UP /* Offset= /* FC_UP AR */ 0 (954) */ ORT */ DRT */ BEDDED_ /* Offset=	/* FC_ALIGNM8 */ /* FC_END */ */ -14 (928) */ [simple_pointer] */ /* FC_PAD */ /* /* 7 */ /* FC_LONG */ /* FC_SHORT */ COMPLEX */
/* 868 */ /* 870 */ FC_EMB /* 872 */ -521 (352 /* 876 */ /* 888 */ /* 884 */ /* 888 */ /* 8890 */ /* 892 */	0x38, 0x8, EDDED_COMPLEX * 0x0,) */ NdrFcShort(0xffffffet 0x1, 0x6, 0x8,	0x6, /* FC_ALIGNM4 */ 0x8, /* FC_LONG */ 0x4c, /* 0 */ NdrFcShort(0xfffffdf 0x5b, 0x12, 0x0, /* FC_UP /6), /* Offset= 0x12, 0x8, /* FC_UP /* FC_BYTE */ 0x5c, 0x12, 0x8, /* FC_UP /* FC_SHORT */ 0x5c, 0x12, 0x8, /* FC_UP /* FC_LONG */ 0x5c, 0x12, 0x8, /* FC_UP	/* FC_LONG */ /* /* 7), /* Offset= /* FC_END */ */ -266 (612) */ [simple_pointer] */ /* FC_PAD */ [simple_pointer] */ /* FC_PAD */ [simple_pointer] */ /* FC_PAD */	/* 938 */ /* 940 */ /* 942 */ /* 944 */ /* 946 */ /* 948 */ FC_BOGI /* 950 */ /* 952 */ /* 956 */ /* 958 */ /* 960 */ /* 962 */	0xb, NdrFcShort(0xffffffff 0x2, US_STRUCT */ NdrFcShort(0x20), NdrFcShort(0x0), NdrFcShort(0x0), 0x8, 0x6, 0x6, 0x4c, NdrFcShort(0xfffffc4	0x39, /* FC_HYI 0x5b, 0x12, 0x0, 2), 0x12, 0x8, /* FC_CH 0x5c, 0x1a, 0x7, /* 32 */ /* 0 */ /* Offset= /* FC_LON 0x8, /* FC_SHO 0x6, /* FC_EMI 0x0, 12), /* FC_PAL	NG */ PER */ /* FC_UP /* Offset= /* FC_UP AR */ 0 (954) */ ORT */ DRT */ BEDDED_ /* Offset=	/* FC_ALIGNM8 */ /* FC_END */ */ -14 (928) */ [simple_pointer] */ /* FC_PAD */ /* 7 */ /* FC_LONG */ /* FC_SHORT */ COMPLEX */ -958 (6) */
/* 868 */ /* 870 */ FC_EMB /* 872 */ -521 (352 /* 876 */ /* 888 */ /* 884 */ /* 888 */ /* 8890 */	0x38, 0x8, EDDED_COMPLEX * 0x0,) */ NdrFcShort(0xffffffet 0x1, 0x6, 0x8,	0x6, /* FC_ALIGNM4 */ 0x8, /* FC_LONG */ 0x4c, /* 0 */ NdrFcShort(0xfffffdf 0x5b, 0x12, 0x0, /* FC_UP /6), /* Offset= 0x12, 0x8, /* FC_UP /* FC_BYTE */ 0x5c, 0x12, 0x8, /* FC_UP /* FC_SHORT */ 0x5c, 0x12, 0x8, /* FC_UP /* FC_LONG */ 0x5c, 0x12, 0x8, /* FC_UP /* FC_LONG */ 0x5c,	/* FC_LONG */ /* /* Offset= /* FC_END */ */ -266 (612) */ [simple_pointer] */ /* FC_PAD */ [simple_pointer] */ /* FC_PAD */ [simple_pointer] */ /* FC_PAD */ [simple_pointer] */	/* 938 */ /* 940 */ /* 942 */ /* 944 */ /* 946 */ /* 948 */ FC_BOGI /* 950 */ /* 952 */ /* 956 */ /* 960 */ /* 962 */ /* 966 */	0xb, NdrFcShort(0xffffffff 0x2, US_STRUCT */ NdrFcShort(0x20), NdrFcShort(0x0), NdrFcShort(0x0), 0x6, 0x6, 0x6, 0x4c, NdrFcShort(0xfffffc2 0x5c,	0x39, /* FC_HYI 0x5b, 0x12, 0x0, 2), 0x12, 0x8, /* FC_CH 0x5c, 0x1a, 0x7, /* 32 */ /* 0 */ /* Offset= /* FC_LON 0x8, /* FC_SHC 0x6, /* FC_SHC 0x0, 42), /* FC_PAI 0x5b,	NG */ PER */ /* FC_UP /* Offset= /* FC_UP AR */ 0 (954) */ ORT */ DRT */ BEDDED_ /* Offset= D*/	/* FC_ALIGNM8 */ /* FC_END */ */ -14 (928) */ [simple_pointer] */ /* FC_PAD */ /* 7 */ /* FC_LONG */ /* FC_SHORT */ COMPLEX */ /* 0 */ -958 (6) */ /* FC_END */
/* 868 */ /* 870 */ FC_EMB /* 872 */ -521 (352 /* 876 */ /* 888 */ /* 884 */ /* 888 */ /* 890 */ /* 892 */	0x38, 0x8, EDDED_COMPLEX * 0x0,) */ NdrFcShort(0xffffffet 0x1, 0x6, 0x8,	0x6, /* FC_ALIGNM4 */ 0x8, /* FC_LONG */ 0x4c, /* 0 */ NdrFcShort(0xfffffdf 0x5b, 0x12, 0x0, /* FC_UP /6), /* Offset= 0x12, 0x8, /* FC_UP /* FC_BYTE */ 0x5c, 0x12, 0x8, /* FC_UP /* FC_SHORT */ 0x5c, 0x12, 0x8, /* FC_UP /* FC_LONG */ 0x5c, 0x12, 0x8, /* FC_UP /* FC_LONG */ 0x5c,	/* FC_LONG */ /* /* 7), /* Offset= /* FC_END */ */ -266 (612) */ [simple_pointer] */ /* FC_PAD */ [simple_pointer] */ /* FC_PAD */ [simple_pointer] */ /* FC_PAD */	/* 938 */ /* 940 */ /* 942 */ /* 944 */ /* 946 */ /* 948 */ FC_BOGI /* 950 */ /* 952 */ /* 954 */ /* 960 */ /* 962 */	0xb, NdrFcShort(0xffffffff 0x2, US_STRUCT */ NdrFcShort(0x20), NdrFcShort(0x0), NdrFcShort(0x0), 0x8, 0x6, 0x6, 0x4c, NdrFcShort(0xfffffc4	0x39, /* FC_HYI 0x5b, 0x12, 0x0, 2), 0x12, 0x8, /* FC_CH/ 0x5c, 0x1a, 0x7, /* 32 */ /* 0 */ /* Offset= /* FC_LON 0x6, /* FC_SHO 0x6, /* FC_EMI 0x0, /* FC_PAI 0x5b, /* FC_USE	NG */ PER */ /* FC_UP /* Offset= /* FC_UP AR */ 0 (954) */ ORT */ DRT */ BEDDED_ /* Offset= D*/	/* FC_ALIGNM8 */ /* FC_END */ */ -14 (928) */ [simple_pointer] */ /* FC_PAD */ /* 7 */ /* FC_LONG */ /* FC_SHORT */ COMPLEX */ -958 (6) */ /* FC_END */ HAL */
/* 868 */ /* 870 */ FC_EMB /* 872 */ -521 (352 /* 876 */ /* 888 */ /* 884 */ /* 888 */ /* 8890 */ /* 892 */	0x38, 0x8, EDDED_COMPLEX * 0x0,) */ NdrFcShort(0xffffffet 0x1, 0x6, 0x8,	0x6, /* FC_ALIGNM4 */ 0x8, /* FC_LONG */ 0x4c, /* 0 */ NdrFcShort(0xfffffdff 0x5b, 0x12, 0x0, /* FC_UP /6), /* Offset= 0x12, 0x8, /* FC_UP /* FC_BYTE */ 0x5c, 0x12, 0x8, /* FC_UP /* FC_SHORT */ 0x5c, 0x12, 0x8, /* FC_UP /* FC_LONG */ 0x5c, 0x12, 0x8, /* FC_UP /* FC_LONG */ 0x5c,	/* FC_LONG */ /* /* Offset= /* FC_END */ */ -266 (612) */ [simple_pointer] */ /* FC_PAD */	/* 938 */ /* 940 */ /* 942 */ /* 944 */ /* 946 */ /* 948 */ FC_BOGU /* 950 */ /* 952 */ /* 954 */ /* 960 */ /* 962 */ /* 968 */	0xb, NdrFcShort(0xffffffff 0x2, US_STRUCT */ NdrFcShort(0x20), NdrFcShort(0x0), NdrFcShort(0x0), 0x6, 0x6, 0x4c, NdrFcShort(0xfffffc2 0x5c, 0xb4,	0x39, /* FC_HYI 0x5b, 0x12, 0x8, /* FC_CH/ 0x5c, 0x12, 0x8, /* FC_CH/ 0x5c, 0x1a, 0x7, /* 32 */ /* 0 */ /* Offset= /* FC_LON 0x8, /* FC_SHC 0x6, /* FC_SHC 0x0, 42), /* FC_PAI 0x5b, /* FC_USE 0x83,	NG */ PER */ /* FC_UP /* Offset= /* FC_UP AR */ 0 (954) */ ORT */ DRT */ BEDDED_ /* Offset= D*/	/* FC_ALIGNM8 */ /* FC_END */ */ -14 (928) */ [simple_pointer] */ /* FC_PAD */ /* 7 */ /* FC_LONG */ /* FC_SHORT */ COMPLEX */ /* 0 */ -958 (6) */ /* FC_END */
/* 868 */ /* 870 */ FC_EMB /* 872 */ -521 (352 /* 876 */ /* 888 */ /* 884 */ /* 888 */ /* 890 */ /* 892 */	0x38, 0x8, EDDED_COMPLEX * 0x0,) */ NdrFcShort(0xfffffet 0x1, 0x6, 0x8,	0x6, /* FC_ALIGNM4 */ 0x8, /* FC_LONG */ 0x4c, /* 0 */ NdrFcShort(0xfffffdf 0x5b, 0x12, 0x0, /* FC_UP /6), /* Offset= 0x12, 0x8, /* FC_UP /* FC_BYTE */ 0x5c, 0x12, 0x8, /* FC_UP /* FC_SHORT */ 0x5c, 0x12, 0x8, /* FC_UP /* FC_LONG */ 0x5c, 0x12, 0x8, /* FC_UP /* FC_LONG */ 0x5c,	/* FC_LONG */ /* /* Offset= /* FC_END */ */ -266 (612) */ [simple_pointer] */ /* FC_PAD */	/* 938 */ /* 940 */ /* 942 */ /* 944 */ /* 946 */ /* 948 */ FC_BOGU /* 950 */ /* 952 */ /* 954 */ /* 960 */ /* 962 */ /* 968 */	0xb, NdrFcShort(0xffffffff 0x2, US_STRUCT */ NdrFcShort(0x20), NdrFcShort(0x0), NdrFcShort(0x0), 0x6, 0x6, 0x6, 0x4c, NdrFcShort(0xfffffc4 0x5c, 0xb4, NdrFcShort(0x0),	0x39, /* FC_HYI 0x5b, 0x12, 0x0, 2), 0x12, 0x8, /* FC_CH/ 0x5c, 0x1a, 0x7, /* 32 */ /* 0 */ /* Offset= /* FC_LON 0x6, /* FC_SHO 0x6, /* FC_EMI 0x0, /* FC_PAI 0x5b, /* FC_USE	NG */ PER */ /* FC_UP /* Offset= /* FC_UP AR */ 0 (954) */ ORT */ DRT */ BEDDED_ /* Offset= D*/	/* FC_ALIGNM8 */ /* FC_END */ */ -14 (928) */ [simple_pointer] */ /* FC_PAD */ /* 7 */ /* FC_LONG */ /* FC_SHORT */ COMPLEX */ -958 (6) */ /* FC_END */ HAL */

```
/* 974 */ NdrFcShort( 0x0 ), /* 0 */
/* 976 */
          NdrFcShort( 0xfffffc32 ),
                                          /* Offset= -974 (2) */
/* 978 */
                               0x11, 0x4, /* FC_RP [alloced_on_stack] */
/* 980 */ NdrFcShort( 0x6 ),
                               /* Offset= 6 (986) */
/* 982 */
                               0x13, 0x0, /* FC OP */
/* 984 */ NdrFcShort( 0xffffffdc ),
                                         /* Offset= -36 (948) */
                               /* FC_USER_MARSHAL */
/* 986 */ 0xb4,
                                                    /* 131 */
                               0x83.
/* 988 */ NdrFcShort( 0x0 ),
                               /* 0 */
/* 990 */ NdrFcShort( 0x10 ), /* 16 */
/* 992 */ NdrFcShort( 0x0 ),
                               /* 0 */
/* 994 */ NdrFcShort( 0xfffffff4 ),
                                          /* Offset= -12 (982) */
                               0x0
const CInterfaceProxyVtbl * tpcc com ps ProxyVtblList[] =
  (CInterfaceProxyVtbl *) & ITPCCProxyVtbl,
  0
};
const CInterfaceStubVtbl * _tpcc_com_ps_StubVtblList[] =
  ( CInterfaceStubVtbl *) &_ITPCCStubVtbl,
  0
};
PCInterfaceName const _tpcc_com_ps_InterfaceNamesList[] =
  "ITPCC".
  0
};
#define _tpcc_com_ps_CHECK_IID(n)
                                          IID_GENERIC_CHECK_IID(
_tpcc_com_ps, pIID, n)
int __stdcall _tpcc_com_ps_IID_Lookup( const IID * pIID, int * pIndex )
  if(! tpcc com ps CHECK IID(0))
    *pIndex = 0;
    return 1;
    }
  return 0;
}
const ExtendedProxyFileInfo tpcc com ps ProxyFileInfo =
  (PCInterfaceProxyVtblList *) & _tpcc_com_ps_ProxyVtblList,
  (PCInterfaceStubVtblList *) & tpcc com ps StubVtblList,
  (const PCInterfaceName * ) & _tpcc_com_ps_InterfaceNamesList,
  0, // no delegation
  & _tpcc_com_ps_IID_Lookup,
  1.
  0, /* table of [async_uuid] interfaces */
  0, /* Filler1 */
  0, /* Filler2 */
  0 /* Filler3 */
};
```

```
#endif /* !defined( M IA64) && !defined( M AXP64)*/
#pragma warning( disable: 4049 ) /* more than 64k source lines */
/* this ALWAYS GENERATED file contains the proxy stub code */
/* File created by MIDL compiler version 5.03.0280 */
/* at Sat Apr 08 16:40:10 2000
/* Compiler settings for .\src\tpcc com ps.idl:
 Oicf (OptLev=i2), W1, Zp8, env=Win64 (32b run,appending), ms_ext, c_ext,
  error checks: allocation ref bounds_check enum stub_data
  VC __declspec() decoration level:
      declspec(uuid()), __declspec(selectany), __declspec(novtable)
    DECLSPEC_UUID(), MIDL_INTERFACE()
//@@MIDL_FILE_HEADING( )
#if defined(_M_IA64) || defined(_M_AXP64)
#define USE STUBLESS PROXY
/* verify that the <rpcproxy.h> version is high enough to compile this file*/
#ifndef __REDQ_RPCPROXY_H_VERSION
#define REQUIRED RPCPROXY H VERSION 475
#endif
#include "rpcproxy.h"
#ifndef RPCPROXY H VERSION
#error this stub requires an updated version of <rpcproxy.h>
#endif // RPCPROXY H VERSION
#include "tpcc_com_ps.h"
#define TYPE FORMAT STRING SIZE 979
#define PROC_FORMAT_STRING_SIZE 253
#define TRANSMIT AS TABLE SIZE 0
#define WIRE_MARSHAL_TABLE_SIZE 1
typedef struct _MIDL_TYPE_FORMAT_STRING
  unsigned char Format[ TYPE_FORMAT_STRING_SIZE ];
  } MIDL TYPE FORMAT STRING;
typedef struct MIDL PROC FORMAT STRING
  short
           Pad:
  unsigned char Format[ PROC FORMAT STRING SIZE ];
  } MIDL_PROC_FORMAT_STRING;
extern const MIDL TYPE FORMAT STRING MIDL TypeFormatString;
extern const MIDL_PROC_FORMAT_STRING __MIDL_ProcFormatString;
/* Standard interface: __MIDL_itf_tpcc_com_ps_0000, ver. 0.0,
x00}} */
```

```
/* Object interface: IUnknown, ver. 0.0,
                                                                        &ITPCC_ServerInfo,
0, /* pure interpreted */
0x46}} */
                                                                        CStdStubBuffer_METHODS
/* Object interface: ITPCC, ver. 0.0,
                                                                      extern const USER MARSHAL ROUTINE QUADRUPLE
                                                                      UserMarshalRoutines[ WIRE_MARSHAL_TABLE_SIZE ];
GUID={0xFEEE6AA2,0x84B1,0x11d2,{0xBA,0x47,0x00,0xC0,0x4F,0xBF,0x
E0,0x8B}} */
                                                                      static const MIDL STUB DESC Object StubDesc =
                                                                        Ò.
extern const MIDL STUB DESC Object StubDesc;
                                                                        NdrOleAllocate,
                                                                        NdrOleFree,
                                                                        0,
extern const MIDL_SERVER_INFO ITPCC_ServerInfo;
                                                                        0,
                                                                        0,
#pragma code_seg(".orpc")
                                                                        0,
static const unsigned short ITPCC_FormatStringOffsetTable[] =
                                                                          MIDL TypeFormatString.Format,
                                                                        1, /* -error bounds_check flag */
  0,
  44,
                                                                        0x50002, /* Ndr library version */
  88.
                                                                        0x5030118, /* MIDL Version 5.3.280 */
  132,
  176,
  220
                                                                        UserMarshalRoutines.
                                                                        0, /* notify & notify flag routine table */
                                                                        0x1, /* MIDL flag */
static const MIDL SERVER INFO ITPCC ServerInfo =
                                                                        0, /* Reserved3 */
                                                                        0, /* Reserved4 */
  &Object StubDesc,
                                                                        0 /* Reserved5 */
  0,
    MIDL ProcFormatString.Format,
  &ITPCC FormatStringOffsetTable[-3],
                                                                      #pragma data seg(".rdata")
  0,
                                                                      static const USER MARSHAL ROUTINE QUADRUPLE
  0,
                                                                      UserMarshalRoutines[ WIRE_MARSHAL_TABLE_SIZE ] =
  0
  };
static const MIDL STUBLESS PROXY INFO ITPCC ProxyInfo =
                                                                             VARIANT UserSize
                                                                             ,VARIANT UserMarshal
  &Object StubDesc,
                                                                             ,VARIANT UserUnmarshal
    MIDL_ProcFormatString.Format,
                                                                             ,VARIANT_UserFree
  &ITPCC FormatStringOffsetTable[-3],
  0,
                                                                          };
  0
  };
                                                                      #if !defined( RPC WIN64 )
CINTERFACE_PROXY_VTABLE(9) _ITPCCProxyVtbl =
                                                                      #error Invalid build platform for this stub.
                                                                      #endif
  &ITPCC_ProxyInfo,
  &IID ITPCC,
                                                                      static const MIDL PROC FORMAT STRING MIDL ProcFormatString =
  IUnknown_QueryInterface_Proxy,
                                                                         {
  IUnknown_AddRef_Proxy,
                                                                          0.
  IUnknown Release Proxy
  (void *)-1 /* ITPCC::NewOrder */,
  (void *)-1 /* ITPCC::Payment */,
                                                                                /* Procedure NewOrder */
  (void *)-1 /* ITPCC::Delivery */,
  (void *)-1 /* ITPCC::StockLevel */.
                                                                                                   0x33,
  (void *)-1 /* ITPCC::OrderStatus */,
                                                                      FC_AUTO_HANDLE */
  (void *)-1 /* ITPCC::CallSetComplete */
                                                                                                                      /* Old Flags: object,
                                                                                                   0x6c.
                                                                      Oi2 */
                                                                      /* 2 */
                                                                                NdrFcLong( 0x0 ),
                                                                                                   /* 0 */
const CInterfaceStubVtbl ITPCCStubVtbl =
                                                                      /* 6 */
                                                                                NdrFcShort(0x3),
                                                                                                  /* 3 */
                                                                      #ifndef ALPHA
  &IID ITPCC,
                                                                      /* 8 */
                                                                               NdrFcShort( 0x38 ), /* ia64 Stack size/offset = 56 */
```

```
/* 58 */ 0x47,
#else
                                                                                                                 /* Oi2 Flags: srv must size, clt must size, has
                                 NdrFcShort(0x30), /* axp64 Stack
                                                                                return, has ext, */
size/offset = 48 */
                                                                                                                 0x3
                                                                                                                                       /* 3 */
                                                                                /* 60 */
#endif
                                                                                           0xa,
                                                                                                                 /* 10 */
/* 10 */
          NdrFcShort( 0x0 ),
                                /* 0 */
                                                                                                                                       /* Ext Flags: new corr
                                                                                                                 0x7
/* 12 */
          NdrFcShort( 0x8 ),
                                 /* 8 */
                                                                                desc, clt corr check, srv corr check, */
/* 14 */
          0x47,
                                 /* Oi2 Flags: srv must size, clt must size, has
                                                                                /* 62 */
                                                                                           NdrFcShort( 0x20 ), /* 32 */
                                                                                /* 64 */
return, has ext, */
                                                                                           NdrFcShort( 0x20 ), /* 32 */
                                                                                /* 66 */
                                                                                           NdrFcShort( 0x0 ), /* 0 */
                                 0x3,
                                                      /* 3 */
                                 /* 10 */
/* 16 */ 0xa,
                                                                                /* 68 */
                                                                                           NdrFcShort( 0x0 ), /* 0 */
                                 0x7,
                                                      /* Ext Flags: new corr
desc, clt corr check, srv corr check, */
                                                                                           /* Parameter txn in */
/* 18 */
          NdrFcShort( 0x20 ), /* 32 */
/* 20 */
          NdrFcShort( 0x20 ), /* 32 */
                                                                                /* 70 */
                                                                                           NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by val, */
/* 22 */
          NdrFcShort( 0x0 ), /* 0 */
                                                                                #ifndef ALPHA
/* 24 */
          NdrFcShort( 0x0 ),
                               /* 0 */
                                                                                /* 72 */
                                                                                           NdrFcShort(0x10), /* ia64 Stack size/offset = 16 */
                                                                                #else
           /* Parameter txn_in */
                                                                                                                 NdrFcShort( 0x8 ), /* axp64 Stack
                                                                                size/offset = 8 */
/* 26 */
          NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by val, */
                                                                                #endif
                                                                                           NdrFcShort( 0x3b6 ), /* Type Offset=950 */
#ifndef _ALPHA
                                                                                /* 74 */
/* 28 */
          NdrFcShort(0x10), /* ia64 Stack size/offset = 16 */
#else
                                                                                           /* Parameter txn out */
                                 NdrFcShort( 0x8 ), /* axp64 Stack
size/offset = 8 */
                                                                                /* 76 */
                                                                                           NdrFcShort( 0x6113 ),
                                                                                                                            /* Flags: must size, must free, out,
#endif
                                                                                simple ref, srv alloc size=24 */
                                                                                #ifindef _ALPHA_ /* 78 */ NdrFcShort( 0x28 ), /* ia64 Stack size/offset = 40 */
/* 30 */
          NdrFcShort( 0x3b6 ), /* Type Offset=950 */
           /* Parameter txn out */
                                                                                #else
                                                                                                                 NdrFcShort(0x20), /* axp64 Stack
/* 32 */
          NdrFcShort( 0x6113 ),
                                           /* Flags: must size, must free, out,
                                                                                size/offset = 32 */
simple ref, srv alloc size=24 */
                                                                                #endif
#ifndef _ALPHA
                                                                                /* 80 */
                                                                                           NdrFcShort( 0x3c8 ), /* Type Offset=968 */
/* 34 */
          NdrFcShort( 0x28 ), /* ia64 Stack size/offset = 40 */
                                                                                           /* Return value */
#else
                                 NdrFcShort( 0x20 ), /* axp64 Stack
                                                                                /* 82 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
size/offset = 32 */
#endif
                                                                                #ifndef ALPHA
          NdrFcShort( 0x3c8 ), /* Type Offset=968 */
                                                                                /* 84 */
                                                                                          NdrFcShort( 0x30 ), /* ia64 Stack size/offset = 48 */
/* 36 */
                                                                                #else
           /* Return value */
                                                                                                                 NdrFcShort(0x28), /* axp64 Stack
                                                                                size/offset = 40 */
/* 38 */
          NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
                                                                                #endif
                                                                                /* 86 */
#ifndef _ALPHA
                                                                                           0x8
                                                                                                                 /* FC_LONG */
          NdrFcShort(0x30), /* ia64 Stack size/offset = 48 */
                                                                                                                                       /* 0 */
                                                                                                                 0x0,
#else
                                 NdrFcShort( 0x28 ), /* axp64 Stack
                                                                                           /* Procedure Delivery */
size/offset = 40 */
                                                                                /* 88 */
                                                                                                                 /* FC_AUTO HANDLE */
#endif
                                                                                           0x33.
/* 42 */
                                 /* FC LONG */
          0x8,
                                                                                                                 0x6c,
                                                                                                                                       /* Old Flags: object,
                                                      /* 0 */
                                                                                Oi2 */
                                 0x0.
                                                                                /* 90 */
                                                                                                                 /* 0 */
                                                                                           NdrFcLong( 0x0),
                                                                                /* 94 */
           /* Procedure Payment */
                                                                                           NdrFcShort( 0x5 ),
                                                                                                                 /* 5 */
                                                                                #ifndef _ALPHA_
/* 44 */
                                 /* FC AUTO HANDLE */
          0x33,
                                                                                /* 96 */
                                                                                           NdrFcShort( 0x38 ), /* ia64 Stack size/offset = 56 */
                                                      /* Old Flags: object,
                                                                                #else
                                 0x6c,
Oi2 */
                                                                                                                 NdrFcShort( 0x30 ), /* axp64 Stack
                                                                                size/offset = 48 */
/* 46 */
          NdrFcLong( 0x0 ),
                                /* 0 */
/* 50 */
          NdrFcShort( 0x4 ),
                                /* 4 */
                                                                                #endif
                                                                                /* 98 */
                                                                                                                 /* 0 */
#ifndef _ALPHA
                                                                                           NdrFcShort( 0x0 ),
                                                                                /* 100 */ NdrFcShort( 0x8 ),
/* 52 */<sup>-</sup>
          NdrFcShort(0x38), /* ia64 Stack size/offset = 56 */
                                                                                                                 /* 8 */
#else
                                                                                /* 102 */ 0x47,
                                                                                                                 /* Oi2 Flags: srv must size, clt must size, has
                                 NdrFcShort( 0x30 ), /* axp64 Stack
                                                                                return, has ext, */
size/offset = 48 */
                                                                                                                 0x3,
                                                                                                                                       /* 3 */
                                                                                /* 104 */ 0xa.
                                                                                                                 /* 10 */
#endif
/* 54 */
          NdrFcShort( 0x0 ),
                               /* 0 */
                                                                                                                 0x7,
                                                                                                                                       /* Ext Flags: new corr
/* 56 */
          NdrFcShort( 0x8 ),
                                /* 8 */
                                                                                desc, clt corr check, srv corr check, */
                                                                                /* 106 */ NdrFcShort( 0x20 ), /* 32 */
```

```
/* 108 */ NdrFcShort( 0x20 ), /* 32 */
                                                                               #ifndef ALPHA
/* 110 */ NdrFcShort( 0x0 ), /* 0 */
                                                                               /* 160 */ NdrFcShort( 0x10 ), /* ia64 Stack size/offset = 16 */
/* 112 */ NdrFcShort( 0x0 ), /* 0 */
                                                                                                               NdrFcShort( 0x8 ), /* axp64 Stack
                                                                               size/offset = 8 */
          /* Parameter txn_in */
                                                                               #endif
/* 114 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by val, */
                                                                               /* 162 */ NdrFcShort( 0x3b6 ), /* Type Offset=950 */
#ifndef ALPHA
/* 116 */ NdrFcShort( 0x10 ), /* ia64 Stack size/offset = 16 */
                                                                                         /* Parameter txn out */
#else
                                NdrFcShort( 0x8 ), /* axp64 Stack
                                                                               /* 164 */ NdrFcShort( 0x6113 ),
                                                                                                                          /* Flags: must size, must free, out,
size/offset = 8 */
                                                                               simple ref, srv alloc size=24 */
                                                                               #ifndef ALPHA
/* 118 */ NdrFcShort( 0x3b6 ), /* Type Offset=950 */
                                                                               /* 166 * \overline{/} NdrFcShort(0x28), /* ia64 Stack size/offset = 40 */
                                                                               #else
           /* Parameter txn out */
                                                                                                               NdrFcShort( 0x20 ), /* axp64 Stack
                                                                               size/offset = 32 */
/* 120 */ NdrFcShort( 0x6113 ),
                                           /* Flags: must size, must free, out,
                                                                               #endif
simple ref, srv alloc size=24 */
                                                                               /* 168 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */
#ifndef ALPHA
/* 122 */ NdrFcShort( 0x28 ), /* ia64 Stack size/offset = 40 */
                                                                                         /* Return value */
#else
                                NdrFcShort( 0x20 ), /* axp64 Stack
                                                                               /* 170 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
size/offset = 32 */
                                                                               #ifndef ALPHA
                                                                               /* 172 */ NdrFcShort( 0x30 ), /* ia64 Stack size/offset = 48 */
#endif
/* 124 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */
                                                                               #else
                                                                                                               NdrFcShort( 0x28 ), /* axp64 Stack
          /* Return value */
                                                                               size/offset = 40 */
                                                                               #endif
/* 126 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
                                                                               /* 174 */ 0x8
                                                                                                               /* FC LONG */
#ifndef ALPHA
                                                                                                               0x0,
                                                                                                                                     /* 0 */
/* 128 */ NdrFcShort( 0x30 ), /* ia64 Stack size/offset = 48 */
                                                                                         /* Procedure OrderStatus */
#else
                                NdrFcShort(0x28), /* axp64 Stack
size/offset = 40 */
                                                                               /* 176 */ 0x33,
                                                                                                               /* FC_AUTO_HANDLE */
                                                                                                                                     /* Old Flags: object,
#endif
                                                                                                               0x6c,
/* 130 */ 0x8.
                                                                               Oi2 */
                                /* FC LONG */
                                0x0,
                                                      /* 0 */
                                                                               /* 178 */ NdrFcLong( 0x0 ),
                                                                                                              /* () */
                                                                               /* 182 */ NdrFcShort( 0x7 ),
                                                                                                               /* 7 */
          /* Procedure StockLevel */
                                                                               #ifndef _ALPHA_
                                                                               /* 184 */ NdrFcShort( 0x38 ), /* ia64 Stack size/offset = 56 */
/* 132 */ 0x33,
                                /* FC_AUTO_HANDLE */
                                                                               #else
                                                                                                               NdrFcShort( 0x30 ), /* axp64 Stack
                                                      /* Old Flags: object,
                                0x6c,
                                                                               size/offset = 48 */
Oi2 */
/* 134 */ NdrFcLong( 0x0 ),
                                /* 0 */
                                                                               #endif
                                                                               /* 186 */ NdrFcShort( 0x0 ),
/* 138 */ NdrFcShort( 0x6 ),
                                /* 6 */
                                                                                                               /* 0 */
#ifndef _ALPHA_
                                                                               /* 188 */ NdrFcShort( 0x8 ),
                                                                                                               /* 8 */
/* 140 */ NdrFcShort( 0x38 ), /* ia64 Stack size/offset = 56 */
                                                                               /* 190 */ 0x47,
                                                                                                               /* Oi2 Flags: srv must size, clt must size, has
                                                                               return, has ext, */
#else
                                NdrFcShort( 0x30 ), /* axp64 Stack
                                                                                                                                     /* 3 */
                                                                                                               0x3,
                                                                                                               /* 10 */
                                                                               /* 192 */ 0xa,
size/offset = 48 */
#endif
                                                                                                               0x7,
                                                                                                                                     /* Ext Flags: new corr
/* 142 */ NdrFcShort( 0x0 ),
                                /* 0 */
                                                                               desc, clt corr check, srv corr check, */
/* 144 */ NdrFcShort( 0x8 ),
                                /* 8 */
                                                                               /* 194 */ NdrFcShort( 0x20 ), /* 32 */
/* 146 */ 0x47,
                                                                               /* 196 */ NdrFcShort( 0x20 ), /* 32 */
                                /* Oi2 Flags: srv must size, clt must size, has
                                                                               /* 198 */ NdrFcShort( 0x0 ), /* 0 */
return, has ext, */
                                                      /* 3 */
                                                                               /* 200 */ NdrFcShort( 0x0 ),
                                0x3,
                                /* 10 */
/* 148 */ 0xa,
                                0x7,
                                                      /* Ext Flags: new corr
                                                                                         /* Parameter txn in */
desc, clt corr check, srv corr check, */
/* 150 */ NdrFcShort( 0x20 ), /* 32 */
                                                                               /* 202 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by val, */
/* 152 */ NdrFcShort( 0x20 ), /* 32 */
                                                                               #ifndef ALPHA
/* 154 */ NdrFcShort( 0x0 ), /* 0 */
                                                                               /* 204 */ NdrFcShort( 0x10 ), /* ia64 Stack size/offset = 16 */
/* 156 */ NdrFcShort( 0x0 ),
                                                                               #else
                                                                                                               NdrFcShort( 0x8 ), /* axp64 Stack
          /* Parameter txn in */
                                                                               size/offset = 8 */
                                                                               #endif
/* 158 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by val, */
                                                                               /* 206 */ NdrFcShort( 0x3b6 ), /* Type Offset=950 */
```

```
/* 8 */
                                                                                                              /* Corr desc: FC USHORT */
                                                                                        0x7,
          /* Parameter txn out */
                                                                                                              0x0,
                                                                                        NdrFcShort( 0xfff8 ), /* -8 */
                                                                              /* 10 */
                                                                              /* 12 */
                                                                                                             /* Corr flags: early, */
/* 208 */ NdrFcShort( 0x6113 ),
                                          /* Flags: must size, must free, out,
                                                                                        NdrFcShort(0x1),
simple ref, srv alloc size=24 */
                                                                              /* 14 */
                                                                                        NdrFcShort( 0x2 ),
                                                                                                             /* Offset= 2 (16) */
#ifndef ALPHA
                                                                              /* 16 */
                                                                                        NdrFcShort( 0x10 ), /* 16 */
/* 210 */ NdrFcShort( 0x28 ), /* ia64 Stack size/offset = 40 */
                                                                              /* 18 */
                                                                                        NdrFcShort( 0x2b ), /* 43 */
                                                                              /* 20 */
#else
                                                                                        NdrFcLong(0x3),
                                                                              /* 24 */
                                NdrFcShort(0x20), /* axp64 Stack
                                                                                        NdrFcShort( 0x8008 ),
                                                                                                                        /* Simple arm type: FC LONG */
size/offset = 32 */
                                                                              /* 26 */
                                                                                        NdrFcLong( 0x11 ), /* 17 */
                                                                              /* 30 */
#endif
                                                                                        NdrFcShort( 0x8001 ),
                                                                                                                        /* Simple arm type: FC_BYTE */
/* 212 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */
                                                                              /* 32 */
                                                                                        NdrFcLong(0x2),
                                                                              /* 36 */
                                                                                        NdrFcShort( 0x8006),
                                                                                                                        /* Simple arm type: FC SHORT */
                                                                              /* 38 */
          /* Return value */
                                                                                        NdrFcLong(0x4),
                                                                              /* 42 */
                                                                                        NdrFcShort( 0x800a ),
                                                                                                                        /* Simple arm type: FC_FLOAT */
                                                                              /* 44 */
                                                                                        NdrFcLong(0x5),
                                                                                                              /* 5 */
/* 214 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifndef ALPHA
                                                                              /* 48 */
                                                                                        NdrFcShort( 0x800c),
                                                                                                                        /* Simple arm type: FC DOUBLE
/* 216 */ NdrFcShort( 0x30 ), /* ia64 Stack size/offset = 48 */
                                                                              /* 50 */
                                                                                        NdrFcLong( 0xb ),
                                                                                                            /* 11 */
#else
                                NdrFcShort(0x28), /* axp64 Stack
                                                                              /* 54 */
                                                                                        NdrFcShort( 0x8006 ),
                                                                                                                        /* Simple arm type: FC SHORT */
size/offset = 40 */
                                                                              /* 56 */
                                                                                                              /* 10 */
                                                                                        NdrFcLong( 0xa ),
                                                                              /* 60 */
                                                                                        NdrFcShort( 0x8008 ),
                                                                                                                        /* Simple arm type: FC LONG */
#endif
/* 218 */ 0x8.
                                /* FC LONG */
                                                                              /* 62 */
                                                                                        NdrFcLong( 0x6 ),
                                                     /* 0 */
                                                                              /* 66 */
                                                                                        NdrFcShort( 0xd6 ), /* Offset= 214 (280) */
                                0x0.
                                                                                                             /* 7 */
                                                                              /* 68 */
                                                                                        NdrFcLong( 0x7 ),
           /* Procedure CallSetComplete */
                                                                              /* 72 */
                                                                                        NdrFcShort( 0x800c),
                                                                                                                        /* Simple arm type: FC_DOUBLE
                                /* FC_AUTO_HANDLE */
                                                                              /* 74 */
/* 220 */ 0x33,
                                                                                        NdrFcLong( 0x8 ),
                                0x6c,
                                                     /* Old Flags: object,
                                                                              /* 78 */
                                                                                        NdrFcShort( 0xd0 ), /* Offset= 208 (286) */
Oi2 */
                                                                              /* 80 */
                                                                                        NdrFcLong( 0xd ),
                                                                                                             /* 13 */
/* 222 */
          NdrFcLong(0x0),
                                /* 0 */
                                                                              /* 84 */
                                                                                        NdrFcShort( 0xe4 ), /* Offset= 228 (312) */
                                                                              /* 86 */
                                                                                                             /* 9 */
/* 226 */
                                /* 8 */
          NdrFcShort(0x8),
                                                                                        NdrFcLong( 0x9 ),
/* 228 */
          NdrFcShort( 0x10 ),
                               /* ia64, axp64 Stack size/offset = 16 */
                                                                              /* 90 */
                                                                                        NdrFcShort( 0xf0 ). /* Offset= 240 (330) */
/* 230 */
          NdrFcShort( 0x0 ),
                                /* 0 */
                                                                              /* 92 */
                                                                                        NdrFcLong(0x2000),
                                                                                                                        /* 8192 */
                                                                                        NdrFcShort( 0xfc ), /* Offset= 252 (348) */
/* 232 */
          NdrFcShort( 0x8 ),
                               /* 8 */
                                                                              /* 96 */
/* 234 */ 0x44,
                                /* Oi2 Flags: has return, has ext, */
                                                                              /* 98 */
                                                                                        NdrFcLong( 0x24 ), /* 36 */
                                                                                        NdrFcShort( 0x2f4 ), /* Offset= 756 (858) */
                                                                              /* 102 */
                                0x1
                                                     /* 1 */
                                /* 10 */
                                                                              /* 104 */
/* 236 */ 0xa,
                                                                                        NdrFcLong(0x4024),
                                                                                                                        /* 16420 */
                                                                              /* 108 */
                                                                                        NdrFcShort( 0x2ee ), /* Offset= 750 (858) */
                                0x1,
                                                     /* Ext Flags: new corr
                                                                              /* 110 */
                                                                                        NdrFcLong(0x4011),
                                                                                                                        /* 16401 */
desc, */
/* 238 */
          NdrFcShort(0x0),
                                /* 0 */
                                                                              /* 114 */
                                                                                        NdrFcShort( 0x2ec ), /* Offset= 748 (862) */
/* 240 */
                                /* 0 */
                                                                              /* 116 */
          NdrFcShort( 0x0 ),
                                                                                        NdrFcLong(0x4002),
                                                                                                                        /* 16386 */
/* 242 */
          NdrFcShort( 0x0 ),
                                /* 0 */
                                                                              /* 120 */
                                                                                        NdrFcShort( 0x2ea ), /* Offset= 746 (866) */
                                /* 0 */
                                                                              /* 122 */
/* 244 */
          NdrFcShort( 0x0 ),
                                                                                        NdrFcLong(0x4003),
                                                                                                                        /* 16387 */
                                                                              /* 126 */
                                                                                        NdrFcShort( 0x2e8 ), /* Offset= 744 (870) */
                                                                              /* 128 */
           /* Return value */
                                                                                        NdrFcLong( 0x4004 ),
                                                                                                                        /* 16388 */
                                                                              /* 132 */
                                                                                        NdrFcShort( 0x2e6 ), /* Offset= 742 (874) */
/* 246 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
                                                                              /* 134 */
                                                                                        NdrFcLong(0x4005),
                                                                                                                        /* 16389 */
                               /* ia64, axp64 Stack size/offset = 8 */
                                                                                        NdrFcShort( 0x2e4 ), /* Offset= 740 (878) */
/* 248 */
          NdrFcShort( 0x8 ),
                                                                              /* 138 */
/* 250 */ 0x8,
                                /* FC LONG */
                                                                              /* 140 */
                                                                                        NdrFcLong(0x400b),
                                                                                                                        /* 16395 */
                                                                                        NdrFcShort( 0x2d2 ), /* Offset= 722 (866) */
                                                     /* 0 */
                                                                              /* 144 */
                                0x0,
                                                                                        NdrFcLong(0x400a),
                                                                              /* 146 */
                                                                                                                        /* 16394 */
                                                                              /* 150 */
                                                                                        NdrFcShort( 0x2d0 ), /* Offset= 720 (870) */
                                0x0
                                                                              /* 152 */
                                                                                        NdrFcLong(0x4006),
                                                                                                                        /* 16390 */
    }
                                                                                        NdrFcShort( 0x2d6 ), /* Offset= 726 (882) */
                                                                              /* 156 */
                                                                                        NdrFcLong( 0x4007),
                                                                              /* 158 */
                                                                                                                        /* 16391 */
static const MIDL TYPE FORMAT STRING MIDL TypeFormatString =
                                                                              /* 162 */
                                                                                        NdrFcShort( 0x2cc ), /* Offset= 716 (878) */
                                                                              /* 164 */
                                                                                        NdrFcLong(0x4008),
                                                                                                                        /* 16392 */
    0,
                                                                                        NdrFcShort( 0x2ce ), /* Offset= 718 (886) */
                                                                              /* 168 */
                                                                              /* 170 */
                                                                                        NdrFcLong( 0x400d ),
                                                                                                                        /* 16397 */
                                NdrFcShort( 0x0 ), /* 0 */
                                                                              /* 174 */
                                                                                        NdrFcShort( 0x2cc ), /* Offset= 716 (890) */
/* 2.*/
                                                                              /* 176 */
                                                                                        NdrFcLong(0x4009),
                                                                                                                        /* 16393 */
                                                                                        NdrFcShort( 0x2ca ), /* Offset= 714 (894) */
                                0x12, 0x0, /* FC UP */
                                                                              /* 180 */
/* 4 */
          NdrFcShort( 0x39e ), /* Offset= 926 (930) */
                                                                              /* 182 */
                                                                                        NdrFcLong( 0x6000 ),
                                                                                                                        /* 24576 */
                                                                              /* 186 */
                                                                                        NdrFcShort( 0x2c8 ), /* Offset= 712 (898) */
/* 6 */
                                0x2b,
                                                                              /* 188 */
                                                                                        NdrFcLong( 0x400c),
                                                                                                                        /* 16396 */
FC_NON_ENCAPSULATED_UNION */
                                                                                        NdrFcShort( 0x2c6 ), /* Offset= 710 (902) */
                                                                              /* 192 */
                                                                              /* 194 */ NdrFcLong( 0x10 ), /* 16 */
                                0x9
                                                     /* FC ULONG */
```

```
/* 198 */ NdrFcShort( 0x8002 ),
                                                                                                                                  /* 0 */
                                          /* Simple arm type: FC_CHAR */
                                                                                                             0x0
          NdrFcLong( 0x12 ), /* 18 */
                                                                              /* 328 */ 0x0,
                                                                                                             /* 0 */
/* 200 */
/* 204 */
          NdrFcShort( 0x8006),
                                          /* Simple arm type: FC SHORT */
                                                                                                             0x46,
                                                                                                                                  /* 70 */
/* 206 */
          NdrFcLong( 0x13 ), /* 19 */
                                                                              /* 330 */
/* 210 */
          NdrFcShort( 0x8008 ),
                                          /* Simple arm type: FC_LONG */
                                                                                                             0x2f.
                                                                                                                                  /* FC IP */
          NdrFcLong( 0x16 ), /* 22 */
/* 212 */
                                                                                                             0x5a,
/* 216 */
          NdrFcShort( 0x8008 ),
                                                                              FC CONSTANT IID */
                                          /* Simple arm type: FC LONG */
/* 218 */
          NdrFcLong( 0x17 ), /* 23 */
                                                                              /* 332 */ NdrFcLong( 0x20400 ),
                                                                                                                        /* 132096 */
                                                                             /* 336 */ NdrFcShort( 0x0 ),
                                                                                                             /* 0 */
/* 222 */
          NdrFcShort( 0x8008),
                                          /* Simple arm type: FC LONG */
/* 224 */
          NdrFcLong( 0xe ), /* 14 */
                                                                              /* 338 */ NdrFcShort( 0x0 ),
                                                                                                             /* 0 */
          NdrFcShort( 0x2aa ), /* Offset= 682 (910) */
                                                                              /* 340 */ 0xc0,
                                                                                                             /* 192 */
/* 228 */
/* 230 */
          NdrFcLong( 0x400e ),
                                          /* 16398 */
                                                                                                             0x0.
                                                                                                                                  /* 0 */
          NdrFcShort( 0x2b0 ), /* Offset= 688 (922) */
/* 234 */
                                                                              /* 342 */ 0x0,
                                                                                                             /* 0 */
/* 236 */
          NdrFcLong( 0x4010 ),
                                          /* 16400 */
                                                                                                             0x0
                                                                                                                                  /* 0 */
/* 240 */
          NdrFcShort( 0x2ae ), /* Offset= 686 (926) */
                                                                              /* 344 */ 0x0,
                                                                                                             /* 0 */
/* 242 */
          NdrFcLong(0x4012),
                                          /* 16402 */
                                                                                                             0x0,
                                                                                                                                  /* 0 */
          NdrFcShort( 0x26c ), /* Offset= 620 (866) */
                                                                                                             /* 0 */
/* 246 */
                                                                              /* 346 */ 0x0,
/* 248 */
          NdrFcLong( 0x4013 ),
                                          /* 16403 */
                                                                                                             0x46,
                                                                                                                                  /* 70 */
/* 252 */
          NdrFcShort( 0x26a ), /* Offset= 618 (870) */
                                                                              /* 348 */
/* 254 */
          NdrFcLong(0x4016),
                                          /* 16406 */
                                                                                                             0x12, 0x10,
                                                                                                                                  /* FC UP
          NdrFcShort( 0x264 ), /* Offset= 612 (870) */
/* 258 */
                                                                              [pointer_deref] */
/* 260 */
          NdrFcLong(0x4017),
                                                                              /* 350 */ NdrFcShort( 0x2 ),
                                                                                                             /* Offset= 2 (352) */
                                          /* 16407 */
/* 264 */
          NdrFcShort( 0x25e ), /* Offset= 606 (870) */
                                                                              /* 352 */
/* 266 */
          NdrFcLong( 0x0 ), /* 0 */
                                                                                                             0x12, 0x0, /* FC UP */
/* 270 */
          NdrFcShort( 0x0 ),
                               /* Offset= 0 (270) */
                                                                              /* 354 */ NdrFcShort( 0x1e6 ), /* Offset= 486 (840) */
/* 272 */
          NdrFcLong(0x1),
                               /* 1 */
                                                                              /* 356 */
/* 276 */
          NdrFcShort( 0x0 ),
                               /* Offset = 0 (276) */
                                                                                                             0x2a,
/* 278 */
                                                                             FC_ENCAPSULATED_UNION */
          NdrFcShort( 0xffffffff ),
                                          /* Offset= -1 (277) */
/* 280 */
                                                                                                             0x89.
                                                                                                                                  /* 137 */
                                                    /* FC STRUCT */
                                                                              /* 358 */ NdrFcShort( 0x20 ), /* 32 */
                               0x15
                               0x7,
                                                     /* 7 */
                                                                              /* 360 */
                                                                                        NdrFcShort( 0xa ),
                                                                                                             /* 10 */
                               /* 8 */
                                                                              /* 362 */
                                                                                                             /* 8 */
/* 282 */ NdrFcShort( 0x8 ),
                                                                                        NdrFcLong( 0x8 ),
/* 284 */ 0xb.
                               /* FC HYPER */
                                                                              /* 366 */
                                                                                        NdrFcShort( 0x50 ), /* Offset= 80 (446) */
                                                     /* FC END */
                                                                              /* 368 */
                                                                                        NdrFcLong( 0xd ), /* 13 */
                               0x5b.
/* 286 */
                                                                              /* 372 */
                                                                                        NdrFcShort( 0x70 ), /* Offset= 112 (484) */
                               0x12, 0x0, /* FC UP */
                                                                              /* 374 */
                                                                                        NdrFcLong( 0x9 ), /* 9 */
                                                                              /* 378 */
                                                                                        NdrFcShort( 0x90 ), /* Offset= 144 (522) */
/* 288 */ NdrFcShort( 0xe ),
                               /* Offset= 14 (302) */
                                                                              /* 380 */
                                                                                        NdrFcLong( 0xc ), /* 12 */
/* 290 */
                                                    /* FC_CARRAY */
                                                                              /* 384 */
                                                                                        NdrFcShort( 0xb0 ), /* Offset= 176 (560) */
                               0x1b.
                                                     /* 1 */
                                                                              /* 386 */
                                                                                        NdrFcLong( 0x24 ), /* 36 */
                               0x1
                               /* 2 */
                                                                                        NdrFcShort( 0x104 ), /* Offset= 260 (650) */
/* 292 */ NdrFcShort( 0x2 ),
                                                                              /* 390 */
                               /* Corr desc: FC_ULONG */
                                                                              /* 392 */
                                                                                        NdrFcLong( 0x800d ),
/* 294 */ 0x9,
                                                                                                                        /* 32781 */
                                                                              /* 396 */
                                                                                        NdrFcShort( 0x120 ), /* Offset= 288 (684) */
                               0x0.
                                                                              /* 398 */
                                                                                        NdrFcLong( 0x10 ), /* 16 */
/* 296 */ NdrFcShort( 0xfffc ), /* -4 */
                               /* Corr flags: early, */
                                                                              /* 402 */
                                                                                        NdrFcShort( 0x13a ), /* Offset= 314 (716) */
/* 298 */ NdrFcShort( 0x1 ),
/* 300 */ 0x6,
                                                                              /* 404 */
                                                                                        NdrFcLong( 0x2 ), /* 2 */
                               /* FC_SHORT */
                               0x5b.
                                                     /* FC END */
                                                                              /* 408 */
                                                                                        NdrFcShort( 0x150 ), /* Offset= 336 (744) */
/* 302 */
                                                                              /* 410 */
                                                                                        NdrFcLong( 0x3 ), /* 3 */
                                                                              /* 414 */
                                                                                        NdrFcShort( 0x166 ), /* Offset= 358 (772) */
                               0x17,
                                                     /* FC_CSTRUCT */
                                                     /* 3 */
                                                                              /* 416 */
                                                                                        NdrFcLong( 0x14 ), /* 20 */
                               0x3,
                                                                                        NdrFcShort( 0x17c ), /* Offset= 380 (800) */
/* 304 */ NdrFcShort( 0x8 ),
                               /* 8 */
                                                                              /* 420 */
/* 306 */ NdrFcShort( 0xfffffff0 ),
                                                                              /* 422 */
                                          /* Offset= -16 (290) */
                                                                                       NdrFcShort( 0xffffffff ),
                                                                                                                        /* Offset= -1 (421) */
/* 308 */ 0x8,
                               /* FC_LONG */
                                                                              /* 424 */
                                                     /* FC LONG */
                                                                                                             0x21,
                               0x8.
                               /* FC PAD */
                                                                             FC BOGUS ARRAY */
/* 310 */ 0x5c,
                                                     /* FC_END */
                                                                                                                                  /* 3 */
                               0x5b,
                                                                                                             0x3.
/* 312 */
                                                                              /* 426 */ NdrFcShort( 0x0 ),
                                                                                                             /* 0 */
                                                                              /* 428 */ 0x19,
                               0x2f
                                                     /* FC_IP */
                                                                                                             /* Corr desc: field pointer, FC_ULONG */
                               0x5a,
                                                                                                             0x0,
                                                                              /* 430 */
                                                                                                             /* 0 */
FC_CONSTANT_IID */
                                                                                       NdrFcShort( 0x0 ),
                                                                                                             /* Corr flags: early, */
/* 314 */ NdrFcLong( 0x0 ),
                               /* 0 */
                                                                              /* 432 */
                                                                                        NdrFcShort( 0x1 ),
                                                                              /* 434 */
/* 318 */
          NdrFcShort( 0x0 ),
                               /* 0 */
                                                                                        NdrFcLong( 0xffffffff ),
/* 320 */ NdrFcShort( 0x0 ),
                               /* 0 */
                                                                              /* 438 */
                                                                                        NdrFcShort( 0x0 ), /* Corr flags: */
/* 322 */ 0xc0,
                               /* 192 */
                                                                              /* 440 */
                               0x0,
                                                    /* 0 */
                                                                                                             0x12, 0x0, /* FC UP */
                               /* 0 */
/* 324 */ 0x0,
                                                                              /* 442 */ NdrFcShort( 0xffffff74 ),
                                                                                                                        /* Offset= -140 (302) */
                                                                              /* 444 */ 0x5c,
                                                                                                             /* FC_PAD */
                                                     /* 0 */
                               0x0,
/* 326 */ 0x0,
                               /* 0 */
                                                                                                             0x5b,
                                                                                                                                  /* FC END */
```

			•			
/* 446 */	0.1	/ 14	/* 522 */	0.26	0x39,	/* FC_ALIGNM8 */
EC DOCUE STRUCT */	0x1a,	/*	/* 532 * /	0X36,	/* FC_POINTER */	/* EC_END */
FC_BOGUS_STRUCT */	0v2	/* 3 */	/* 534 */		0x5b,	/* FC_END */
/* 448 */ NdrFcShort(0x10	0x3, , /* 16 */	7.3.7	/ 334 /		0x11, 0x0, /* FC RP	*/
/* 450 */ NdrFcShort(0x0),	/* 0 */		/* 536 */	NdrFcShort(0xffffffd		-36 (500) */
/* 452 */ NdrFcShort(0x6),	/* Offset= 6 (458) */		/* 538 */	ruit conord oxititie	ic), / Offset	-30 (300) 7
/* 454 */ 0x8,	/* FC_LONG */		7 336 7		0x21,	/*
7 15 1 7 OAG,	0x39,	/* FC_ALIGNM8 */	FC BOG	US_ARRAY */	0.21,	,
/* 456 */ 0x36,	/* FC POINTER */	,			0x3,	/* 3 */
	0x5b,	/* FC END */	/* 540 */	NdrFcShort(0x0),	/* 0 */	
/* 458 */		_	/* 542 */		/* Corr desc: field po	inter, FC ULONG */
	0x11, 0x0, /* FC_RF) */			0x0,	/* */
/* 460 */ NdrFcShort(0xffff	ffdc), /* Offset=	= -36 (424) */	/* 544 */	NdrFcShort(0x0),	/ * 0 * /	
/* 462 */			/* 546 */	\ //	/* Corr flags: early, *	4/
	0x21,	/*	/* 548 */	<u>U</u> \		
FC_BOGUS_ARRAY */			/* 552 */	NdrFcShort(0x0),	/* Corr flags: */	
	0x3,	/* 3 */	/* 554 */			
/* 464 */ NdrFcShort(0x0),	/* 0 */				0x12, 0x0, /* FC_UP	
/* 466 */ 0x19,	-	ointer, FC_ULONG */		NdrFcShort(0x176),		*/
/* 460 */ NIE CL ((0.0)	0x0,	/* */	/* 558 */	0x5c,	/* FC_PAD */	/* EC END */
/* 468 */ NdrFcShort(0x0),	/* 0 */	sk /	14.500 41		0x5b,	/* FC_END */
/* 470 */ NdrFcShort(0x1),		*/	/* 560 */		01-	/*
/* 472 */ NdrFcLong(0xffff			EC DOC	UC CTDUCT */	0x1a,	/ *
/* 476 */ NdrFcShort(0x0), /* 478 */ 0x4c,	/* Corr flags: */ /* FC EMBEDDED	COMPLEY */	LC_POG	US_STRUCT */	0x3,	/* 3 */
/ 4/8 / 0x4c,	0x0,		/* 562 */	NdrFcShort(0x10),	/* 16 */	1.3.1
/* 480 */ NdrFcShort(0xffff		= -168 (312) */		NdrFcShort(0x10),	/* 0 */	
/* 482 */ 0x5c,	/* FC PAD */	100 (312)	/* 566 */		/* Offset= 6 (572) */	
, 102 , 01100,	0x5b,	/* FC END */	/* 568 */		/* FC_LONG */	
/* 484 */	******,	,	, , ,	*****,	0x39,	/* FC ALIGNM8 */
	0x1a,	/*	/* 570 * /	0x36,	/* FC POINTER */	· -
FC_BOGUS_STRUCT */	,			,	0x5b,	/* FC_END */
	0x3,	/* 3 */	/* 572 */			_
/* 486 */ NdrFcShort(0x10	, /* 16 */				0x11, 0x0, /* FC RP	*/
/* 488 */ NdrFcShort(0x0),	/* 0 */		/* 574 */	NdrFcShort(0xffffffc		-36 (538) */
/* 490 */ NdrFcShort(0x6),	/* Offset= 6 (496) */		/* 574 */ /* 576 */	NdrFcShort(0xffffffc	lc), /* Offset=	-36 (538) */
` //	/* Offset= 6 (496) */ /* FC_LONG */			NdrFcShort(0xffffffc	0x2f, /* Offset=	-36 (538) */ /* FC_IP */
/* 490 */ NdrFcShort(0x6), /* 492 */ 0x8,	/* Offset= 6 (496) */ /* FC_LONG */ 0x39,	/* FC_ALIGNM8 */	/* 576 */		lc), /* Offset=	-36 (538) */
/* 490 */ NdrFcShort(0x6),	/* Offset= 6 (496) */ /* FC_LONG */ 0x39, /* FC_POINTER */	/* FC_ALIGNM8 */	/* 576 */ FC_CON	STANT_IID */	0x2f, 0x5a,	-36 (538) */ /* FC_IP */
/* 490 */ NdrFcShort(0x6), /* 492 */ 0x8, /* 494 */ 0x36,	/* Offset= 6 (496) */ /* FC_LONG */ 0x39,		/* 576 */ FC_CON /* 578 */	STANT_IID */ NdrFcLong(0x2f),	0x2f, 0x5a, /* 47 */	-36 (538) */ /* FC_IP */
/* 490 */ NdrFcShort(0x6), /* 492 */ 0x8,	/* Offset= 6 (496) */ /* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b,	/* FC_ALIGNM8 */ /* FC_END */	/* 576 */ FC_CON /* 578 */ /* 582 */	STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0),	0x2f, 0x5a, /* 47 */ /* 0 */	-36 (538) */ /* FC_IP */
/* 490 */ NdrFcShort(0x6), /* 492 */ 0x8, /* 494 */ 0x36, /* 496 */	/* Offset= 6 (496) */ /* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RF	/* FC_ALIGNM8 */ /* FC_END */	/* 576 */ FC_CON /* 578 */ /* 582 */ /* 584 */	STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0),	0x2f, 0x5a, /* 47 */ /* 0 */ /* 0 */	-36 (538) */ /* FC_IP */
/* 490 */ NdrFcShort(0x6), /* 492 */ 0x8, /* 494 */ 0x36, /* 496 */ /* 498 */ NdrFcShort(0xffff	/* Offset= 6 (496) */ /* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RF	/* FC_ALIGNM8 */ /* FC_END */	/* 576 */ FC_CON /* 578 */ /* 582 */	STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0),	0x2f, 0x5a, /* 47 */ /* 0 */ /* 192 */	-36 (538) */ /* FC_IP */ /*
/* 490 */ NdrFcShort(0x6), /* 492 */ 0x8, /* 494 */ 0x36, /* 496 */	/* Offset= 6 (496) */ /* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RF ffdc), /* Offset=	/* FC_ALIGNM8 */ /* FC_END */ 0 */ = -36 (462) */	/* 576 */ FC_CON /* 578 */ /* 582 */ /* 584 */	STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0xc0,	0x2f, 0x5a, /* 47 */ /* 0 */ /* 192 */ 0x0,	-36 (538) */ /* FC_IP */
/* 490 */ NdrFcShort(0x6), /* 492 */ 0x8, /* 494 */ 0x36, /* 496 */ /* 498 */ NdrFcShort(0xffff /* 500 */	/* Offset= 6 (496) */ /* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RF	/* FC_ALIGNM8 */ /* FC_END */	/* 576 */ FC_CON /* 578 */ /* 582 */ /* 584 */	STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0xc0,	0x2f, 0x5a, /* 47 */ /* 0 */ /* 192 */ 0x0, /* 0 */	-36 (538) */ /* FC_IP */ /* /* 0 */
/* 490 */ NdrFcShort(0x6), /* 492 */ 0x8, /* 494 */ 0x36, /* 496 */ /* 498 */ NdrFcShort(0xffff	/* Offset= 6 (496) */ /* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RF ffdc), /* Offset=	/* FC_ALIGNM8 */ /* FC_END */ 0 */ = -36 (462) */	/* 576 */ FC_CON /* 578 */ /* 582 */ /* 584 */ /* 586 */	STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0xc0,	0x2f, 0x5a, /* 47 */ /* 0 */ /* 192 */ 0x0,	-36 (538) */ /* FC_IP */ /*
/* 490 */ NdrFcShort(0x6), /* 492 */ 0x8, /* 494 */ 0x36, /* 496 */ /* 498 */ NdrFcShort(0xffff /* 500 */ FC_BOGUS_ARRAY */	/* Offset= 6 (496) */ /* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RF ffdc), /* Offset=	/* FC_ALIGNM8 */ /* FC_END */ 0 */ = -36 (462) */ /*	/* 576 */ FC_CON /* 578 */ /* 582 */ /* 584 */	STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0xc0,	0x2f, 0x5a, /* 47 */ /* 0 */ /* 192 */ 0x0, /* 0 */ 0x0,	-36 (538) */ /* FC_IP */ /* /* 0 */
/* 490 */ NdrFcShort(0x6), /* 492 */ 0x8, /* 494 */ 0x36, /* 496 */ /* 498 */ NdrFcShort(0xffff /* 500 */	/* Offset= 6 (496) */ /* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RF ffdc), /* Offset= 0x21, 0x3, /* 0 */	/* FC_ALIGNM8 */ /* FC_END */ 0 */ = -36 (462) */ /*	/* 576 */ FC_CON /* 578 */ /* 582 */ /* 584 */ /* 586 */	STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0xc0, 0x0,	0x2f, 0x5a, /* 47 */ /* 0 */ /* 192 */ 0x0, /* 0 */ /* 0 */	-36 (538) */ /* FC_IP */ /* /* 0 */
/* 490 */ NdrFcShort(0x6), /* 492 */ 0x8, /* 494 */ 0x36, /* 496 */ /* 500 */ FC_BOGUS_ARRAY */ /* 502 */ NdrFcShort(0x0),	/* Offset= 6 (496) */ /* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RF ffdc), /* Offset= 0x21, 0x3, /* 0 */	/* FC_ALIGNM8 */ /* FC_END */ = -36 (462) */ /* /* 3 */	/* 576 */ FC_CON /* 578 */ /* 582 */ /* 584 */ /* 586 */ /* 589 */	STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0xc0, 0x0,	0x2f, 0x2f, 0x5a, /* 47 */ /* 0 */ /* 0 */ 0x0, /* 0 */ 0x0, /* 0 */	-36 (538) */ /* FC_IP */ /* /* 0 */
/* 490 */ NdrFcShort(0x6), /* 492 */ 0x8, /* 494 */ 0x36, /* 496 */ /* 500 */ FC_BOGUS_ARRAY */ /* 502 */ NdrFcShort(0x0),	/* Offset= 6 (496) */ /* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RF ffdc), /* Offset= 0x21, 0x3, /* 0 */ /* Corr desc: field p	/* FC_ALIGNM8 */ /* FC_END */ D */ = -36 (462) */ /* 3 */ cointer, FC_ULONG */	/* 576 */ FC_CON /* 578 */ /* 582 */ /* 584 */ /* 586 */ /* 589 */	STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0xc0, 0x0,	0x2f, 0x2f, 0x5a, /* 47 */ /* 0 */ /* 0 */ 0x0, /* 0 */ 0x0, /* 0 */ 0x0, /* 0 */	-36 (538) */ /* FC_IP */ /* /* 0 */ /* 0 */
/* 490 */ NdrFcShort(0x6), /* 492 */ 0x8, /* 494 */ 0x36, /* 496 */ /* 500 */ FC_BOGUS_ARRAY */ /* 502 */ NdrFcShort(0x0), /* 504 */ 0x19, /* 506 */ NdrFcShort(0x0), /* 508 */ NdrFcShort(0x1),	/* Offset= 6 (496) */ /* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RF ffdc), /* Offset= 0x21, 0x3, /* 0 */ /* Corr desc: field p 0x0, /* 0 */ /* Corr flags: early,	/* FC_ALIGNM8 */ /* FC_END */ 0 */ = -36 (462) */ /* /* 3 */ cointer, FC_ULONG */ /* */	/* 576 */ FC_CON /* 578 */ /* 582 */ /* 584 */ /* 588 */ /* 590 */ /* 592 */	STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0xc0, 0x0,	0x2f, 0x2f, 0x5a, /* 47 */ /* 0 */ /* 0 */ 0x0, /* 0 */ 0x0, /* 0 */ 0x0, /* 0 */	-36 (538) */ /* FC_IP */ /* /* 0 */ /* 0 */
/* 490 */ NdrFcShort(0x6), /* 492 */ 0x8, /* 494 */ 0x36, /* 496 */ /* 498 */ NdrFcShort(0xfffff /* 500 */ FC_BOGUS_ARRAY */ /* 502 */ NdrFcShort(0x0), /* 504 */ 0x19, /* 506 */ NdrFcShort(0x0),	/* Offset= 6 (496) */ /* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RF ffdc), /* Offset= 0x21, 0x3, /* 0 */ /* Corr desc: field p 0x0, /* 0 */ /* Corr flags: early, ffff), /* -1 */	/* FC_ALIGNM8 */ /* FC_END */ 0 */ = -36 (462) */ /* /* 3 */ cointer, FC_ULONG */ /* */	/* 576 */ FC_CON /* 578 */ /* 582 */ /* 584 */ /* 588 */ /* 590 */ /* 592 */	STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0xc0, 0x0,	lc), /* Offset= 0x2f, 0x5a, /* 47 */	-36 (538) */ /* FC_IP */ /* /* 0 */ /* 0 */ /* 0 */ /* 70 */
/* 490 */ NdrFcShort(0x6), /* 492 */ 0x8, /* 494 */ 0x36, /* 496 */ /* 500 */ FC_BOGUS_ARRAY */ /* 502 */ NdrFcShort(0x0), /* 504 */ 0x19, /* 508 */ NdrFcShort(0x0), /* 508 */ NdrFcShort(0x1), /* 510 */ NdrFcLong(0xfffff /* 514 */ NdrFcShort(0x0),	/* Offset= 6 (496) */ /* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RF ffdc), /* Offset= 0x21, 0x3, /* 0 */ /* Corr desc: field p 0x0, /* 0 */ /* Corr flags: early, /* Corr flags: */	/* FC_ALIGNM8 */ /* FC_END */ 0 */ = -36 (462) */ /* /* 3 */ cointer, FC_ULONG */ /* */	/* 576 */ FC_CON /* 578 */ /* 582 */ /* 584 */ /* 588 */ /* 590 */ /* 592 */ /* 594 */	STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0xc0, 0x0, 0x0, 0x0, NdrFcShort(0x1),	0x2f, 0x2f, 0x5a, /* 47 */ /* 0 */ /* 192 */ 0x0, /* 0 */ 0x0, /* 0 */ 0x0, /* 0 */ 0x0, /* 0 */ 0x0, /* 1 */ 0x0, 0x0, /* 1 */ 0x0, 0x0, /* 1 */ 0x0,	-36 (538) */ /* FC_IP */ /* 0 */ /* 0 */ /* 0 */ /* 70 */ /* FC_CARRAY */ /* 0 */
/* 490 */ NdrFcShort(0x6), /* 492 */ 0x8, /* 494 */ 0x36, /* 496 */ /* 498 */ NdrFcShort(0xfffff /* 500 */ FC_BOGUS_ARRAY */ /* 502 */ NdrFcShort(0x0), /* 504 */ 0x19, /* 506 */ NdrFcShort(0x0), /* 508 */ NdrFcShort(0x1), /* 510 */ NdrFcLong(0xfffff	/* Offset= 6 (496) */ /* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RF ffdc), /* Offset= 0x21, 0x3, /* 0 */ /* Corr desc: field p 0x0, /* 0 */ /* Corr flags: early, /* Corr flags: */ /* FC_EMBEDDED	/* FC_ALIGNM8 */ /* FC_END */ D */ = -36 (462) */ /* /* 3 */ cointer, FC_ULONG */ /* */ */ COMPLEX */	/* 576 */ FC_CON /* 578 */ /* 582 */ /* 584 */ /* 588 */ /* 590 */ /* 592 */ /* 594 */	STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0xc0, 0x0, 0x0, 0x0, NdrFcShort(0x1),	0x2f, 0x2f, 0x5a, /* 47 */ /* 0 */ /* 192 */ 0x0, /* 0 */ 0x0, /* 0 */ 0x0, /* 0 */ 0x0, /* 1 */ 0x0, /* 1 */ 0x0, /* 2 */ 0x0, /* 3 */ 0x0, /* 5 */ 0x1b, 0x2, 0x3, 0x4, 0x4, 0x4, 0x4, 0x4, 0x5, 0x7, 0x7, 0x7, 0x8, 0x8, 0x9, 0x	-36 (538) */ /* FC_IP */ /* 0 */ /* 0 */ /* 0 */ /* 70 */ /* 70 */ /* FC_CARRAY */ /* 0 */ sinter, FC_ULONG */
/* 490 */ NdrFcShort(0x6), /* 492 */ 0x8, /* 494 */ 0x36, /* 496 */ /* 498 */ NdrFcShort(0xfffff /* 500 */ FC_BOGUS_ARRAY */ /* 502 */ NdrFcShort(0x0), /* 504 */ 0x19, /* 508 */ NdrFcShort(0x1), /* 510 */ NdrFcShort(0x1), /* 510 */ NdrFcShort(0x0), /* 514 */ NdrFcShort(0x0), /* 516 */ Ox4c,	/* Offset= 6 (496) */ /* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RF ffdc), /* Offset= 0x21, 0x3, /* 0 */ /* Corr desc: field p 0x0, /* 0 */ /* Corr flags: early, /* Corr flags: */ /* FC_EMBEDDED 0x0,	/* FC_ALIGNM8 */ /* FC_END */ 0 */ = -36 (462) */ /* /* 3 */ ointer, FC_ULONG */ /* */ */ COMPLEX */ /* 0 */	/* 576 */ FC_CON /* 578 */ /* 582 */ /* 584 */ /* 588 */ /* 590 */ /* 592 */ /* 594 */ /* 598 */	STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0xc0, 0x0, 0x0, 0x0, NdrFcShort(0x1), 0x19,	0x2f, 0x2f, 0x5a, /* 47 */ /* 0 */ /* 0 */ /* 192 */ 0x0, /* 0 */ 0x0, /* 0 */ 0x0, /* 0 */ 0x0, /* 1 */ 0x0, /* 0 */ 0x0, /* 1 */ /* Corr desc: field pc 0x0,	-36 (538) */ /* FC_IP */ /* 0 */ /* 0 */ /* 0 */ /* 70 */ /* FC_CARRAY */ /* 0 */
/* 490 */ NdrFcShort(0x6), /* 492 */ 0x8, /* 494 */ 0x36, /* 496 */ /* 498 */ NdrFcShort(0xfffff /* 500 */ FC_BOGUS_ARRAY */ /* 502 */ NdrFcShort(0x0), /* 504 */ 0x19, /* 508 */ NdrFcShort(0x0), /* 508 */ NdrFcShort(0x1), /* 510 */ NdrFcLong(0xfffff /* 514 */ NdrFcShort(0x0), /* 516 */ NdrFcShort(0x0), /* 516 */ NdrFcShort(0x0), /* 518 */ NdrFcShort(0x0),	/* Offset= 6 (496) */ /* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RF ffdc), /* Offset= 0x21, 0x3, /* 0 */ /* Corr desc: field p 0x0, /* 0 */ /* Corr flags: early, /* Corr flags: */ /* FC_EMBEDDED 0x0, ff44), /* Offset=	/* FC_ALIGNM8 */ /* FC_END */ D */ = -36 (462) */ /* /* 3 */ cointer, FC_ULONG */ /* */ */ COMPLEX */	/* 576 */ FC_CON /* 578 */ /* 582 */ /* 584 */ /* 588 */ /* 590 */ /* 592 */ /* 594 */ /* 598 */ /* 600 */	STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0xc0, 0x0, 0x0, 0x0, NdrFcShort(0x1), 0x19, NdrFcShort(0x4),	0x2f, 0x2f, 0x5a, /* 47 */ /* 0 */ /* 0 */ /* 192 */ 0x0, /* 0 */ 0x0, /* 0 */ 0x0, /* 0 */ 0x0, /* 1 */ 0x46, 0x1b, 0x0, /* 1 */ /* Corr desc: field po	-36 (538) */ /* FC_IP */ /* /* 0 */ /* 0 */ /* 0 */ /* 70 */ /* FC_CARRAY */ /* 0 */ inter, FC_ULONG */ /* */
/* 490 */ NdrFcShort(0x6), /* 492 */ 0x8, /* 494 */ 0x36, /* 496 */ /* 498 */ NdrFcShort(0xfffff /* 500 */ FC_BOGUS_ARRAY */ /* 502 */ NdrFcShort(0x0), /* 504 */ 0x19, /* 508 */ NdrFcShort(0x1), /* 510 */ NdrFcShort(0x1), /* 510 */ NdrFcShort(0x0), /* 514 */ NdrFcShort(0x0), /* 516 */ Ox4c,	/* Offset= 6 (496) */ /* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RF ffdc), /* Offset= 0x21, 0x3, /* 0 */ /* Corr desc: field p 0x0, /* 0 */ /* Corr flags: early, /* Corr flags: */ /* FC_EMBEDDED 0x0, ff44), /* Offset= /* FC_PAD */	/* FC_ALIGNM8 */ /* FC_END */ 0 */ = -36 (462) */ /* /* 3 */ ointer, FC_ULONG */ /* */ */ COMPLEX */	/* 576 */ FC_CON /* 578 */ /* 582 */ /* 584 */ /* 580 */ /* 590 */ /* 594 */ /* 598 */ /* 600 */ /* 602 */	STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0xc0, 0x0, 0x0, 0x0, NdrFcShort(0x1), 0x19, NdrFcShort(0x4), NdrFcShort(0x1),	0x2f, 0x2f, 0x5a, /* 47 */ /* 0 */ /* 0 */ /* 192 */ 0x0, /* 0 */ 0x0, /* 1 */ /* Corr desc: field pc 0x0, /* 4 */ /* Corr flags: early, *	-36 (538) */ /* FC_IP */ /* /* 0 */ /* 0 */ /* 0 */ /* 70 */ /* FC_CARRAY */ /* 0 */ inter, FC_ULONG */ /* */
/* 490 */ NdrFcShort(0x6), /* 492 */ 0x8, /* 494 */ 0x36, /* 496 */ /* 498 */ NdrFcShort(0xfffff /* 500 */ FC_BOGUS_ARRAY */ /* 502 */ NdrFcShort(0x0), /* 504 */ 0x19, /* 508 */ NdrFcShort(0x1), /* 510 */ NdrFcShort(0x1), /* 510 */ NdrFcShort(0x0), /* 514 */ NdrFcShort(0x0), /* 514 */ NdrFcShort(0x0), /* 518 */ NdrFcShort(0x0), /* 518 */ NdrFcShort(0xfffff /* 520 */ 0x5c,	/* Offset= 6 (496) */ /* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RF ffdc), /* Offset= 0x21, 0x3, /* 0 */ /* Corr desc: field p 0x0, /* 0 */ /* Corr flags: early, /* Corr flags: */ /* FC_EMBEDDED 0x0, ff44), /* Offset=	/* FC_ALIGNM8 */ /* FC_END */ D */ = -36 (462) */ /* /* 3 */ cointer, FC_ULONG */ /* */ */ COMPLEX */ /* 0 */	/* 576 */ FC_CON /* 578 */ /* 582 */ /* 584 */ /* 588 */ /* 590 */ /* 592 */ /* 594 */ /* 598 */ /* 600 */	STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0xc0, 0x0, 0x0, 0x0, NdrFcShort(0x1), 0x19, NdrFcShort(0x4), NdrFcShort(0x1),	0x2f, 0x2f, 0x5a, /* 47 */ /* 0 */ /* 0 */ /* 192 */ 0x0, /* 0 */ 0x0, /* 1 */ /* Corr desc: field pc 0x0, /* 4 */ /* Corr flags: early, */ /* FC_BYTE */	-36 (538) */ /* FC_IP */ /* 0 */ /* 0 */ /* 0 */ /* 70 */ /* FC_CARRAY */ /* 0 */ sinter, FC_ULONG */ /* */
/* 490 */ NdrFcShort(0x6), /* 492 */ 0x8, /* 494 */ 0x36, /* 496 */ /* 498 */ NdrFcShort(0xfffff /* 500 */ FC_BOGUS_ARRAY */ /* 502 */ NdrFcShort(0x0), /* 504 */ 0x19, /* 508 */ NdrFcShort(0x0), /* 508 */ NdrFcShort(0x1), /* 510 */ NdrFcLong(0xfffff /* 514 */ NdrFcShort(0x0), /* 516 */ NdrFcShort(0x0), /* 516 */ NdrFcShort(0x0), /* 518 */ NdrFcShort(0x0),	/* Offset= 6 (496) */ /* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RF ffdc), /* Offset= 0x21, 0x3, /* 0 */ /* Corr desc: field p 0x0, /* 0 */ /* Corr flags: early, /* Corr flags: */ /* FC_EMBEDDED 0x0, ff44), /* Offset= /* FC_PAD */ 0x5b,	/* FC_ALIGNM8 */ /* FC_END */ -36 (462) */ /* /* 3 */ cointer, FC_ULONG */ /* */ */ COMPLEX */	/* 576 */ FC_CON /* 578 */ /* 582 */ /* 584 */ /* 588 */ /* 590 */ /* 594 */ /* 596 */ /* 598 */ /* 600 */ /* 602 */ /* 604 */	STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0xc0, 0x0, 0x0, 0x0, NdrFcShort(0x1), 0x19, NdrFcShort(0x4), NdrFcShort(0x1),	0x2f, 0x2f, 0x5a, /* 47 */ /* 0 */ /* 0 */ /* 192 */ 0x0, /* 0 */ 0x0, /* 1 */ /* Corr desc: field pc 0x0, /* 4 */ /* Corr flags: early, *	-36 (538) */ /* FC_IP */ /* /* 0 */ /* 0 */ /* 0 */ /* 70 */ /* FC_CARRAY */ /* 0 */ inter, FC_ULONG */ /* */
/* 490 */ NdrFcShort(0x6), /* 492 */ 0x8, /* 494 */ 0x36, /* 496 */ /* 498 */ NdrFcShort(0xfffff /* 500 */ FC_BOGUS_ARRAY */ /* 502 */ NdrFcShort(0x0), /* 504 */ 0x19, /* 508 */ NdrFcShort(0x1), /* 510 */ NdrFcShort(0x0), /* 514 */ NdrFcShort(0x0), /* 514 */ NdrFcShort(0x0), /* 514 */ NdrFcShort(0x0), /* 515 */ NdrFcShort(0x0), /* 516 */ 0x4c,	/* Offset= 6 (496) */ /* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RF ffdc), /* Offset= 0x21, 0x3, /* 0 */ /* Corr desc: field p 0x0, /* 0 */ /* Corr flags: early, /* Corr flags: */ /* FC_EMBEDDED 0x0, ff44), /* Offset= /* FC_PAD */	/* FC_ALIGNM8 */ /* FC_END */ 0 */ = -36 (462) */ /* /* 3 */ ointer, FC_ULONG */ /* */ */ COMPLEX */	/* 576 */ FC_CON /* 578 */ /* 582 */ /* 584 */ /* 580 */ /* 590 */ /* 594 */ /* 598 */ /* 600 */ /* 602 */	STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0xc0, 0x0, 0x0, 0x0, NdrFcShort(0x1), 0x19, NdrFcShort(0x4), NdrFcShort(0x1),	0x2f, 0x2f, 0x5a, /* 47 */ /* 0 */ /* 0 */ /* 192 */ 0x0, /* 0 */ 0x0, /* 1 */ /* Corr desc: field po 0x0, /* 4 */ /* Corr flags: early, */ /* FC_BYTE */ 0x5b,	-36 (538) */ /* FC_IP */ /* O */ /* 0 */ /* 0 */ /* 70 */ /* FC_CARRAY */ /* 0 */ sinter, FC_ULONG */ /* */ /* FC_END */
/* 490 */ NdrFcShort(0x6), /* 492 */ 0x8, /* 494 */ 0x36, /* 496 */ /* 498 */ NdrFcShort(0xfffff /* 500 */ FC_BOGUS_ARRAY */ /* 502 */ NdrFcShort(0x0), /* 504 */ 0x19, /* 508 */ NdrFcShort(0x1), /* 510 */ NdrFcShort(0x1), /* 510 */ NdrFcShort(0x0), /* 514 */ NdrFcShort(0x0), /* 514 */ NdrFcShort(0x0), /* 518 */ NdrFcShort(0x0), /* 518 */ NdrFcShort(0xfffff /* 520 */ 0x5c,	/* Offset= 6 (496) */ /* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RF ffdc), /* Offset= 0x21, 0x3, /* 0 */ /* Corr desc: field p 0x0, /* 0 */ /* Corr flags: early, /* Corr flags: */ /* FC_EMBEDDED 0x0, ff44), /* Offset= /* FC_PAD */ 0x5b, 0x1a,	/* FC_ALIGNM8 */ /* FC_END */ 0 */ = -36 (462) */ /* /* 3 */ cointer, FC_ULONG */ /* */ */ COMPLEX */	/* 576 */ FC_CON /* 578 */ /* 582 */ /* 584 */ /* 588 */ /* 590 */ /* 592 */ /* 594 */ /* 596 */ /* 598 */ /* 600 */ /* 602 */ /* 604 */	STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0x0, 0x0, 0x0, NdrFcShort(0x1), 0x19, NdrFcShort(0x4), NdrFcShort(0x1), 0x1,	0x2f, 0x2f, 0x5a, /* 47 */ /* 0 */ /* 0 */ /* 192 */ 0x0, /* 0 */ 0x0, /* 1 */ /* Corr desc: field pc 0x0, /* 4 */ /* Corr flags: early, */ /* FC_BYTE */	-36 (538) */ /* FC_IP */ /* 0 */ /* 0 */ /* 0 */ /* 70 */ /* FC_CARRAY */ /* 0 */ sinter, FC_ULONG */ /* */
/* 490 */ NdrFcShort(0x6), /* 492 */ 0x8, /* 494 */ 0x36, /* 496 */ /* 498 */ NdrFcShort(0xfffff /* 500 */ FC_BOGUS_ARRAY */ /* 502 */ NdrFcShort(0x0), /* 504 */ 0x19, /* 508 */ NdrFcShort(0x1), /* 510 */ NdrFcShort(0x0), /* 514 */ NdrFcShort(0x0), /* 514 */ NdrFcShort(0x0), /* 518 */ NdrFcShort(0x0), /* 518 */ NdrFcShort(0x0), /* 522 */ FC_BOGUS_STRUCT */	/* Offset= 6 (496) */ /* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RF ffdc), /* Offset= 0x21, 0x3, /* 0 */ /* Corr desc: field p 0x0, /* 0 */ /* Corr flags: early, /fff), /* -1 */ /* FC_EMBEDDED 0x0, ff44), /* Offset= /* FC_PAD */ 0x5b, 0x1a, 0x3,	/* FC_ALIGNM8 */ /* FC_END */ -36 (462) */ /* /* 3 */ cointer, FC_ULONG */ /* */ */ COMPLEX */	/* 576 */ FC_CON /* 578 */ /* 582 */ /* 584 */ /* 588 */ /* 590 */ /* 592 */ /* 594 */ /* 596 */ /* 598 */ /* 600 */ /* 602 */ /* 604 */	STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0xc0, 0x0, 0x0, 0x0, NdrFcShort(0x1), 0x19, NdrFcShort(0x4), NdrFcShort(0x1),	lc), /* Offset= 0x2f, 0x5a, /* 47 */ /* 0 */ /* 0 */ /* 192 */ 0x0, /* 0 */ 0x0, /* 0 */ 0x46, 0x1b, 0x0, /* 1 */ /* Corr desc: field po 0x0, /* 4 */ /* FC_BYTE */ 0x1a,	-36 (538) */ /* FC_IP */ /* O */ /* 0 */ /* 0 */ /* 0 */ /* 70 */ /* FC_CARRAY */ /* 0 */ sinter, FC_ULONG */ /* */ /* FC_END */ /*
/* 490 */ NdrFcShort(0x6), /* 492 */ 0x8, /* 494 */ 0x36, /* 496 */ /* 498 */ NdrFcShort(0xfffff /* 500 */ FC_BOGUS_ARRAY */ /* 502 */ NdrFcShort(0x0), /* 504 */ 0x19, /* 508 */ NdrFcShort(0x1), /* 510 */ NdrFcShort(0x0), /* 514 */ NdrFcShort(0x0), /* 514 */ NdrFcShort(0x0), /* 518 */ NdrFcShort(0x0), /* 518 */ NdrFcShort(0x0), /* 520 */ 0x5c, /* 522 */ FC_BOGUS_STRUCT */ /* 524 */ NdrFcShort(0x10)	/* Offset= 6 (496) */ /* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RF ffdc), /* Offset= 0x21, 0x3, /* 0 */ /* Corr desc: field p 0x0, /* 0 */ /* Corr flags: early, ffff), /* -1 */ /* FC_EMBEDDED 0x0, ff44), /* Offset= /* FC_PAD */ 0x5b, 0x1a, 0x3, /* 16 */	/* FC_ALIGNM8 */ /* FC_END */ 0 */ = -36 (462) */ /* /* 3 */ cointer, FC_ULONG */ /* */ */ COMPLEX */	/* 576 */ FC_CON /* 578 */ /* 582 */ /* 584 */ /* 588 */ /* 590 */ /* 592 */ /* 594 */ /* 596 */ /* 598 */ /* 600 */ /* 600 */ FC_BOG	STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0x0, 0x0, 0x0, NdrFcShort(0x1), 0x19, NdrFcShort(0x4), NdrFcShort(0x1), 0x1, US_STRUCT */	lc), /* Offset= 0x2f, 0x5a, /* 47 */	-36 (538) */ /* FC_IP */ /* O */ /* 0 */ /* 0 */ /* 70 */ /* FC_CARRAY */ /* 0 */ sinter, FC_ULONG */ /* */ /* FC_END */
/* 490 */ NdrFcShort(0x6), /* 492 */ 0x8, /* 494 */ 0x36, /* 496 */ /* 498 */ NdrFcShort(0xfffff /* 500 */ FC_BOGUS_ARRAY */ /* 502 */ NdrFcShort(0x0), /* 504 */ 0x19, /* 508 */ NdrFcShort(0x1), /* 510 */ NdrFcShort(0x0), /* 514 */ NdrFcShort(0x0), /* 514 */ NdrFcShort(0x0), /* 518 */ NdrFcShort(0x0), /* 518 */ NdrFcShort(0x0), /* 520 */ 0x5c, /* 522 */ FC_BOGUS_STRUCT */ /* 524 */ NdrFcShort(0x10), /* 526 */ NdrFcShort(0x0),	/* Offset= 6 (496) */ /* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RF ffdc), /* Offset= 0x21, 0x3, /* 0 */ /* Corr desc: field p 0x0, /* 0 */ /* Corr flags: early, /* FC_EMBEDDED 0x0, fff44), /* Offset= /* FC_PAD */ 0x5b, 0x1a, 0x3, /* 16 */ /* 0 */	/* FC_ALIGNM8 */ /* FC_END */ D */ = -36 (462) */ /* /* 3 */ cointer, FC_ULONG */ /* */ */ COMPLEX */ /* 0 */ = -188 (330) */ /* FC_END */ /* 3 */	/* 576 */ FC_CON /* 578 */ /* 582 */ /* 584 */ /* 588 */ /* 590 */ /* 592 */ /* 594 */ /* 596 */ /* 598 */ /* 600 */ /* 600 */ FC_BOG /* 608 */	STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0x0, 0x0, 0x0, NdrFcShort(0x1), 0x19, NdrFcShort(0x4), NdrFcShort(0x1), 0x1, US_STRUCT */ NdrFcShort(0x18),	lc), /* Offset= 0x2f, 0x5a, /* 47 */ /* 0 */ /* 0 */ /* 192 */ 0x0, /* 0 */ 0x46, 0x1b, 0x0, /* 1 */ /* Corr desc: field po 0x0, /* 4 */ /* FC_BYTE */ 0x3, /* 24 */	-36 (538) */ /* FC_IP */ /* O */ /* 0 */ /* 0 */ /* 0 */ /* 70 */ /* FC_CARRAY */ /* 0 */ sinter, FC_ULONG */ /* */ /* FC_END */ /*
/* 490 */ NdrFcShort(0x6), /* 492 */ 0x8, /* 494 */ 0x36, /* 496 */ /* 498 */ NdrFcShort(0xfffff /* 500 */ FC_BOGUS_ARRAY */ /* 502 */ NdrFcShort(0x0), /* 504 */ 0x19, /* 508 */ NdrFcShort(0x1), /* 510 */ NdrFcShort(0x0), /* 514 */ NdrFcShort(0x0), /* 514 */ NdrFcShort(0x0), /* 518 */ NdrFcShort(0x0), /* 518 */ NdrFcShort(0x0), /* 520 */ 0x5c, /* 522 */ FC_BOGUS_STRUCT */ /* 524 */ NdrFcShort(0x10)	/* Offset= 6 (496) */ /* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RF ffdc), /* Offset= 0x21, 0x3, /* 0 */ /* Corr desc: field p 0x0, /* 0 */ /* Corr flags: early, ffff), /* -1 */ /* FC_EMBEDDED 0x0, ff44), /* Offset= /* FC_PAD */ 0x5b, 0x1a, 0x3, /* 16 */	/* FC_ALIGNM8 */ /* FC_END */ D */ = -36 (462) */ /* /* 3 */ cointer, FC_ULONG */ /* */ */ COMPLEX */ /* 0 */ = -188 (330) */ /* FC_END */ /* 3 */	/* 576 */ FC_CON /* 578 */ /* 582 */ /* 584 */ /* 588 */ /* 590 */ /* 592 */ /* 594 */ /* 596 */ /* 600 */ /* 602 */ /* 604 */ FC_BOG /* 608 */ /* 610 */	STANT_IID */ NdrFcLong(0x2f), NdrFcShort(0x0), NdrFcShort(0x0), 0x0, 0x0, 0x0, NdrFcShort(0x1), 0x19, NdrFcShort(0x4), NdrFcShort(0x1), 0x1, US_STRUCT */	lc), /* Offset= 0x2f, 0x5a, /* 47 */	-36 (538) */ /* FC_IP */ /* O */ /* 0 */ /* 0 */ /* 70 */ /* FC_CARRAY */ /* 0 */ inter, FC_ULONG */ /* */ /* FC_END */ /* 3 */

/ * 614 * /	0x8.	/* FC LONG */		/* 692 */	0x8.	/* FC_LONG */	
	,	0x8,	/* FC_LONG */		,	0x39,	/* FC_ALIGNM8 */
/ * 616 * /	0x4c	/* FC EMBEDDED		/* 694 */	0v36	/* FC_POINTER */	, re_religionio
/ 010 /	oate,	0x0,	_* 0 */	/ 0)4 /	0.750,		/*
/ * (10 */	NIECI (O CCCCC	*		EC EMB	EDDED COMBLEY	0x4c,	/ ·
	NdrFcShort(0xffffff				EDDED_COMPLEX *		
/* 620 */	0x39,	/* FC_ALIGNM8 */		/* 696 */	0x0,	/* 0 */	
		0x36,	/* FC_POINTER */			NdrFcShort(0xffffffe	e7), /* Offset=
/* 622 */	0x5c,	/* FC_PAD */		-25 (672)	*/		
		0x5b,	/* FC_END */			0x5b,	/* FC_END */
/ * 624 * /				/* 700 * /			
		0x12, 0x0, /* FC_UP	*/			0x11, 0x0, /* FC_RP	*/
/* 626 */	NdrFcShort(0xffffff6	e0), /* Offset=		/* 702 */	NdrFcShort(0xfffffff		-240 (462) */
/* 628 */		,, , , , , , , , , , , , , , , , , , , ,	= (=, 1)	/* 704 */		, , , , ,	(, ,
, 020 ,		0x21,	/ *	, , , , ,		0x1b,	/* FC CARRAY */
EC DOC	IIC ADDAX/*/	0.721,	/			0x10, $0x0$,	/* 0 */
rc_bog	US_ARRAY */	0.2	(* 2 * l	/* 70C */	NIECL ((O.1)		/ · 0 ·/
(* C20 */	NIE CL ((0.0)	0x3,	/* 3 */		NdrFcShort(0x1),	/* 1 */	· · FG III ONG */
	NdrFcShort(0x0),	/* 0 */		/* 708 */	0x19,		ointer, FC_ULONG */
/ * 632 * /	0x19,		ointer, FC_ULONG */			0x0,	/* */
		0x0,	/* */		NdrFcShort(0x0),	/* 0 */	
	NdrFcShort(0x0),	/* 0 */			NdrFcShort(0x1),	/* Corr flags: early, *	*/
/ * 636 * /	NdrFcShort(0x1),	/* Corr flags: early, '	*/	/* 714 */	0x1,	/* FC_BYTE */	
/ * 638 * /	NdrFcLong(0xfffffff	f), /* -1 */				0x5b,	/* FC_END */
/* 642 */	NdrFcShort(0x0),	/* Corr flags: */		/* 716 */			_
/* 644 */	· //	J				0x1a,	/*
,,		0x12, 0x0, /* FC_UP	*/	FC BOG	US STRUCT */	vu,	•
/ * 646 * /	NdrFcShort(0xffffff6	18) /* Officet=	-40 (606) */	TC_BOG	os_stroct /	0x3,	/* 3 */
/* 648 */			-40 (000) /	/ * 710 * /	NdrFcShort(0x10),		/ 3 /
/ 048 /	OXSC,	/* FC_PAD */	/* FG END */			/* 0 */	
(d) 6 7 0 d) (0x5b,	/* FC_END */		NdrFcShort(0x0),		
/ * 650 * /					NdrFcShort(0x6),	/* Offset= 6 (728) */	
		0x1a,	/*	/* 724 */	0x8,	/* FC_LONG */	
FC_BOG	US_STRUCT */					0x39,	/* FC_ALIGNM8 */
		0x3,	/* 3 */	/* 726 */	0x36,	/* FC_POINTER */	
/* 652 */	NdrFcShort(0x10),	/* 16 */				0x5b,	/* FC END */
	NdrFcShort(0x0),	/* 0 */		/* 728 */			_
	NdrFcShort(0x6),	/* Offset= 6 (662) */				0x12, 0x0, /* FC UP	*/
				/* 730 */	NdrFcShort(0xffffffe		
/* 658 */		/* FC_LONG */			NdrFcShort(0xffffffe		
/* 658 */	0x8,	/* FC_LONG */ 0x39,	/* FC_ALIGNM8 */	/* 730 */ /* 732 */	NdrFcShort(0xffffff6	e6), /* Offset=	: -26 (704) */
	0x8,	/* FC_LONG */ 0x39, /* FC_POINTER */	/* FC_ALIGNM8 */		NdrFcShort(0xffffff6	e6), /* Offset= 0x1b,	-26 (704) */ /* FC_CARRAY */
/* 658 */ /* 660 */	0x8,	/* FC_LONG */ 0x39,		/* 732 */		0x1b, 0x1,	: -26 (704) */
/* 658 */	0x8,	/* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b,	/* FC_ALIGNM8 */ /* FC_END */	/* 732 */ /* 734 */	NdrFcShort(0x2),	0x1b, 0x1, /* 2 */	-26 (704) */ /* FC_CARRAY */ /* 1 */
/* 658 */ /* 660 */ /* 662 */	0x8, 0x36,	/* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RP	/* FC_ALIGNM8 */ /* FC_END */ */	/* 732 */	NdrFcShort(0x2),	0x1b, 0x1, /* 2 */ /* Corr desc: field po	-26 (704) */ /* FC_CARRAY */ /* 1 */ pinter, FC_ULONG */
/* 658 */ /* 660 */ /* 662 */ /* 664 */	0x8,	/* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RP	/* FC_ALIGNM8 */ /* FC_END */	/* 732 */ /* 734 */ /* 736 */	NdrFcShort(0x2), 0x19,	0x1b, 0x1, 0x1, /* 2 */ /* Corr desc: field po 0x0,	-26 (704) */ /* FC_CARRAY */ /* 1 */
/* 658 */ /* 660 */ /* 662 */	0x8, 0x36,	/* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RP	/* FC_ALIGNM8 */ /* FC_END */ */	/* 732 */ /* 734 */ /* 736 */	NdrFcShort(0x2),	0x1b, 0x1, /* 2 */ /* Corr desc: field po	-26 (704) */ /* FC_CARRAY */ /* 1 */ pinter, FC_ULONG */
/* 658 */ /* 660 */ /* 662 */ /* 664 */	0x8, 0x36,	/* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RP	/* FC_ALIGNM8 */ /* FC_END */ */	/* 732 */ /* 734 */ /* 736 */ /* 738 */	NdrFcShort(0x2), 0x19,	0x1b, 0x1, /* 2 */ /* Corr desc: field pc 0x0, /* 0 */ /* Corr flags: early, *	-26 (704) */ /* FC_CARRAY */ /* 1 */ sinter, FC_ULONG */ /* */
/* 658 */ /* 660 */ /* 662 */ /* 664 */	0x8, 0x36,	/* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RP dc), /* Offset=	/* FC_ALIGNM8 */ /* FC_END */ */ :-36 (628) */	/* 732 */ /* 734 */ /* 736 */ /* 738 */	NdrFcShort(0x2), 0x19, NdrFcShort(0x0), NdrFcShort(0x1),	0x1b, 0x1, /* 2*/ /* Corr desc: field po 0x0, /* 0*/	-26 (704) */ /* FC_CARRAY */ /* 1 */ sinter, FC_ULONG */ /* */
/* 658 */ /* 660 */ /* 662 */ /* 664 */ /* 666 */	0x8, 0x36, NdrFcShort(0xfffffff	/* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RP dc), /* Offset= 0x1d,	/* FC_ALIGNM8 */ /* FC_END */ */ :-36 (628) */ /* FC_SMFARRAY */	/* 732 */ /* 734 */ /* 736 */ /* 738 */ /* 740 */	NdrFcShort(0x2), 0x19, NdrFcShort(0x0), NdrFcShort(0x1),	0x1b, 0x1, /* 2 */ /* Corr desc: field pc 0x0, /* 0 */ /* Corr flags: early, *	-26 (704) */ /* FC_CARRAY */ /* 1 */ sinter, FC_ULONG */ /* */
/* 658 */ /* 660 */ /* 662 */ /* 664 */ /* 666 */	0x8, 0x36, NdrFcShort(0xfffffff) NdrFcShort(0x8),	/* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RP dc), /* Offset= 0x1d, 0x0,	/* FC_ALIGNM8 */ /* FC_END */ */ :-36 (628) */ /* FC_SMFARRAY */	/* 732 */ /* 734 */ /* 736 */ /* 738 */ /* 740 */ /* 742 */	NdrFcShort(0x2), 0x19, NdrFcShort(0x0), NdrFcShort(0x1),	0x1b, 0x1, /* 2 */ /* Corr desc: field po 0x0, /* 0 */ /* Corr flags: early, */ /* FC_SHORT */	-26 (704) */ /* FC_CARRAY */ /* 1 */ sinter, FC_ULONG */ /* */
/* 658 */ /* 660 */ /* 662 */ /* 664 */ /* 666 */	0x8, 0x36, NdrFcShort(0xfffffff) NdrFcShort(0x8),	/* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RP dc), /* Offset= 0x1d, 0x0, /* 8 */ /* FC_CHAR */	/* FC_ALIGNM8 */ /* FC_END */ */ :-36 (628) */ /* FC_SMFARRAY */ /* 0 */	/* 732 */ /* 734 */ /* 736 */ /* 738 */ /* 740 */	NdrFcShort(0x2), 0x19, NdrFcShort(0x0), NdrFcShort(0x1),	0x1b, 0x1b, 0x1, /* 2 */ /* Corr desc: field po 0x0, /* 0 */ /* Corr flags: early, */ /* FC_SHORT */ 0x5b,	-26 (704) */ /* FC_CARRAY */ /* 1 */ sinter, FC_ULONG */ /* */
/* 658 */ /* 660 */ /* 662 */ /* 664 */ /* 666 */ /* 668 */ /* 670 */	0x8, 0x36, NdrFcShort(0xfffffff) NdrFcShort(0x8),	/* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RP dc), /* Offset= 0x1d, 0x0, /* 8 */	/* FC_ALIGNM8 */ /* FC_END */ */ :-36 (628) */ /* FC_SMFARRAY */	/* 732 */ /* 734 */ /* 736 */ /* 738 */ /* 740 */ /* 742 */ /* 744 */	NdrFcShort(0x2), 0x19, NdrFcShort(0x0), NdrFcShort(0x1), 0x6,	0x1b, 0x1, /* 2 */ /* Corr desc: field po 0x0, /* 0 */ /* Corr flags: early, */ /* FC_SHORT */	*-26 (704) */ /* FC_CARRAY */ /* 1 */ binter, FC_ULONG */ /* */ /* FC_END */
/* 658 */ /* 660 */ /* 662 */ /* 664 */ /* 666 */	0x8, 0x36, NdrFcShort(0xfffffff) NdrFcShort(0x8),	/* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RP dc), /* Offset= 0x1d, 0x0, /* 8 */ /* FC_CHAR */ 0x5b,	/* FC_ALIGNM8 */ /* FC_END */ */ :-36 (628) */ /* FC_SMFARRAY */ /* 0 */ /* FC_END */	/* 732 */ /* 734 */ /* 736 */ /* 738 */ /* 740 */ /* 742 */ /* 744 */	NdrFcShort(0x2), 0x19, NdrFcShort(0x0), NdrFcShort(0x1),	0x1b, 0x1b, 0x1, /* 2 */ /* Corr desc: field po 0x0, /* 0 */ /* Corr flags: early, */ /* FC_SHORT */ 0x5b,	*-26 (704) */ /* FC_CARRAY */ /* 1 */ binter, FC_ULONG */ /* */ /* FC_END */ /*
/* 658 */ /* 660 */ /* 662 */ /* 664 */ /* 666 */ /* 668 */ /* 670 */	0x8, 0x36, NdrFcShort(0xfffffff) NdrFcShort(0x8),	/* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RP dc), /* Offset= 0x1d, 0x0, /* 8 */ /* FC_CHAR */ 0x5b, 0x15,	/* FC_ALIGNM8 */ /* FC_END */ */ :-36 (628) */ /* FC_SMFARRAY */ /* 0 */ /* FC_END */ /* FC_STRUCT */	/* 732 */ /* 734 */ /* 736 */ /* 738 */ /* 740 */ /* 742 */ FC_BOGI	NdrFcShort(0x2), 0x19, NdrFcShort(0x0), NdrFcShort(0x1), 0x6,	0x1b, 0x1b, 0x1, /* 2 */ /* Corr desc: field po 0x0, /* 0 */ /* Corr flags: early, */ /* FC_SHORT */ 0x5b, 0x1a, 0x3,	*-26 (704) */ /* FC_CARRAY */ /* 1 */ binter, FC_ULONG */ /* */ /* FC_END */
/* 658 */ /* 660 */ /* 662 */ /* 664 */ /* 666 */ /* 670 */ /* 672 */	0x8, 0x36, NdrFcShort(0xfffffff) NdrFcShort(0x8), 0x2,	/* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RP dc), /* Offset= 0x1d, 0x0, /* 8 */ /* FC_CHAR */ 0x5b, 0x15, 0x3,	/* FC_ALIGNM8 */ /* FC_END */ */ :-36 (628) */ /* FC_SMFARRAY */ /* 0 */ /* FC_END */	/* 732 */ /* 734 */ /* 736 */ /* 738 */ /* 740 */ /* 742 */ FC_BOGI /* 746 */	NdrFcShort(0x2), 0x19, NdrFcShort(0x0), NdrFcShort(0x1), 0x6, US_STRUCT */	0x1b, 0x1b, 0x1, /* 2 */ /* Corr desc: field po 0x0, /* 0 */ /* Corr flags: early, */ /* FC_SHORT */ 0x5b, 0x1a, 0x3, /* 16 */	*-26 (704) */ /* FC_CARRAY */ /* 1 */ binter, FC_ULONG */ /* */ /* FC_END */ /*
/* 658 */ /* 660 */ /* 662 */ /* 664 */ /* 666 */ /* 670 */ /* 672 */	0x8, 0x36, NdrFcShort(0xffffffffffffffffffffffffffffffffffff	/* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RP dc), /* Offset= 0x1d, 0x0, /* 8 */ /* FC_CHAR */ 0x5b, 0x15, 0x3, /* 16 */	/* FC_ALIGNM8 */ /* FC_END */ */ :-36 (628) */ /* FC_SMFARRAY */ /* 0 */ /* FC_END */ /* FC_STRUCT */	/* 732 */ /* 734 */ /* 736 */ /* 738 */ /* 740 */ /* 742 */ FC_BOGI /* 746 */ /* 748 */	NdrFcShort(0x2), 0x19, NdrFcShort(0x0), NdrFcShort(0x1), 0x6, US_STRUCT */ NdrFcShort(0x10), NdrFcShort(0x0),	0x1b, 0x1b, 0x1, /* 2 */ /* Corr desc: field po 0x0, /* 0 */ /* Corr flags: early, */ /* FC_SHORT */ 0x5b, 0x1a, 0x3, /* 16 */ /* 0 */	*-26 (704) */ /* FC_CARRAY */ /* 1 */ binter, FC_ULONG */ /* */ /* FC_END */ /*
/* 658 */ /* 660 */ /* 662 */ /* 664 */ /* 666 */ /* 670 */ /* 672 */	0x8, 0x36, NdrFcShort(0xffffffffffffffffffffffffffffffffffff	/* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RP dc), /* Offset= 0x1d, 0x0, /* 8 */ /* FC_CHAR */ 0x5b, 0x15, 0x3, /* 16 */ /* FC_LONG */	/* FC_ALIGNM8 */ /* FC_END */ */ *-36 (628) */ /* FC_SMFARRAY */ /* 0 */ /* FC_END */ /* FC_STRUCT */ /* 3 */	/* 732 */ /* 734 */ /* 736 */ /* 738 */ /* 740 */ /* 742 */ FC_BOGI /* 746 */ /* 748 */ /* 750 */	NdrFcShort(0x2), 0x19, NdrFcShort(0x0), NdrFcShort(0x1), 0x6, US_STRUCT */ NdrFcShort(0x10), NdrFcShort(0x0), NdrFcShort(0x6),	0x1b, 0x1b, 0x1, /* 2 */ /* Corr desc: field po 0x0, /* 0 */ /* Corr flags: early, */ /* FC_SHORT */ 0x5b, 0x1a, 0x3, /* 16 */ /* 0 */ /* Offset= 6 (756) */	*-26 (704) */ /* FC_CARRAY */ /* 1 */ binter, FC_ULONG */ /* */ /* FC_END */ /*
/* 658 */ /* 660 */ /* 662 */ /* 664 */ /* 666 */ /* 670 */ /* 672 */ /* 676 */	0x8, 0x36, NdrFcShort(0xffffffffffffffffffffffffffffffffffff	/* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RP dc), /* Offset= 0x1d, 0x0, /* 8 */ /* FC_CHAR */ 0x5b, 0x15, 0x3, /* 16 */ /* FC_LONG */ 0x6,	/* FC_ALIGNM8 */ /* FC_END */ */ :-36 (628) */ /* FC_SMFARRAY */ /* 0 */ /* FC_END */ /* FC_STRUCT */	/* 732 */ /* 734 */ /* 736 */ /* 738 */ /* 740 */ /* 742 */ FC_BOGI /* 746 */ /* 748 */	NdrFcShort(0x2), 0x19, NdrFcShort(0x0), NdrFcShort(0x1), 0x6, US_STRUCT */ NdrFcShort(0x10), NdrFcShort(0x0), NdrFcShort(0x6),	0x1b, 0x1b, 0x1, /* 2 */ /* Corr desc: field po 0x0, /* 0 */ /* Corr flags: early, */ /* FC_SHORT */ 0x5b, 0x1a, 0x3, /* 16 */ /* 0 */ /* Offset= 6 (756) */ /* FC_LONG */	*-26 (704) */ /* FC_CARRAY */ /* 1 */ pointer, FC_ULONG */ /* */ /* FC_END */ /* 3 */
/* 658 */ /* 660 */ /* 662 */ /* 664 */ /* 666 */ /* 670 */ /* 672 */	0x8, 0x36, NdrFcShort(0xffffffffffffffffffffffffffffffffffff	/* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RP dc), /* Offset= 0x1d, 0x0, /* 8 */ /* FC_CHAR */ 0x5b, 0x15, 0x3, /* 16 */ /* FC_LONG */ 0x6, /* FC_SHORT */	/* FC_ALIGNM8 */ /* FC_END */ */ :-36 (628) */ /* FC_SMFARRAY */ /* 0 */ /* FC_END */ /* FC_STRUCT */ /* 3 */ /* FC_SHORT */	/* 732 */ /* 734 */ /* 736 */ /* 738 */ /* 740 */ /* 742 */ FC_BOGI /* 746 */ /* 748 */ /* 750 */ /* 752 */	NdrFcShort(0x2), 0x19, NdrFcShort(0x0), NdrFcShort(0x1), 0x6, US_STRUCT */ NdrFcShort(0x10), NdrFcShort(0x0), NdrFcShort(0x6), 0x8,	0x1b, 0x1b, 0x1, /* 2 */ /* Corr desc: field po 0x0, /* 0 */ /* Corr flags: early, */ /* FC_SHORT */ 0x5b, 0x1a, 0x3, /* 16 */ /* 0 */ /* Offset= 6 (756) */ /* FC_LONG */ 0x39,	*-26 (704) */ /* FC_CARRAY */ /* 1 */ binter, FC_ULONG */ /* */ /* FC_END */ /*
/* 658 */ /* 660 */ /* 662 */ /* 664 */ /* 666 */ /* 670 */ /* 674 */ /* 676 */ /* 678 */	0x8, 0x36, NdrFcShort(0xffffffffffffffffffffffffffffffffffff	/* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RP dc), /* Offset= 0x1d, 0x0, /* 8 */ /* FC_CHAR */ 0x5b, 0x15, 0x3, /* 16 */ /* FC_LONG */ 0x6, /* FC_SHORT */ 0x4c,	/* FC_ALIGNM8 */ /* FC_END */ */ *-36 (628) */ /* FC_SMFARRAY */ /* 0 */ /* FC_END */ /* FC_STRUCT */ /* 3 */	/* 732 */ /* 734 */ /* 736 */ /* 738 */ /* 740 */ /* 742 */ FC_BOGI /* 746 */ /* 748 */ /* 750 */	NdrFcShort(0x2), 0x19, NdrFcShort(0x0), NdrFcShort(0x1), 0x6, US_STRUCT */ NdrFcShort(0x10), NdrFcShort(0x0), NdrFcShort(0x6), 0x8,	0x1b, 0x1, 0x1, /* 2 */ /* Corr desc: field po 0x0, /* 0 */ /* Corr flags: early, */ /* FC_SHORT */ 0x5b, 0x1a, 0x3, /* 16 */ /* 0 */ /* Offset= 6 (756) */ /* FC_LONG */ 0x39, /* FC_POINTER */	*-26 (704) */ /* FC_CARRAY */ /* 1 */ pinter, FC_ULONG */ /* */ */ /* FC_END */ /* 3 */ /* FC_ALIGNM8 */
/* 658 */ /* 660 */ /* 662 */ /* 664 */ /* 666 */ /* 670 */ /* 674 */ /* 676 */ /* 678 */	0x8, 0x36, NdrFcShort(0xffffffffffffffffffffffffffffffffffff	/* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RP dc), /* Offset= 0x1d, 0x0, /* 8 */ /* FC_CHAR */ 0x5b, 0x15, 0x3, /* 16 */ /* FC_LONG */ 0x6, /* FC_SHORT */ 0x4c,	/* FC_ALIGNM8 */ /* FC_END */ */ :-36 (628) */ /* FC_SMFARRAY */ /* 0 */ /* FC_END */ /* FC_STRUCT */ /* 3 */ /* FC_SHORT */	/* 732 */ /* 734 */ /* 736 */ /* 738 */ /* 740 */ /* 742 */ FC_BOGI /* 746 */ /* 748 */ /* 750 */ /* 752 */	NdrFcShort(0x2), 0x19, NdrFcShort(0x0), NdrFcShort(0x1), 0x6, US_STRUCT */ NdrFcShort(0x10), NdrFcShort(0x0), NdrFcShort(0x6), 0x8,	0x1b, 0x1b, 0x1, /* 2 */ /* Corr desc: field po 0x0, /* 0 */ /* Corr flags: early, */ /* FC_SHORT */ 0x5b, 0x1a, 0x3, /* 16 */ /* 0 */ /* Offset= 6 (756) */ /* FC_LONG */ 0x39,	*-26 (704) */ /* FC_CARRAY */ /* 1 */ pointer, FC_ULONG */ /* */ /* FC_END */ /* 3 */
/* 658 */ /* 660 */ /* 662 */ /* 664 */ /* 666 */ /* 670 */ /* 674 */ /* 676 */ /* 678 */	0x8, 0x36, NdrFcShort(0xffffffffffffffffffffffffffffffffffff	/* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RP dc), /* Offset= 0x1d, 0x0, /* 8 */ /* FC_CHAR */ 0x5b, 0x15, 0x3, /* 16 */ /* FC_LONG */ 0x6, /* FC_SHORT */ 0x4c,	/* FC_ALIGNM8 */ /* FC_END */ */ :-36 (628) */ /* FC_SMFARRAY */ /* 0 */ /* FC_END */ /* FC_STRUCT */ /* 3 */ /* FC_SHORT */	/* 732 */ /* 734 */ /* 736 */ /* 738 */ /* 740 */ /* 742 */ FC_BOGI /* 746 */ /* 748 */ /* 750 */ /* 752 */	NdrFcShort(0x2), 0x19, NdrFcShort(0x0), NdrFcShort(0x1), 0x6, US_STRUCT */ NdrFcShort(0x10), NdrFcShort(0x0), NdrFcShort(0x6), 0x8,	0x1b, 0x1b, 0x1, /* 2 */ /* Corr desc: field po 0x0, /* 0 */ /* Corr flags: early, */ /* FC_SHORT */ 0x5b, 0x1a, 0x3, /* 16 */ /* 0 */ /* Offset= 6 (756) */ /* ST_LONG */ 0x39, /* FC_POINTER */ 0x5b,	*-26 (704) */ /* FC_CARRAY */ /* 1 */ binter, FC_ULONG */ /* */ /* FC_END */ /* GEND */ /* FC_ALIGNM8 */ /* FC_END */
/* 658 */ /* 660 */ /* 662 */ /* 664 */ /* 666 */ /* 670 */ /* 672 */ /* 674 */ /* 678 */ FC_EMB	0x8, 0x36, NdrFcShort(0xffffffffffffffffffffffffffffffffffff	/* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RP dc), /* Offset= 0x1d, 0x0, /* 8 */ /* FC_CHAR */ 0x5b, 0x15, 0x3, /* 16 */ /* FC_LONG */ 0x6, /* FC_SHORT */ 0x4c, */	/* FC_ALIGNM8 */ /* FC_END */ */ :-36 (628) */ /* FC_SMFARRAY */ /* FC_END */ /* FC_STRUCT */ /* 3 */ /* FC_SHORT */ /*	/* 732 */ /* 734 */ /* 736 */ /* 738 */ /* 740 */ /* 742 */ FC_BOGI /* 746 */ /* 750 */ /* 752 */ /* 754 */	NdrFcShort(0x2), 0x19, NdrFcShort(0x0), NdrFcShort(0x1), 0x6, US_STRUCT */ NdrFcShort(0x10), NdrFcShort(0x0), NdrFcShort(0x6), 0x8,	0x1b, 0x1b, 0x1, /* 2 */ /* Corr desc: field po 0x0, /* 0 */ /* Corr flags: early, */ /* FC_SHORT */ 0x5b, 0x1a, 0x3, /* 16 */ /* 0 */ /* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b,	*-26 (704) */ /* FC_CARRAY */ /* 1 */ binter, FC_ULONG */ /* */ /* FC_END */ /* 3 */ /* FC_ALIGNM8 */ /* FC_END */ */ */
/* 658 */ /* 660 */ /* 662 */ /* 664 */ /* 666 */ /* 670 */ /* 672 */ /* 674 */ /* 678 */ FC_EMB	0x8, 0x36, NdrFcShort(0xffffffffffffffffffffffffffffffffffff	/* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RP dc), /* Offset= 0x1d, 0x0, /* 8 */ /* FC_CHAR */ 0x5b, 0x15, 0x3, /* 16 */ /* FC_LONG */ 0x6, /* FC_SHORT */ 0x4c, */ /* 0 */	/* FC_ALIGNM8 */ /* FC_END */ */ :-36 (628) */ /* FC_SMFARRAY */ /* FC_END */ /* FC_STRUCT */ /* 3 */ /* FC_SHORT */ /*	/* 732 */ /* 734 */ /* 736 */ /* 738 */ /* 740 */ /* 742 */ FC_BOGI /* 746 */ /* 750 */ /* 752 */ /* 754 */	NdrFcShort(0x2), 0x19, NdrFcShort(0x0), NdrFcShort(0x1), 0x6, US_STRUCT */ NdrFcShort(0x10), NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x36,	0x1b, 0x1b, 0x1, /* 2 */ /* Corr desc: field po 0x0, /* 0 */ /* Corr flags: early, */ /* FC_SHORT */ 0x5b, 0x1a, 0x3, /* 16 */ /* 0 */ /* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b,	*-26 (704) */ /* FC_CARRAY */ /* 1 */ binter, FC_ULONG */ /* */ /* FC_END */ /* 3 */ /* FC_ALIGNM8 */ /* FC_END */ */ */
/* 658 */ /* 660 */ /* 662 */ /* 664 */ /* 666 */ /* 670 */ /* 672 */ /* 674 */ /* 678 */ FC_EMB /* 680 */	0x8, 0x36, NdrFcShort(0xffffffffffffffffffffffffffffffffffff	/* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RP dc), /* Offset= 0x1d, 0x0, /* 8 */ /* FC_CHAR */ 0x5b, 0x15, 0x3, /* 16 */ /* FC_LONG */ 0x6, /* FC_SHORT */ 0x4c, */ NdrFcShort(0xfffffff)	/* FC_ALIGNM8 */ /* FC_END */ */ :-36 (628) */ /* FC_SMFARRAY */ /* FC_END */ /* FC_STRUCT */ /* 3 */ /* FC_SHORT */ /* /* ft_), /* Offset=	/* 732 */ /* 734 */ /* 736 */ /* 738 */ /* 740 */ /* 742 */ FC_BOGI /* 746 */ /* 750 */ /* 752 */ /* 754 */ /* 758 */	NdrFcShort(0x2), 0x19, NdrFcShort(0x0), NdrFcShort(0x1), 0x6, US_STRUCT */ NdrFcShort(0x10), NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x36,	0x1b, 0x1b, 0x1, /* 2 */ /* Corr desc: field po 0x0, /* 0 */ /* Corr flags: early, */ /* FC_SHORT */ 0x5b, 0x1a, 0x3, /* 16 */ /* 0 */ /* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b,	*-26 (704) */ /* FC_CARRAY */ /* 1 */ binter, FC_ULONG */ /* */ /* FC_END */ /* GEND */ /* FC_ALIGNM8 */ /* FC_END */
/* 658 */ /* 660 */ /* 662 */ /* 664 */ /* 666 */ /* 670 */ /* 672 */ /* 678 */ FC_EMB /* 680 */ -15 (666)	0x8, 0x36, NdrFcShort(0xffffffffffffffffffffffffffffffffffff	/* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RP dc), /* Offset= 0x1d, 0x0, /* 8 */ /* FC_CHAR */ 0x5b, 0x15, 0x3, /* 16 */ /* FC_LONG */ 0x6, /* FC_SHORT */ 0x4c, */ /* 0 */	/* FC_ALIGNM8 */ /* FC_END */ */ :-36 (628) */ /* FC_SMFARRAY */ /* FC_END */ /* FC_STRUCT */ /* 3 */ /* FC_SHORT */ /*	/* 732 */ /* 734 */ /* 736 */ /* 738 */ /* 740 */ /* 742 */ /* 744 */ FC_BOGI /* 746 */ /* 750 */ /* 752 */ /* 754 */	NdrFcShort(0x2), 0x19, NdrFcShort(0x0), NdrFcShort(0x1), 0x6, US_STRUCT */ NdrFcShort(0x10), NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x36,	0x1b, 0x1b, 0x1, /* 2 */ /* Corr desc: field po 0x0, /* 0 */ /* Corr flags: early, */ /* Corr flags: early, */ /* TC_SHORT */ 0x5b, 0x1a, 0x3, /* 16 */ /* 0 */ /* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b,	*-26 (704) */ /* FC_CARRAY */ /* 1 */ pointer, FC_ULONG */ /* */ /* FC_END */ /* 3 */ /* FC_ALIGNM8 */ /* FC_END */ */ *-26 (732) */
/* 658 */ /* 660 */ /* 662 */ /* 664 */ /* 666 */ /* 670 */ /* 672 */ /* 674 */ /* 678 */ FC_EMB /* 680 */	0x8, 0x36, NdrFcShort(0xffffffffffffffffffffffffffffffffffff	/* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RP dc), /* Offset= 0x1d, 0x0, /* 8 */ /* FC_CHAR */ 0x5b, 0x15, 0x3, /* 16 */ /* FC_LONG */ 0x6, /* FC_SHORT */ 0x4c, */ /* O */ NdrFeShort(0xfffffff) 0x5b,	/* FC_ALIGNM8 */ /* FC_END */ */ -36 (628) */ /* FC_SMFARRAY */ /* 0 */ /* FC_END */ /* FC_STRUCT */ /* 3 */ /* FC_SHORT */ /* /* FI_), /* Offset= /* FC_END */	/* 732 */ /* 734 */ /* 736 */ /* 738 */ /* 740 */ /* 742 */ FC_BOGI /* 746 */ /* 750 */ /* 752 */ /* 754 */ /* 758 */	NdrFcShort(0x2), 0x19, NdrFcShort(0x0), NdrFcShort(0x1), 0x6, US_STRUCT */ NdrFcShort(0x10), NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x36,	0x1b, 0x1b, 0x1, /* 2 */ /* Corr desc: field po 0x0, /* 0 */ /* Corr flags: early, */ /* FC_SHORT */ 0x5b, 0x1a, 0x3, /* 16 */ /* 0 */ /* OFfset= 6 (756) */ /* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b,	*-26 (704) */ /* FC_CARRAY */ /* 1 */ pointer, FC_ULONG */ /* */ /* FC_END */ /* 3 */ /* FC_END */ ** /* FC_END */ ** /* FC_END */ ** -26 (732) */ /* FC_CARRAY */
/* 658 */ /* 660 */ /* 662 */ /* 664 */ /* 666 */ /* 670 */ /* 672 */ /* 678 */ FC_EMB /* 680 */ -15 (666) /* 684 */	0x8, 0x36, NdrFcShort(0xffffffffffffffffffffffffffffffffffff	/* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RP dc), /* Offset= 0x1d, 0x0, /* 8 */ /* FC_CHAR */ 0x5b, 0x15, 0x3, /* 16 */ /* FC_LONG */ 0x6, /* FC_SHORT */ 0x4c, */ NdrFcShort(0xfffffff)	/* FC_ALIGNM8 */ /* FC_END */ */ :-36 (628) */ /* FC_SMFARRAY */ /* FC_END */ /* FC_STRUCT */ /* 3 */ /* FC_SHORT */ /* /* ft_), /* Offset=	/* 732 */ /* 734 */ /* 736 */ /* 738 */ /* 740 */ /* 742 */ FC_BOGI /* 746 */ /* 750 */ /* 754 */ /* 756 */ /* 760 */	NdrFcShort(0x2), 0x19, NdrFcShort(0x0), NdrFcShort(0x1), 0x6, US_STRUCT */ NdrFcShort(0x10), NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x36,	0x1b, 0x1b, 0x1, /* 2 */ /* Corr desc: field po 0x0, /* 0 */ /* Corr flags: early, */ /* FC_SHORT */ 0x5b, 0x1a, 0x3, /* 16 */ /* 0 */ /* Offset= 6 (756) */ /* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x12, 0x0, /* FC_UP 26), /* Offset=	*-26 (704) */ /* FC_CARRAY */ /* 1 */ pointer, FC_ULONG */ /* */ /* FC_END */ /* 3 */ /* FC_ALIGNM8 */ /* FC_END */ */ *-26 (732) */
/* 658 */ /* 660 */ /* 662 */ /* 664 */ /* 666 */ /* 670 */ /* 672 */ /* 678 */ FC_EMB /* 680 */ -15 (666) /* 684 */	0x8, 0x36, NdrFcShort(0xffffffffffffffffffffffffffffffffffff	/* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RP dc), /* Offset= 0x1d, 0x0, /* 8 */ /* FC_CHAR */ 0x5b, 0x15, 0x3, /* 16 */ /* FC_LONG */ 0x6, /* FC_SHORT */ 0x4c, */ NdrFcShort(0xfffffff) 0x5b, 0x1a,	/* FC_ALIGNM8 */ /* FC_END */ */ :-36 (628) */ /* FC_SMFARRAY */ /* O */ /* FC_END */ /* FC_STRUCT */ /* 3 */ /* FC_SHORT */ /* f1), /* Offset= /* FC_END */ /*	/* 732 */ /* 734 */ /* 736 */ /* 738 */ /* 740 */ /* 742 */ FC_BOGI /* 746 */ /* 750 */ /* 752 */ /* 756 */ /* 760 */ /* 762 */	NdrFcShort(0x2), 0x19, NdrFcShort(0x0), NdrFcShort(0x1), 0x6, US_STRUCT */ NdrFcShort(0x10), NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x36, NdrFcShort(0x4),	0x1b, 0x1b, 0x1, /* 2 */ /* Corr desc: field po 0x0, /* 0 */ /* Corr flags: early, */ /* FC_SHORT */ 0x5b, 0x1a, 0x3, /* 16 */ /* 0 */ /* Offset= 6 (756) */ /* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x12, 0x0, /* FC_UP 26), /* Offset= 0x1b, 0x3, /* 4 */	*-26 (704) */ /* FC_CARRAY */ /* 1 */ binter, FC_ULONG */ /* */ /* FC_END */ /* 3 */ /* FC_END */ ** /* FC_END */ */ *-26 (732) */ /* FC_CARRAY */ /* 3 */
/* 658 */ /* 660 */ /* 662 */ /* 664 */ /* 666 */ /* 670 */ /* 672 */ /* 678 */ FC_EMB /* 680 */ -15 (666) /* 684 */ FC_BOG	0x8, 0x36, NdrFcShort(0xffffffffffffffffffffffffffffffffffff	/* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RP dc), /* Offset= 0x1d, 0x0, /* 8 */ /* FC_CHAR */ 0x5b, 0x15, 0x3, /* 16 */ /* FC_LONG */ 0x6, /* FC_SHORT */ 0x4c, */ /* 0 */ NdrFcShort(0xfffffff) 0x5b, 0x1a, 0x3,	/* FC_ALIGNM8 */ /* FC_END */ */ -36 (628) */ /* FC_SMFARRAY */ /* 0 */ /* FC_END */ /* FC_STRUCT */ /* 3 */ /* FC_SHORT */ /* /* FI_), /* Offset= /* FC_END */	/* 732 */ /* 734 */ /* 736 */ /* 738 */ /* 740 */ /* 742 */ FC_BOGI /* 746 */ /* 750 */ /* 754 */ /* 756 */ /* 760 */	NdrFcShort(0x2), 0x19, NdrFcShort(0x0), NdrFcShort(0x1), 0x6, US_STRUCT */ NdrFcShort(0x10), NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x36, NdrFcShort(0x4),	0x1b, 0x1b, 0x1, /* 2 */ /* Corr desc: field po 0x0, /* 0 */ /* Corr flags: early, */ /* FC_SHORT */ 0x5b, 0x1a, 0x3, /* 16 */ /* 0 */ /* Offset= 6 (756) */ /* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x12, 0x0, /* FC_UP 26), /* Offset= 0x1b, 0x3, /* 4 */ /* Corr desc: field po	*/ FC_CARRAY */ /* FC_CARRAY */ /* 1 */ pointer, FC_ULONG */ /* FC_END */ /* FC_END */ /* FC_END */ */ */ /* FC_END */ */ */ */ */ */ */ */ */ */
/* 658 */ /* 660 */ /* 662 */ /* 664 */ /* 666 */ /* 668 */ /* 672 */ /* 676 */ /* 678 */ FC_EMB /* 680 */ -15 (666) /* 684 */ FC_BOG /* 686 */	0x8, 0x36, NdrFcShort(0xffffffffffffffffffffffffffffffffffff	/* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RP dc), /* Offset= 0x1d, 0x0, /* 8 */ /* FC_CHAR */ 0x5b, 0x15, 0x3, /* 16 */ /* FC_LONG */ 0x6, /* FC_SHORT */ 0x4c, */ /* 0 */ NdrFcShort(0xffffffff) 0x5b, 0x1a, 0x3, /* 32 */	/* FC_ALIGNM8 */ /* FC_END */ */ :-36 (628) */ /* FC_SMFARRAY */ /* O */ /* FC_END */ /* FC_STRUCT */ /* 3 */ /* FC_SHORT */ /* f1), /* Offset= /* FC_END */ /*	/* 732 */ /* 734 */ /* 736 */ /* 738 */ /* 740 */ /* 742 */ /* 744 */ FC_BOGI /* 746 */ /* 750 */ /* 752 */ /* 756 */ /* 760 */ /* 764 */	NdrFcShort(0x2), 0x19, NdrFcShort(0x0), NdrFcShort(0x1), 0x6, US_STRUCT */ NdrFcShort(0x10), NdrFcShort(0x6), 0x8, 0x36, NdrFcShort(0xffffffft NdrFcShort(0x4), 0x19,	0x1b, 0x1b, 0x1, /* 2 */ /* Corr desc: field po 0x0, /* 0 */ /* Corr flags: early, */ /* FC_SHORT */ 0x5b, 0x1a, 0x3, /* 16 */ /* 0 */ /* Offset= 6 (756) */ /* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x12, 0x0, /* FC_UP 26), /* Offset= 0x1b, 0x3, /* 4 */ /* Corr desc: field po 0x0,	*-26 (704) */ /* FC_CARRAY */ /* 1 */ binter, FC_ULONG */ /* */ /* FC_END */ /* 3 */ /* FC_END */ ** /* FC_END */ */ *-26 (732) */ /* FC_CARRAY */ /* 3 */
/* 658 */ /* 660 */ /* 662 */ /* 664 */ /* 666 */ /* 668 */ /* 672 */ /* 676 */ /* 678 */ FC_EMB /* 680 */ -15 (666) /* 684 */ FC_BOG /* 686 */ /* 688 */	0x8, 0x36, NdrFcShort(0xffffffffffffffffffffffffffffffffffff	/* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RP dc), /* Offset= 0x1d, 0x0, /* 8 */ /* FC_CHAR */ 0x5b, 0x15, 0x3, /* 16 */ /* FC_LONG */ 0x6, /* FC_SHORT */ 0x4c, */ /* 0 */ NdrFcShort(0xffffffff 0x5b, 0x1a, 0x3, /* 32 */ /* 0 */	/* FC_ALIGNM8 */ /* FC_END */ */ :-36 (628) */ /* FC_SMFARRAY */ /* FC_END */ /* FC_STRUCT */ /* 3 */ /* FC_SHORT */ /* /* FC_END */ /* /* GI), /* Offset= /* FC_END */ /* /* 3 */	/* 732 */ /* 734 */ /* 736 */ /* 738 */ /* 740 */ /* 742 */ /* 744 */ FC_BOGI /* 746 */ /* 752 */ /* 752 */ /* 756 */ /* 764 */ /* 764 */ /* 766 */	NdrFcShort(0x2), 0x19, NdrFcShort(0x0), NdrFcShort(0x1), 0x6, US_STRUCT */ NdrFcShort(0x10), NdrFcShort(0x0), NdrFcShort(0x6), 0x8, 0x36, NdrFcShort(0x4), 0x19, NdrFcShort(0x0),	0x1b, 0x1b, 0x1, /* 2 */ /* Corr desc: field po 0x0, /* 0 */ /* Corr flags: early, */ /* FC_SHORT */ 0x5b, 0x1a, 0x3, /* 16 */ /* 0 */ /* Offset= 6 (756) */ /* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x12, 0x0, /* FC_UP 26), /* Offset= 0x1b, 0x3, /* 4 */ /* Corr desc: field po 0x0, /* 0 */	*-26 (704) */ /* FC_CARRAY */ /* 1 */ binter, FC_ULONG */ /* */ /* FC_END */ /* 3 */ /* FC_END */ ** -26 (732) */ /* FC_CARRAY */ /* 3 */ binter, FC_ULONG */ /* */ binter, FC_ULONG */ /* */
/* 658 */ /* 660 */ /* 662 */ /* 664 */ /* 666 */ /* 668 */ /* 672 */ /* 676 */ /* 678 */ FC_EMB /* 680 */ -15 (666) /* 684 */ FC_BOG /* 686 */ /* 688 */	0x8, 0x36, NdrFcShort(0xffffffffffffffffffffffffffffffffffff	/* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x11, 0x0, /* FC_RP dc), /* Offset= 0x1d, 0x0, /* 8 */ /* FC_CHAR */ 0x5b, 0x15, 0x3, /* 16 */ /* FC_LONG */ 0x6, /* FC_SHORT */ 0x4c, */ /* 0 */ NdrFcShort(0xffffffff) 0x5b, 0x1a, 0x3, /* 32 */	/* FC_ALIGNM8 */ /* FC_END */ */ :-36 (628) */ /* FC_SMFARRAY */ /* FC_END */ /* FC_STRUCT */ /* 3 */ /* FC_SHORT */ /* /* FC_END */ /* /* GI), /* Offset= /* FC_END */ /* /* 3 */	/* 732 */ /* 734 */ /* 736 */ /* 738 */ /* 740 */ /* 742 */ /* 744 */ FC_BOGI /* 746 */ /* 752 */ /* 752 */ /* 756 */ /* 764 */ /* 764 */ /* 766 */	NdrFcShort(0x2), 0x19, NdrFcShort(0x0), NdrFcShort(0x1), 0x6, US_STRUCT */ NdrFcShort(0x10), NdrFcShort(0x6), 0x8, 0x36, NdrFcShort(0xffffffft NdrFcShort(0x4), 0x19,	0x1b, 0x1b, 0x1, /* 2 */ /* Corr desc: field po 0x0, /* 0 */ /* Corr flags: early, */ /* FC_SHORT */ 0x5b, 0x1a, 0x3, /* 16 */ /* 0 */ /* Offset= 6 (756) */ /* FC_LONG */ 0x39, /* FC_POINTER */ 0x5b, 0x12, 0x0, /* FC_UP 26), /* Offset= 0x1b, 0x3, /* 4 */ /* Corr desc: field po 0x0,	*-26 (704) */ /* FC_CARRAY */ /* 1 */ binter, FC_ULONG */ /* */ /* FC_END */ /* 3 */ /* FC_END */ ** -26 (732) */ /* FC_CARRAY */ /* 3 */ binter, FC_ULONG */ /* */ binter, FC_ULONG */ /* */

/* 770 */ 0x8,	/* FC LONG */		/* 848 */ 0x6,	/* FC_SHORT */	
,	0x5b,	/* FC_END */	,	0x6,	/* FC_SHORT */
/* 772 */	,	, ,	/* 850 */ 0x38,	/* FC_ALIGNM4 *	
, ,,= ,	0x1a,	/*	, 650 , 6150,	0x8,	/* FC_LONG */
FC_BOGUS_STRUCT */	oxiu,	,	/* 852 */ 0x8,	/* FC_LONG */	/ TC_LONG /
re_bodos_street /	0x3,	/* 3 */	/ 832 / 086,	0x4c,	/*
/* 774 */ NJ-E-Cl(0-		/ 3 /	EC EMBEDDED COMPLEY		, .
/* 774 */ NdrFcShort(0x			FC_EMBEDDED_COMPLEX		
/* 776 */ NdrFcShort(0x			/* 854 */ 0x4,	/* 4 */	(1)
/* 778 */ NdrFcShort(0x		/	1	NdrFcShort(0xfffff	fe0d), /* Offset=
/* 780 */ 0x8,	/* FC_LONG */		-499 (356) */		
	0x39,	/* FC_ALIGNM8 */		0x5b,	/* FC_END */
/* 782 */ 0x36,	/* FC_POINTER */		/* 858 */		
	0x5b,	/* FC_END */		0x12, 0x0, /* FC_U	
/* 784 */			/* 860 */ NdrFcShort(0xfffff	ff02), /* Offse	t= -254 (606) */
	0x12, 0x0, /* FC_U	P */	/* 862 */		
/* 786 */ NdrFcShort(0x	ffffffe6), /* Offset	= -26 (760) */		0x12, 0x8, /* FC U	JP [simple_pointer] */
/* 788 */	<i>"</i>	` ′	/* 864 */ 0x1,	/* FC_BYTE */	
	0x1b,	/* FC_CARRAY */	,	0x5c,	/* FC_PAD */
	0x7,	/* 7 */	/* 866 */	*******	,
/* 790 */ NdrFcShort(0x	,		, 666 ,	0x12 0x8 /* FC L	JP [simple_pointer] */
/* 792 */ 0x19,		oointer, FC_ULONG */	/* 868 */ 0x6,	/* FC_SHORT */	a [simple_pointer]
7 732 7 0X13,	0x0,	/* */	7 000 7 000,	0x5c,	/* FC_PAD */
/* 794 */ NdrFcShort(0x	,	, ,	/* 870 */	oase,	/ IC_IAD /
	,,	* /	/ 8/0 -/	0-12 0-0 /* EC I	ID [-i1i4] */
/* 796 */ NdrFcShort(0x	1), /* Corr flags: early,	, */	/* 0.72 */ O O		JP [simple_pointer] */
/* 798 */ 0xb,	/* FC_HYPER */	# EG END #/	/* 872 */ 0x8,	/* FC_LONG */	#EC DID #/
W 000 #/	0x5b,	/* FC_END */	(t) 054 t/	0x5c,	/* FC_PAD */
/ * 800 * /			/* 874 */		
	0x1a,	/*			JP [simple_pointer] */
FC_BOGUS_STRUCT */			/* 876 */ 0xa,	/* FC_FLOAT */	
	0x3,	/* 3 */		0x5c,	/* FC_PAD */
/* 802 */ NdrFcShort(0x			/* 878 */		
/* 804 */ NdrFcShort(0x	0), /* 0 */			0x12, 0x8, /* FC_U	P [simple_pointer] */
/* 806 */ NdrFcShort(0x	6), /* Offset= 6 (812) *.	/	/* 880 */ 0xc,	/* FC_DOUBLE */	•
/* 808 */ 0x8,	/* FC_LONG */			0x5c,	/* FC_PAD */
	0x39,	/* FC_ALIGNM8 */	/* 882 */		
/* 810 */ 0x36,	/* FC_POINTER */			0x12, 0x0, /* FC U	JP */
	0x5b,	/* FC_END */	/* 884 */ NdrFcShort(0xfffff	fda4), /* Offse	t= -604 (280) */
/* 812 */	•	_	/* 886 */	<i>"</i>	` /
	0x12, 0x0, /* FC_U	P */		0x12, 0x10,	/* FC_UP
/* 814 */ NdrFcShort(0x			[pointer deref] */	, ,	_
/* 816 */	,,	. ()	/* 888 */ NdrFcShort(0xfffff	fda6). /* Offse	t= -602 (286) */
, , , ,	0x15,	/* FC STRUCT */	/* 890 */	,, , , , , , , , , , , , , , , , , , , ,	
	0x3,	/* 3 */	, 6,50 ,	0x12, 0x10,	/* FC_UP
/* 818 */ NdrFcShort(0x	,	, 3 ,	[pointer deref] */	0.112, 0.110,	, 16_61
/* 820 */ 0x8,	/* FC LONG */		/* 892 */ NdrFcShort(0xfffff	fdhc) /* Offse	t= -580 (312) */
/ 820 / 000,	0x8,	/* FC_LONG */	/* 894 */	idoc), / Offsc	1-380 (312)
/* 922 */ Ov.5a		/ rc_Long /	/* 894 */	012 010	/* EC LID
/* 822 */ 0x5c,	/* FC_PAD */	/* EC_END */	[:t	0x12, 0x10,	/* FC_UP
/* O24 */	0x5b,	/* FC_END */	[pointer_deref] */	Edon) /* Off	£ 566 (220) */
/* 824 */	0.11	/VEC CARRAGE	/* 896 */ NdrFcShort(0xfffff	aca), /* Offse	i= -366 (330) */
	0x1b,	/* FC_CARRAY */	/* 898 */	0 10 0 10	# FQ ***
/# 00/ #/ 371 F ~	0x3,	/* 3 */	F	0x12, 0x10,	/* FC_UP
/* 826 */ NdrFcShort(0x		~~~~	[pointer_deref] */		/- 10) ::
/* 828 */ 0x7,	/* Corr desc: FC_US		/* 900 */ NdrFcShort(0xfffff	fdd8), /* Offse	t= -552 (348) */
	0x0,	/* */	/* 902 */		
/* 830 */ NdrFcShort(0x	ffc8), /* -56 */			0x12, 0x10,	/* FC_UP
/* 832 */ NdrFcShort(0x	 /* Corr flags: early, 	*/	[pointer_deref] */		
/* 834 */ 0x4c,	/* FC_EMBEDDED	D_COMPLEX */	/* 904 */ NdrFcShort(0x2),	/* Offset= 2 (906) *	*/
	0x0,	/* 0 */	/* 906 */		
/* 836 */ NdrFcShort(0x	ffffffec), /* Offset	= -20 (816) */		0x12, 0x0, /* FC U	JP */
/* 838 */ 0x5c,	/* FC PAD */		/* 908 */ NdrFcShort(0x16)	, /* Offset= 22 (930)	*/
,	0x5b,	/* FC_END */	/* 910 */	()	
/* 840 */	,	_ '		0x15,	/* FC STRUCT */
	0x1a,	/*		0x7,	/* 7 */
FC BOGUS STRUCT */	***7		/* 912 */ NdrFcShort(0x10)		
	0x3,	/* 3 */	/* 914 */ 0x6,	/* FC SHORT */	
/* 842 */ NdrFcShort(0x		, 3 ,	, , , , , , , , , , , , , , , , , , , ,	0x1,	/* FC BYTE */
/* 844 */ NdrFcShort(0x		= -20 (824) */	/* 916 */ 0x1,	/* FC_BYTE */	, 10_D11L /
/* 846 */ NdrFcShort(0x			, 510 / OA1,	0x38,	/* FC ALIGNM4 */
, 646 / Mail Collott (0A	o j, / Offset o (040)	,		JA30,	, I C_ILLIGININIT /

```
/* 918 */ 0x8,
                               /* FC LONG */
                                                    /* FC ALIGNM8 */
                               0x39,
/* 920 */ 0xb,
                               /* FC HYPER */
                               0x5b,
                                                    /* FC_END */
                                                                             #define _tpcc_com_ps_CHECK_IID(n)
                                                                                                                      IID_GENERIC_CHECK_IID(
/* 922 */
                                                                             _tpcc_com_ps, pIID, n)
                               0x12, 0x0, /* FC UP */
/* 924 */
          NdrFcShort( 0xfffffff2 ),
                                         /* Offset= -14 (910) */
                                                                             int stdcall tpcc com ps IID Lookup(const IID * pIID, int * pIndex)
/* 926 */
                               0x12, 0x8, /* FC UP [simple pointer] */
/* 928 */ 0x2,
                               /* FC CHAR */
                                                                               if(!_tpcc_com_ps_CHECK_IID(0))
                               0x5c,
                                                    /* FC PAD */
/* 930 */
                                                                                 *pIndex = 0;
                               0x1a,
                                                                                 return 1;
FC_BOGUS_STRUCT */
                               0x7,
                               /* 32 */
/* 932 */
          NdrFcShort(0x20),
                                                                               return 0;
/* 934 */
          NdrFcShort( 0x0 ).
                               /* 0 */
/* 936 */ NdrFcShort( 0x0 ),
                               /* Offset= 0 (936) */
/* 938 */ 0x8,
                               /* FC_LONG */
                                                                             const ExtendedProxyFileInfo tpcc_com_ps_ProxyFileInfo =
                                                    /* FC LONG */
                               0x8.
                               /* FC_SHORT */
/* 940 */ 0x6,
                                                                               (PCInterfaceProxyVtblList *) & _tpcc_com_ps_ProxyVtblList,
                                                                               (PCInterfaceStubVtblList *) & tpcc com ps StubVtblList,
                               0x6,
                                                    /* FC SHORT */
/* 942 */ 0x6,
                               /* FC_SHORT */
                                                                               (const PCInterfaceName * ) & _tpcc_com_ps_InterfaceNamesList,
                                                    /* FC SHORT */
                                                                               0, // no delegation
                               0x6.
/* 944 */ 0x4c,
                               /* FC EMBEDDED COMPLEX */
                                                                               & _tpcc_com_ps_IID_Lookup,
                                                    /* 0 */
                               0x0.
/* 946 */ NdrFcShort( 0xfffffc54 ),
                                          /* Offset= -940 (6) */
                               /* FC_PAD */
/* 948 */ 0x5c,
                                                                               0, /* table of [async_uuid] interfaces */
                               0x5b,
                                                    /* FC END */
                                                                               0, /* Filler1 */
/* 950 */ 0xb4,
                               /* FC_USER_MARSHAL */
                                                                               0, /* Filler2 */
                               0x83,
                                                    /* 131 */
                                                                               0 /* Filler3 */
/* 952 */
          NdrFcShort( 0x0 ),
                               /* 0 */
/* 954 */
          NdrFcShort( 0x18 ),
                               /* 24 */
                               /* 0 */
/* 956 */
          NdrFcShort( 0x0 ),
/* 958 */
          NdrFcShort( 0xfffffc44 ),
                                         /* Offset= -956 (2) */
                                                                             #endif /* defined(_M_IA64) || defined(_M_AXP64)*/
/* 960 */
                               0x11, 0x4, /* FC_RP [alloced_on_stack] */
/* 962 */
                               /* Offset= 6 (968) */
          NdrFcShort( 0x6 ),
                                                                             tpcc dblib.cpp
/* 964 */
                               0x13, 0x0, /* FC OP */
                                                                                       FILE:
                                                                                                            TPCC_DBLIB.CPP
/* 966 */ NdrFcShort( 0xffffffdc ),
                                         /* Offset= -36 (930) */
                                                                                                                      Microsoft TPC-C Kit Ver.
                               /* FC_USER_MARSHAL */
/* 968 */
          0xb4
                                                                             4.20.000
                                                    /* 131 */
                               0x83,
                                                                                                                      Copyright Microsoft, 1999
/* 970 */
         NdrFcShort( 0x0 ),
                               /* 0 */
                                                                             *
                                                                                       All Rights Reserved
/* 972 */
          NdrFcShort( 0x18 ), /* 24 */
/* 974 */ NdrFcShort( 0x0 ),
                               /* 0 */
                                                                                                                       Version 4.10.000 audited by
/* 976 */ NdrFcShort( 0xfffffff4 ),
                                         /* Offset= -12 (964) */
                                                                             Richard Gimarc, Performance Metrics, 3/17/99
                               0x0
                                                                                       PURPOSE:
                                                                                                            Implements dblib calls for TPC-C txns.
                                                                                       Contact: Charles Levine (clevine@microsoft.com)
                                                                               Change history:
const CInterfaceProxyVtbl * _tpcc_com_ps_ProxyVtblList[] =
                                                                                                  4.20.000 - updated rev number to match kit
                                                                                                 4.10.001 - not deleting error class in catch handler on
  (CInterfaceProxyVtbl *) & ITPCCProxyVtbl,
                                                                             deadlock retry;
  0
                                                                                                                        not a functional bug, but a
};
                                                                             memory leak
                                                                                                                       - had to tweak some declarations
const CInterfaceStubVtbl * _tpcc_com_ps_StubVtblList[] =
                                                                             to compile with latest SDK; no functional change
  (CInterfaceStubVtbl *) & ITPCCStubVtbl,
  0
                                                                             #include <windows.h>
};
                                                                             #include <stdio.h>
                                                                             #include <assert.h>
PCInterfaceName const _tpcc_com_ps_InterfaceNamesList[] =
                                                                             #define DBNTWIN32
  "ITPCC".
                                                                             #include <sqlfront.h>
  0
                                                                            #include <sqldb.h>
```

```
#ifdef ICECAP
#include <icapexp.h>
#endif
// need to declare functions for export
#define DllDecl declspec( dllexport )
#include "..\..\common\src\error.h"
#include "..\..\common\src\trans.h"
#include "..\..\common\src\txn_base.h"
#include "tpcc_dblib.h"
         DEFCLPACKSIZE
                                                    4096
#define
// version string; must match return value from tpcc_version stored proc
const char sVersion[] = "4.10.000";
const
                               iMaxRetries = 10;
                                                               // how many
retries on deadlock
                     iConnectionCount = 0; // number of current dblib
static
          long
connections
const int iErrOleDbProvider = 7312;
const char sErrTimeoutExpired[] = "Timeout expired";
BOOL APIENTRY DllMain(HMODULE hModule, DWORD
ul_reason_for_call, LPVOID pReserved)
          switch(ul_reason_for_call)
                     case DLL_PROCESS_ATTACH:
                               DisableThreadLibraryCalls(hModule);
                               dbinit();
                                                    // initialize dblib
                               break;
                     case DLL PROCESS DETACH:
                               dbexit();
                                                    // close all dblib
structures/connections
                               break:
                     default:
                               /* nothing */;
          return TRUE;
int err_handler(DBPROCESS *dbproc, int severity, int dberr, int oserr,
LPCSTR dberrstr, LPCSTR oserrstr)
          CTPCC DBLIB
                                                     *pConn;
          assert(dbproc != NULL);
          pConn = (CTPCC DBLIB*)dbgetuserdata(dbproc);
          if (pConn != NULL)
                     pConn->SetDbLibError( severity, dberr, oserr, dberrstr,
oserrstr);
          return INT_CANCEL;
/* FUNCTION: int msg handler(DBPROCESS *dbproc, DBINT msgno, int
msgstate, int severity, char *msgtext)
```

```
* PURPOSE:
                    This function handles DB-Library SQL Server error
messages
                    DBPROCESS
* ARGUMENTS:
                                                   *dbproc
DBPROCESS id pointer
                                         DBINT
msgno
                              message number
                                        int
msgstate
                    message state
                                        int
severity
                    message severity
                                        char
*msgtext
                    printable message description
* RETURNS:
                    continue if error is SQLETIME else INT_CANCEL
INT CONTINUE
action
INT_CANCEL
                              cancel operation
* COMMENTS:
                    This function also sets the dead lock dbproc variable if
necessary.
// typedef INT (SQLAPI *DBMSGHANDLE PROC)(PDBPROCESS, DBINT,
INT, INT, LPCSTR, LPCSTR, LPCSTR, DBUSMALLINT);
int msg handler(DBPROCESS *dbproc, DBINT msgno, int msgstate, int
severity.
                                        LPCSTR msgtext, LPCSTR
srvname, LPCSTR procname, DBUSMALLINT line)
          CTPCC DBLIB
                                                   *pConn;
          assert(dbproc != NULL);
          pConn = (CTPCC_DBLIB*)dbgetuserdata(dbproc);
          if (pConn != NULL)
                    pConn->SetSqlError( msgno, msgstate, severity, msgtext
);
          return 0;
/* FUNCTION: void UtilStrCpy(char * pDest, char * pSrc, int n)
* PURPOSE:
                    This function copies n characters from string pSrc to pDst
and places a
                              null character at the end of the destination
string.
* ARGUMENTS:
                    char
                                                   *pDest
                                                             destination
string pointer
                                        char
*pSrc
          source string pointer
                                        int
                    number of characters to copy
n
* RETURNS:
                              None
* COMMENTS:
                    Unlike strncpy this function ensures that the result string
is
                                        always null terminated.
```

```
inline static void UtilStrCpy(char * pDest, const BYTE * pSrc, int n)
  strncpy(pDest, (char *)pSrc, n);
  pDest[n] = '\0';
          return;
/* FUNCTION: CTPCC DBLIB ERR::ErrorText
*/
char* CTPCC_DBLIB_ERR::ErrorText(void)
          int i;
          static SERRORMSG errorMsgs[] =
                    { ERR WRONG SP VERSION,
"Wrong version of stored procs on database server"
                    { ERR INVALID CUST,
"Invalid Customer id,name."
          },
                    { ERR_NO_SUCH_ORDER,
                                                              "No orders
found for customer."
                    { ERR_RETRIED_TRANS,
                                                              "Retries
before transaction succeeded."
                                                              }.
          };
          static char szNotFound[] = "Unknown error number.";
          for(i=0; errorMsgs[i].szMsg[0]; i++)
                    if ( m_errno == errorMsgs[i].iError )
                               break;
          if (!errorMsgs[i].szMsg[0])
                    return szNotFound;
          else
                    return errorMsgs[i].szMsg;
// wrapper routine for class constructor
__declspec(dllexport) CTPCC_DBLIB* CTPCC_DBLIB_new(
          LPCSTR szServer,
                                         // name of SQL server
          LPCSTR szUser,
                                                   // user name for login
          LPCSTR szPassword,
                                         // password for login
          LPCSTR szHost,
                                                   // workstation name;
shows up in sp_who; max 30 chars, only first 10 kept by SQL Server
          LPCSTR szDatabase)
                                         // name of database to use
          return new CTPCC DBLIB( szServer, szUser, szPassword, szHost,
szDatabase);
CTPCC DBLIB::CTPCC DBLIB (
          LPCSTR szServer,
                                         // name of SQL server
          LPCSTR szUser,
                                                   // user name for login
          LPCSTR szPassword,
                                         // password for login
          LPCSTR szHost,
                                                   // workstation name;
shows up in sp_who; max 30 chars, only first 10 kept by SQL Server
          LPCSTR szDatabase)
                                         // name of database to use
```

```
LOGINREC
                              *login;
          const BYTE
                               *pData;
          // initialization
          m dbproc = NULL;
          m_DbLibErr = (CDBLIBERR*)NULL;
          m_SqlErr = (CSQLERR*)NULL;
          m MaxRetries = 10;
                                         // how many retries on deadlock
          // increase max number of connections if getting close
          if (dbgetmaxprocs() < (iConnectionCount+5))
                    if ( dbsetmaxprocs(iConnectionCount+10) == FAIL )
ThrowError(CDBLIBERR::eDbSetMaxProcs);
          // allocate a login structure
          login = dblogin();
          if (login == NULL)
                    ThrowError(CDBLIBERR::eLogin);
          InterlockedIncrement(&iConnectionCount);
          // register error and message handler functions
          if (dbprocerrhandle(login, err handler) == NULL)
                    ThrowError(CDBLIBERR::eDbProcHandler);
          if (dbprocmsghandle(login, msg_handler) == NULL)
                    ThrowError(CDBLIBERR::eDbProcHandler);
          DBSETLUSER(login, szUser);
          DBSETLPWD(login, szPassword);
          DBSETLHOST(login, szHost);
          DBSETLPACKET(login, (unsigned short)DEFCLPACKSIZE);
          DBSETLVERSION(login, DBVER60);
                                                             // use dblib
ver 6.0 client behavior
          // set time to wait for login
          if (dbsetlogintime(60) == FAIL)
                    ThrowError(CDBLIBERR::eDbSet);
          // set time to wait for statement execution
          if(dbsettime(180) == FAIL)
                    ThrowError(CDBLIBERR::eDbSet);
          m_dbproc = dbopen(login, szServer);
          // deallocate login structure before checking for success
          dbfreelogin( login );
          if (m dbproc == NULL)
                    ThrowError(CDBLIBERR::eDbOpen);
          // save address of class instance so that the message and error
handler
          // can get to data.
          dbsetuserdata(m_dbproc, (LPVOID)this);
          // Use the the right database
          if (dbuse(m_dbproc, szDatabase) == FAIL)
                    ThrowError(CDBLIBERR::eDbUse);
          dbcmd(m dbproc, "set nocount on ");
                                                                       //
do not return row counts
```

```
dbcmd(m_dbproc, "set XACT_ABORT ON");
rollback transaction on abort
          if (dbsqlexec(m_dbproc) == FAIL)
                                                                            void CTPCC DBLIB::SetSqlError( int /*DBINT*/ msgno, int msgstate, int
                    ThrowError(CDBLIBERR::eDbSqlExec);
                                                                            severity, LPCSTR msgtext)
          DiscardNextResults(2);
                                                                                      if (m SqlErr == NULL)
                                                                                                m SqlErr = new CSQLERR();
          // verify that version of stored procs on server is correct
          dbrpcinit(m dbproc, "tpcc version", 0);
                                                                                      m SqlErr->m msgno = msgno;
                                                                                      m_SqlErr->m_msgstate = msgstate;
          if (dbrpcexec(m dbproc) == FAIL)
                                                                                      m_SqlErr->m_severity = severity;
                    ThrowError(CDBLIBERR::eDbRpcExec);
                                                                                      delete [] m_SqlErr->m_msgtext;
          if (dbresults(m dbproc) != SUCCEED)
                                                                                      if (msgtext != NULL)
                    ThrowError(CDBLIBERR::eDbResults);
                                                                                                m SqlErr->m msgtext = new char[strlen(msgtext)+1];
          if (dbnextrow(m_dbproc) != REG_ROW)
                                                                                                strcpy( m_SqlErr->m_msgtext, msgtext );
                    ThrowError(CDBLIBERR::eDbNextRow);
          char szSrvVersion[16];
          pData=dbdata(m dbproc, 1);
                                                                            void CTPCC DBLIB::ThrowError( CDBLIBERR::ACTION eAction )
          if (pData)
                    UtilStrCpy(szSrvVersion, pData, dbdatlen(m dbproc,
                                                                                      // discard anything still in return buffer
1));
                                                                                      DiscardNextRows(-1);
                                                                                      DiscardNextResults(-1);
          else
                    szSrvVersion[0]=0;
          if (strcmp(szSrvVersion,sVersion))
                                                                                      // check for SQL Server error first; if yes, throw it and ignore any
                    throw new CTPCC DBLIB ERR(
                                                                            DBLib error.
CTPCC_DBLIB_ERR::ERR_WRONG_SP_VERSION );
                                                                                      if (m SqlErr!= NULL)
                                                                                                 CSQLERR
                                                                                                                      *pSqlErr;
          DiscardNextRows(0):
                                                                                                pSqlErr = m SqlErr;
          DiscardNextResults(0);
                                                                                                m SqlErr = NULL;
                                                                                                                     // clear our pointer to instance;
                                                                            catch handler will delete
}
                                                                                                throw pSqlErr;
CTPCC DBLIB::~CTPCC DBLIB( void )
                                                                                      CDBLIBERR
                                                                                                           *pDbLibErr;
          // close db connection and deallocate resources
                                                                                      if (m DbLibErr == NULL)
          dbclose(m dbproc);
                                                                                                 // this case isn't expected to happen, since it means that an
          InterlockedDecrement( &iConnectionCount );
                                                                            error was returned
          if (m DbLibErr != NULL)
                                                                                                // but the error handlers were not called.
                    delete m_DbLibErr;
                                                                                                pDbLibErr = new CDBLIBERR(eAction);
          if (m SqlErr != NULL)
                                                                                      else
                    delete m_SqlErr;
                                                                                                pDbLibErr = m DbLibErr;
}
                                                                                                pDbLibErr->m eAction = eAction;
void CTPCC_DBLIB::SetDbLibError(int severity, int dberr, int oserr, LPCSTR
                                                                                                m DbLibErr = NULL;
                                                                                                                                          // clear our
dberrstr, LPCSTR oserrstr)
                                                                            pointer to instance; catch handler will delete
          delete m DbLibErr;
          m_DbLibErr = new CDBLIBERR(CDBLIBERR::eUnknown,
                                                                                      throw pDbLibErr;
severity, dberr, oserr);
          if (dberrstr!= NULL)
                                                                            // Read and discard rows until no more. Throw an exception if number of rows
                                                                            // match number of rows expected. The row count will be ignored if the
                    m_DbLibErr->m_dberrstr = new char[ strlen(dberrstr)+1
];
                                                                            expected count value
                    strcpy( m_DbLibErr->m_dberrstr, dberrstr );
                                                                            // passed in is negative. A typical use of this routine is to verify that there are
                                                                            no more
                                                                            // rows to be read.
          if (oserrstr!= NULL)
                                                                            void CTPCC_DBLIB::DiscardNextRows(int iExpectedCount)
                    m DbLibErr->m oserrstr = new char[ strlen(oserrstr)+1
                                                                                                           iRowsRead = 0;
                                                                                      int
];
                                                                                      RETCODE
                                                                                                           rc;
                    strcpy( m_DbLibErr->m_oserrstr, oserrstr );
          }
                                                                                      while (TRUE)
```

```
{
                     rc = dbnextrow(m_dbproc);
                     if (rc == NO MORE ROWS)
                               break;
                     if (rc == FAIL)
                               if (iExpectedCount \ge 0)
ThrowError(CDBLIBERR::eDbNextRow);
                               else
                                          break;
                     iRowsRead++;
          if ((iExpectedCount \ge 0) \&\&
                     (iExpectedCount != iRowsRead))
                     ThrowError(CDBLIBERR::eWrongRowCount);
}
// Read and discard results until no more. Throw an exception if number of
result sets read doesn't
// match number expected. The result set count will be ignored if the expected
count value
// passed in is negative. A typical use of this routine is to verify that there are
no more
// result sets to be read.
void CTPCC_DBLIB::DiscardNextResults(int iExpectedCount)
          int
                               iResultsRead = 0;
          RETCODE
                               rc;
          while (TRUE)
                     rc = dbresults(m_dbproc);
                     if (rc == NO MORE RESULTS)
                               break;
                     if (rc == FAIL)
                               if (iExpectedCount >= 0)
ThrowError(CDBLIBERR::eDbResults);
                                          break;
                     DiscardNextRows(-1);
                     iResultsRead++;
          }
          if ((iExpectedCount \geq = 0) &&
                     (iExpectedCount != iResultsRead))
                     ThrowError(CDBLIBERR::eWrongRowCount);
}
void CTPCC_DBLIB::StockLevel()
          int
                                          iTryCount = 0;
          const BYTE
                               *pData;
          ResetError();
          while (TRUE)
                     try
                               dbrpcinit(m_dbproc, "tpcc_stocklevel", 0);
```

```
dbrpcparam(m_dbproc, NULL, 0, SQLINT2,
-1, -1, (BYTE *) &m_txn.StockLevel.w_id);
                                                  // @w_id smallint
                              dbrpcparam(m dbproc, NULL, 0, SQLINT1,
-1, -1, (BYTE *) &m_txn.StockLevel.d_id);
                                                  // @d_id tinyint
                              dbrpcparam(m_dbproc, NULL, 0, SQLINT2,
-1, -1, (BYTE *) &m txn.StockLevel.threshold); // @threshhold smallint
                              if (dbrpcexec(m_dbproc) == FAIL)
ThrowError(CDBLIBERR::eDbRpcExec);
                              if (dbresults(m_dbproc) != SUCCEED)
ThrowError(CDBLIBERR::eDbResults);
                              if (dbnextrow(m_dbproc) != REG_ROW)
ThrowError(CDBLIBERR::eDbNextRow);
                              if (pData=dbdata(m dbproc, 1))
                                       m_txn.StockLevel.low_stock =
*((long *) pData);
                              DiscardNextRows(0);
                              DiscardNextResults(0);
                              m txn.StockLevel.exec status code = eOK;
                              return:
                   catch (CSQLERR *e)
                              if ((e->m_msgno == 1205 ||
                                        (e->m_msgno ==
iErrOleDbProvider &&
                                        strstr(e->m_msgtext,
sErrTimeoutExpired) != NULL)) &&
                                        (++iTryCount <= iMaxRetries))
                                       // hit deadlock; backoff for
increasingly longer period
                                        delete e;
                                       Sleep(10 * iTryCount);
                              else
                                        throw;
                    // while (TRUE)
         //if (iTryCount)
                   throw new
CTPCC_DBLIB_ERR(CTPCC_DBLIB_ERR::ERR_RETRIED_TRANS,
iTryCount);
void CTPCC_DBLIB::NewOrder()
          int
                                       i;
          DBINT
                              commit flag;
         DBDATETIME
                              datetime;
          DBDATEREC
                              daterec;
                                       iTryCount = 0;
          int
          const BYTE
                              *pData;
          ResetError();
          while (TRUE)
```

```
{
                                                                          m txn.NewOrder.OL[i].ol stock = (*(DBSMALLINT *) pData);
                                                                                                                  if(pData=dbdata(m dbproc, 3))
                              dbrpcinit(m_dbproc, "tpcc_neworder", 0);
                                                                          UtilStrCpy(m_txn.NewOrder.OL[i].ol_brand_generic, pData,
                              dbrpcparam(m dbproc, NULL, 0, SQLINT2,
                                                                          dbdatlen(m dbproc, 3));
-1, -1, (BYTE *) &m txn.NewOrder.w id);
                                                                                                                  if(pData=dbdata(m dbproc, 4))
                              dbrpcparam(m_dbproc, NULL, 0, SQLINT1,
                                                                                                                            dbconvert(m dbproc,
-1, -1, (BYTE *) &m txn.NewOrder.d id);
                                                                          SQLNUMERIC, (LPCBYTE)pData, dbdatlen(m dbproc,4).
                                                                                                                            SQLFLT8, (BYTE
                              dbrpcparam(m dbproc, NULL, 0, SQLINT4,
-1, -1, (BYTE *) &m_txn.NewOrder.c_id);
                                                                          *)&m_txn.NewOrder.OL[i].ol_i_price, 8);
                              dbrpcparam(m_dbproc, NULL, 0, SQLINT1,
                                                                                                                  if(pData=dbdata(m_dbproc, 5))
-1, -1, (BYTE *) &m txn.NewOrder.o ol cnt);
                                                                                                                            dbconvert(m_dbproc,
                                                                          SQLNUMERIC, (LPCBYTE)pData, dbdatlen(m_dbproc,5),
                              // check whether any order lines are for a
remote warehouse
                                                                                                                            SQLFLT8, (BYTE
                              m txn.NewOrder.o all local = 1;
                                                                          *)&m txn.NewOrder.OL[i].ol amount, 8);
                              for (i = 0; i < m_txn.NewOrder.o_ol_cnt; i++)
                                                                                                                  m txn.NewOrder.total amount =
                                                                          m txn.NewOrder.total amount + m txn.NewOrder.OL[i].ol amount;
(m_txn.NewOrder.OL[i].ol_supply_w_id != m_txn.NewOrder.w_id)
                                                                                                                  DiscardNextRows(0);
m txn.NewOrder.o all local = 0; // at least one remote warehouse
                                                                                                        // get remaining values for w_tax, d_tax, o_id,
                                                                          c last, c discount, c credit, o entry d, commit flag
                              dbrpcparam(m_dbproc, NULL, 0, SQLINT1,
                                                                                                        if (dbresults(m_dbproc) != SUCCEED)
-1, -1, (BYTE *) &m txn.NewOrder.o all local);
                                                                          ThrowError(CDBLIBERR::eDbResults);
                              for (i = 0; i \le m \text{ txn.NewOrder.o ol cnt}; i++)
                                                                                                        if (dbnextrow(m_dbproc) != REG_ROW)
                                        dbrpcparam(m_dbproc, NULL, 0,
SQLINT4, -1, -1, (BYTE *) &m txn.NewOrder.OL[i].ol i id);
                                                                          ThrowError(CDBLIBERR::eDbNextRow);
                                        dbrpcparam(m_dbproc, NULL, 0.
SQLINT2, -1, -1, (BYTE *) &m txn.NewOrder.OL[i].ol supply w id);
                                                                                                        if (dbnumcols(m dbproc) != 8)
                                        dbrpcparam(m_dbproc, NULL, 0,
SQLINT2, -1, -1, (BYTE *) &m txn.NewOrder.OL[i].ol quantity);
                                                                          ThrowError(CDBLIBERR::eWrongNumCols);
                                                                                                        if (pData=dbdata(m_dbproc, 1))
                              if(dbrpcexec(m dbproc) == FAIL)
ThrowError(CDBLIBERR::eDbRpcExec);
                                                                                                                   dbconvert(m dbproc,
                                                                          SQLNUMERIC, (LPCBYTE)pData, dbdatlen(m_dbproc,1), SQLFLT8, (BYTE
                                                                          *)&m txn.NewOrder.w tax, 8);
                                                                                                        if (pData=dbdata(m_dbproc, 2))
                              // Get order line results
                              m txn.NewOrder.total amount = 0;
                                                                                                                   dbconvert(m dbproc,
                              for (i = 0; i < m_txn.NewOrder.o_ol_cnt; i++)
                                                                          SQLNUMERIC, (LPCBYTE)pData, dbdatlen(m_dbproc,2), SQLFLT8, (BYTE
                                                                          *)&m txn.NewOrder.d tax, 8);
                                                                                                        if (pData=dbdata(m_dbproc, 3))
                                        if (dbresults(m_dbproc) !=
SUCCEED)
                                                                                                                  m txn.NewOrder.o id =
                                                                          (*(DBINT *) pData);
ThrowError(CDBLIBERR::eDbResults);
                                                                                                        if (pData=dbdata(m dbproc, 4))
                                        if (dbnumcols(m dbproc) != 5)
                                                                          UtilStrCpy(m txn.NewOrder.c last, pData, dbdatlen(m dbproc, 4));
                                                                                                        if (pData=dbdata(m dbproc, 5))
ThrowError(CDBLIBERR::eWrongNumCols);
                                                                                                                  dbconvert(m dbproc,
                                                                          SQLNUMERIC, (LPCBYTE)pData, dbdatlen(m dbproc,5), SQLFLT8, (BYTE
                                        if (dbnextrow(m_dbproc) !=
                                                                          *)&m_txn.NewOrder.c_discount, 8);
REG ROW)
                                                                                                        if (pData=dbdata(m dbproc, 6))
ThrowError(CDBLIBERR::eDbNextRow);
                                                                          UtilStrCpy(m_txn.NewOrder.c_credit, pData, dbdatlen(m_dbproc, 6));
                                                                                                        if (pData=dbdata(m_dbproc, 7))
                                        if(pData=dbdata(m_dbproc, 1))
                                                                                                                  datetime = *((DBDATETIME *)
UtilStrCpy(m_txn.NewOrder.OL[i].ol_i_name, pData, dbdatlen(m_dbproc, 1));
                                                                          pData);
                                        if(pData=dbdata(m dbproc, 2))
```

```
iTryCount = 0;
                                        dbdatecrack(m_dbproc, &daterec,
                                                                                     int
&datetime);
                                                                                     const BYTE
                                                                                                         *pData;
                                        m txn.NewOrder.o entry d.year
= daterec.year;
                                                                                     ResetError();
                                        m_txn.NewOrder.o_entry_d.month
= daterec.month;
                                                                                     while (TRUE)
                                        m txn.NewOrder.o entry d.day
= daterec.day;
                                                                                               try
                                        m_txn.NewOrder.o_entry_d.hour
                                                                                                         dbrpcinit(m dbproc, "tpcc payment", 0);
= daterec.hour;
                                                                                                         dbrpcparam(m_dbproc, NULL, 0, SQLINT2,
m_txn.NewOrder.o_entry_d.minute = daterec.minute;
                                                                           -1, -1, (BYTE *) &m txn.Payment.w id);
m_txn.NewOrder.o_entry_d.second = daterec.second;
                                                                                                         dbrpcparam(m_dbproc, NULL, 0, SQLINT2,
                                                                           -1, -1, (BYTE *) &m_txn.Payment.c_w_id);
                              if (pData=dbdata(m dbproc, 8))
                                                                                                         dbrpcparam(m_dbproc, NULL, 0, SQLFLT8,
                                        commit_flag = (*(DBTINYINT *)
                                                                           -1, -1, (BYTE *) &m txn.Payment.h amount);
pData);
                                                                                                         dbrpcparam(m_dbproc, NULL, 0, SQLINT1,
                                                                           -1, -1, (BYTE *) &m_txn.Payment.d_id);
                              DiscardNextRows(0);
                                                                                                         dbrpcparam(m dbproc, NULL, 0, SQLINT1,
                                                                           -1, -1, (BYTE *) &m_txn.Payment.c_d_id);
                              DiscardNextResults(0);
                                                                                                         dbrpcparam(m dbproc, NULL, 0, SQLINT4,
                                                                           -1, -1, (BYTE *) &m txn.Payment.c id);
                              if (commit flag == 1)
                                                                                                         // if customer id is zero, then payment is by
                                        m txn.NewOrder.total amount *=
                                                                          name
((1 + m txn.NewOrder.w tax + m txn.NewOrder.d tax) * (1 -
                                                                                                         if (m txn.Payment.c id == 0)
m_txn.NewOrder.c_discount));
                                                                                                                   dbrpcparam(m_dbproc, NULL, 0,
                                                                          SQLCHAR, -1, strlen(m_txn.Payment.c_last), (unsigned char
m_txn.NewOrder.exec_status_code = eOK;
                                                                           *)m_txn.Payment.c_last);
                              else
                                                                                                         if (dbrpcexec(m_dbproc) == FAIL)
m txn.NewOrder.exec status code = eInvalidItem;
                                                                          ThrowError(CDBLIBERR::eDbRpcExec);
                                                                                                         if (dbresults(m dbproc) != SUCCEED)
                              return;
                    catch (CSQLERR *e)
                                                                          ThrowError(CDBLIBERR::eDbResults);
                                                                                                         if (dbnextrow(m_dbproc) != REG_ROW)
                              if ((e->m_msgno == 1205 ||
                                        (e->m msgno ==
iErrOleDbProvider &&
                                                                          ThrowError(CDBLIBERR::eDbNextRow);
                                        strstr(e->m msgtext,
                                                                                                         if (dbnumcols(m_dbproc) != 27)
sErrTimeoutExpired) != NULL)) &&
                                        (++iTryCount <= iMaxRetries))
                                                                          ThrowError(CDBLIBERR::eWrongNumCols);
                                        // hit deadlock; backoff for
increasingly longer period
                                                                                                         if (pData=dbdata(m dbproc, 1))
                                                                                                                   m_{txn.Payment.c_id} = *((DBINT)
                                        delete e:
                                        Sleep(10 * iTryCount);
                                                                           *) pData);
                                                                                                         if (pData=dbdata(m_dbproc, 2))
                              else
                                                                                                                   UtilStrCpy(m txn.Payment.c last,
                                                                          pData, dbdatlen(m_dbproc, 2));
                                        throw;
                                                                                                         if (pData=dbdata(m_dbproc, 3))
                    // while (TRUE)
                                                                                                                   datetime = *((DBDATETIME *)
                                                                          pData);
          if (iTryCount)
                    throw new
                                                                                                                   dbdatecrack(m_dbproc, &daterec,
CTPCC DBLIB ERR(CTPCC DBLIB ERR::ERR RETRIED TRANS,
                                                                          &datetime);
iTryCount);
                                                                                                                   m_txn.Payment.h_date.year =
                                                                          daterec.year;
                                                                                                                   m_txn.Payment.h_date.month =
                                                                          daterec.month;
void CTPCC_DBLIB::Payment()
                                                                                                                   m_txn.Payment.h_date.day =
                                                                          daterec.day;
          DBDATETIME
                              datetime;
                                                                                                                   m txn.Payment.h date.hour =
          DBDATEREC
                              daterec;
                                                                          daterec.hour;
```

```
m_txn.Payment.h_date.minute =
                                                                                                                     m txn.Payment.c since.month =
daterec.minute;
                                                                            daterec.month;
                                         m txn.Payment.h date.second =
                                                                                                                     m txn.Payment.c since.day =
daterec.second;
                                                                            daterec.day;
                                                                                                                     m_txn.Payment.c_since.hour =
                               if (pData=dbdata(m dbproc, 4))
                                                                            daterec.hour;
                                                                                                                     m txn.Payment.c since.minute =
UtilStrCpy(m txn.Payment.w street 1, pData, dbdatlen(m dbproc, 4));
                                                                            daterec.minute;
                               if (pData=dbdata(m dbproc, 5))
                                                                                                                     m_txn.Payment.c_since.second =
                                                                            daterec.second:
UtilStrCpy(m_txn.Payment.w_street_2, pData, dbdatlen(m_dbproc, 5));
                               if (pData=dbdata(m_dbproc, 6))
                                                                                                           if(pData=dbdata(m_dbproc, 23))
                                                                            UtilStrCpy(m_txn.Payment.c_credit, pData, dbdatlen(m_dbproc, 23));
UtilStrCpy(m_txn.Payment.w_city, pData, dbdatlen(m_dbproc, 6));
                               if (pData=dbdata(m dbproc, 7))
                                                                                                           if(pData=dbdata(m dbproc, 24))
                                                                                                                     dbconvert(m dbproc,
                                                                            SOLNUMERIC, (LPCBYTE)pData, dbdatlen(m_dbproc,24), SQLFLT8,
UtilStrCpy(m txn.Payment.w state, pData, dbdatlen(m dbproc, 7));
                               if (pData=dbdata(m_dbproc, 8))
                                                                            (BYTE *)&m_txn.Payment.c_credit_lim, 8);
                                                                                                           if(pData=dbdata(m_dbproc, 25))
                                         UtilStrCpy(m_txn.Payment.w_zip,
pData, dbdatlen(m dbproc, 8));
                                                                                                                     dbconvert(m dbproc,
                                                                            SQLNUMERIC, (LPCBYTE)pData, dbdatlen(m_dbproc,25), SQLFLT8,
                               if (pData=dbdata(m_dbproc, 9))
                                                                            (BYTE *)&m txn.Payment.c discount, 8);
UtilStrCpy(m_txn.Payment.d_street_1, pData, dbdatlen(m_dbproc, 9));
                                                                                                           if(pData=dbdata(m dbproc, 26))
                               if (pData=dbdata(m dbproc, 10))
                                                                                                                     dbconvert(m dbproc,
                                                                            SQLNUMERIC, (LPCBYTE)pData, dbdatlen(m_dbproc,26), SQLFLT8,
                                                                            (BYTE *)&m_txn.Payment.c_balance, 8);
UtilStrCpy(m txn.Payment.d street 2, pData, dbdatlen(m dbproc, 10));
                               if (pData=dbdata(m dbproc, 11))
                                                                                                           if(pData=dbdata(m dbproc, 27))
                                         UtilStrCpy(m_txn.Payment.d_city.
                                                                                                                     UtilStrCpy(m_txn.Payment.c_data,
pData, dbdatlen(m dbproc, 11));
                                                                            pData, dbdatlen(m dbproc, 27));
                               if (pData=dbdata(m_dbproc, 12))
                                                                                                           DiscardNextRows(0);
UtilStrCpy(m_txn.Payment.d_state, pData, dbdatlen(m_dbproc, 12));
                                                                                                           DiscardNextResults(0);
                               if (pData=dbdata(m dbproc, 13))
                                         UtilStrCpy(m txn.Payment.d zip,
                                                                                                           if(m_txn.Payment.c_id == 0)
pData, dbdatlen(m_dbproc, 13));
                               if (pData=dbdata(m dbproc, 14))
                                                                                                                     throw new CTPCC DBLIB ERR(
                                                                            CTPCC DBLIB_ERR::ERR_INVALID_CUST );
                                         UtilStrCpy(m_txn.Payment.c_first,
pData, dbdatlen(m dbproc, 14));
                               if (pData=dbdata(m_dbproc, 15))
                                                                                                                     m_txn.Payment.exec_status_code
                                                                            = eOK:
UtilStrCpy(m txn.Payment.c middle, pData, dbdatlen(m dbproc, 15));
                               if (pData=dbdata(m_dbproc, 16))
                                                                                                           return:
                                                                                                catch (CSQLERR *e)
UtilStrCpy(m_txn.Payment.c_street_1, pData, dbdatlen(m_dbproc, 16));
                               if (pData=dbdata(m dbproc, 17))
                                                                                                           if ((e->m_msgno == 1205 ||
UtilStrCpy(m txn.Payment.c street 2, pData, dbdatlen(m dbproc, 17));
                                                                                                                     (e->m msgno ==
                               if (pData=dbdata(m dbproc, 18))
                                                                            iErrOleDbProvider &&
                                         UtilStrCpy(m_txn.Payment.c_city,
                                                                                                                     strstr(e->m_msgtext,
                                                                            sErrTimeoutExpired) != NULL)) &&
pData, dbdatlen(m_dbproc, 18));
                               if (pData=dbdata(m_dbproc, 19))
                                                                                                                     (++iTryCount <= iMaxRetries))
UtilStrCpy(m_txn.Payment.c_state, pData, dbdatlen(m_dbproc, 19));
                                                                                                                     // hit deadlock; backoff for
                               if (pData=dbdata(m dbproc, 20))
                                                                            increasingly longer period
                                         UtilStrCpy(m_txn.Payment.c_zip,
                                                                                                                     delete e;
                                                                                                                     Sleep(10 * iTryCount);
pData, dbdatlen(m_dbproc, 20));
                               if (pData=dbdata(m dbproc, 21))
                                                                                                           else
UtilStrCpy(m txn.Payment.c phone, pData, dbdatlen(m dbproc, 21));
                                                                                                                     throw;
                               if (pData=dbdata(m_dbproc, 22))
                                                                                                // while (TRUE)
                                         datetime = *((DBDATETIME *)
pData);
                                                                                      if (iTryCount)
                                         dbdatecrack(m_dbproc, &daterec,
                                                                                                throw new
                                                                            CTPCC_DBLIB_ERR(CTPCC_DBLIB_ERR::ERR_RETRIED_TRANS,
&datetime);
                                         m txn.Payment.c since.year =
                                                                            iTryCount);
daterec.year;
```

```
if(pData=dbdata(m_dbproc, 2))
                                                                          m txn.OrderStatus.OL[i].ol i id = (*(DBINT *) pData);
void CTPCC DBLIB::OrderStatus()
                                                                                                                   if(pData=dbdata(m_dbproc, 3))
          int
                                         i;
          DBDATETIME
                              datetime;
                                                                           m txn.OrderStatus.OL[i].ol quantity = (*(DBSMALLINT *) pData);
          DBDATEREC
                              daterec;
                                                                                                                   if(pData=dbdata(m dbproc, 4))
                                                                                                                              dbconvert(m dbproc,
          int
                                        iTryCount = 0;
                                                                           SQLNUMERIC, (LPCBYTE)pData, dbdatlen(m dbproc,4),
          RETCODE
          const BYTE
                              *pData;
                                                                           SQLFLT8, (BYTE *)&m_txn.OrderStatus.OL[i].ol_amount, 8);
                                                                                                                   if(pData=dbdata(m_dbproc, 5))
          ResetError();
                                                                                                                              datetime =
                                                                           *((DBDATETIME *) pData);
          while (TRUE)
                                                                                                                              dbdatecrack(m_dbproc,
                                                                           &daterec, &datetime);
                    try
                              dbrpcinit(m_dbproc, "tpcc_orderstatus", 0);
                                                                           m_txn.OrderStatus.OL[i].ol_delivery_d.year = daterec.year;
                              dbrpcparam(m_dbproc, NULL, 0, SQLINT2,
                                                                           m_txn.OrderStatus.OL[i].ol_delivery_d.month = daterec.month;
-1, -1, (BYTE *) &m txn.OrderStatus.w id);
                              dbrpcparam(m_dbproc, NULL, 0, SQLINT1,
                                                                           m_txn.OrderStatus.OL[i].ol_delivery_d.day = daterec.day;
-1, -1, (BYTE *) &m txn.OrderStatus.d id);
                              dbrpcparam(m_dbproc, NULL, 0, SQLINT4,
                                                                           m_txn.OrderStatus.OL[i].ol_delivery_d.hour = daterec.hour;
-1, -1, (BYTE *) &m_txn.OrderStatus.c_id);
                                                                           m txn.OrderStatus.OL[i].ol delivery d.minute = daterec.minute;
                              // if customer id is zero, then order status is by
name
                                                                           m txn.OrderStatus.OL[i].ol delivery d.second = daterec.second;
                              if (m txn.OrderStatus.c id == 0)
                                         dbrpcparam(m dbproc, NULL, 0,
                                                                                                                   i++;
SQLCHAR, -1, strlen(m_txn.OrderStatus.c_last), (unsigned char
*)m_txn.OrderStatus.c_last);
                                                                                                         m_txn.OrderStatus.o_ol_cnt = i;
                              if(dbrpcexec(m dbproc) == FAIL)
                                                                                                         if (dbresults(m dbproc) != SUCCEED)
ThrowError(CDBLIBERR::eDbRpcExec);
                                                                           ThrowError(CDBLIBERR::eDbResults);
                                                                                                         if (dbnextrow(m dbproc) != REG ROW)
                              // Get order lines
                              if (dbresults(m dbproc) != SUCCEED)
                                                                           ThrowError(CDBLIBERR::eDbNextRow);
                                         if ((m DbLibErr == NULL) &&
(m_SqlErr == NULL))
                                                                                                         if (dbnumcols(m_dbproc) != 8)
                                                   throw new
CTPCC_DBLIB_ERR( CTPCC_DBLIB_ERR::ERR_NO_SUCH_ORDER );
                                                                           ThrowError(CDBLIBERR::eWrongNumCols);
                                                                                                         if(pData=dbdata(m dbproc, 1))
ThrowError(CDBLIBERR::eDbResults);
                                                                                                                   m_txn.OrderStatus.c_id =
                                                                           (*(DBINT *) pData);
                                                                                                         if(pData=dbdata(m_dbproc, 2))
                              if (dbnumcols(m dbproc) != 5)
                                                                           UtilStrCpy(m_txn.OrderStatus.c_last, pData, dbdatlen(m_dbproc,2));
ThrowError(CDBLIBERR::eWrongNumCols);
                                                                                                         if(pData=dbdata(m_dbproc, 3))
                              i = 0;
                                                                           UtilStrCpy(m_txn.OrderStatus.c_first, pData, dbdatlen(m_dbproc,3));
                                                                                                         if(pData=dbdata(m dbproc, 4))
                              while (TRUE)
                                         rc = dbnextrow(m dbproc);
                                                                           UtilStrCpy(m txn.OrderStatus.c middle, pData, dbdatlen(m dbproc, 4));
                                         if (rc == NO\_MORE\_ROWS)
                                                   break;
                                                                                                         if(pData=dbdata(m dbproc, 5))
                                         if (rc != REG ROW)
                                                                                                                   datetime = *((DBDATETIME *)
ThrowError(CDBLIBERR::eDbNextRow);
                                                                           pData);
                                                                                                                   dbdatecrack(m_dbproc, &daterec,
                                         if(pData=dbdata(m dbproc, 1))
                                                                           &datetime);
                                                                                                                   m_txn.OrderStatus.o_entry_d.year
m\_txn.OrderStatus.OL[i].ol\_supply\_w\_id = (*(DBSMALLINT *) pData);
                                                                           = daterec.year;
```

```
m txn.OrderStatus.o entry d.month = daterec.month;
                                                                                    int
                                        m txn.OrderStatus.o entry d.day
                                                                                                                   iTryCount = 0;
                                                                                    int
= daterec.day;
                                                                                     const BYTE
                                                                                                         *pData;
                                        m_txn.OrderStatus.o_entry_d.hour
 = daterec.hour;
                                                                                    ResetError();
m_txn.OrderStatus.o_entry_d.minute = daterec.minute;
                                                                                    while (TRUE)
m_txn.OrderStatus.o_entry_d.second = daterec.second;
                                                                                              try
                              if(pData=dbdata(m dbproc, 6))
                                                                                                         dbrpcinit(m_dbproc, "tpcc_delivery", 0);
                                        m_txn.OrderStatus.o_carrier_id =
                                                                                                         dbrpcparam(m_dbproc, NULL, 0, SQLINT2,
(*(DBSMALLINT *) pData);
                                                                           -1, -1, (BYTE *) &m_txn.Delivery.w_id);
                              if(pData=dbdata(m_dbproc, 7))
                                                                                                         dbrpcparam(m_dbproc, NULL, 0, SQLINT1,
                                        dbconvert(m dbproc,
                                                                          -1, -1, (BYTE *) &m_txn.Delivery.o_carrier_id);
SQLNUMERIC, (LPCBYTE)pData, dbdatlen(m_dbproc,7),
                                                             SOLFLT8.
                                                                                                         if (dbrpcexec(m_dbproc) == FAIL)
(BYTE *)&m txn.OrderStatus.c balance, 8);
                                                                          ThrowError(CDBLIBERR::eDbRpcExec);
                              if(pData=dbdata(m_dbproc, 8))
                                        m txn.OrderStatus.o id =
(*(DBINT *) pData);
                                                                                                         if (dbresults(m dbproc) != SUCCEED)
                              DiscardNextRows(0);
                                                                          ThrowError(CDBLIBERR::eDbResults);
                              DiscardNextResults(0);
                                                                                                         if (dbnextrow(m dbproc) != REG ROW)
                              if (m txn.OrderStatus.o ol cnt == 0)
                                                                          ThrowError(CDBLIBERR::eDbNextRow);
                                        throw new CTPCC DBLIB ERR(
CTPCC DBLIB ERR::ERR NO SUCH ORDER);
                                                                                                         if (dbnumcols(m dbproc)!= 10)
                              else if (m_txn.OrderStatus.c_id == 0 &&
m_txn.OrderStatus.c_last[0] == 0
                                                                          ThrowError(CDBLIBERR::eWrongNumCols);
                                        throw new CTPCC DBLIB ERR(
CTPCC_DBLIB_ERR::ERR_INVALID_CUST );
                                                                                                         for (i=0; i<10; i++)
                                                                                                                   if (pData = dbdata(m_dbproc, i+1))
                                                                                                                             m txn.Delivery.o id[i]
m txn.OrderStatus.exec status code = eOK;
                                                                          = *((DBINT *)pData);
                              return:
                    catch (CSQLERR *e)
                                                                                                         DiscardNextRows(0);
                                                                                                         DiscardNextResults(0);
                              if ((e->m_msgno == 1205 ||
                                                                                                         m txn.Delivery.exec status code = eOK;
                                        (e->m msgno =
iErrOleDbProvider &&
                                                                                                        return;
                                        strstr(e->m_msgtext,
sErrTimeoutExpired) != NULL)) &&
                                                                                              catch (CSQLERR *e)
                                        (++iTryCount <= iMaxRetries))
                                                                                                         if ((e->m msgno == 1205 ||
                                        // hit deadlock; backoff for
                                                                                                                   (e->m_msgno ==
                                                                          iErrOleDbProvider &&
increasingly longer period
                                        delete e;
                                                                                                                   strstr(e->m_msgtext,
                                        Sleep(10 * iTryCount);
                                                                          sErrTimeoutExpired) != NULL)) &&
                                                                                                                   (++iTryCount <= iMaxRetries))
                              else
                                                                                                                   // hit deadlock; backoff for
                                        throw;
                                                                          increasingly longer period
                    // while (TRUE)
                                                                                                                   delete e;
                                                                                                                   Sleep(10 * iTryCount);
//
          if (iTryCount)
                    throw new
                                                                                                         else
CTPCC_DBLIB_ERR(CTPCC_DBLIB_ERR::ERR_RETRIED_TRANS,
                                                                                                                   throw;
iTryCount);
                                                                                              // while (TRUE)
                                                                          //
                                                                                    if (iTryCount)
void CTPCC DBLIB::Delivery()
```

```
FILE:
                               TPCC DBLIB.H
                                         Microsoft TPC-C Kit Ver.
4.20.000
                                         Copyright Microsoft, 1999
         All Rights Reserved
                                         Version 4.10.000 audited by
Richard Gimarc, Performance Metrics, 3/17/99
          PURPOSE:
                               Header file for TPC-C txn class
implementation.
  Change history:
                    4.20.000 - updated rev number to match kit
#pragma once
#ifndef PDBPROCESS
#define DBPROCESS void // dbprocess structure type
typedef DBPROCESS * PDBPROCESS;
// need to declare functions for import, unless define has already been created
// by the DLL's .cpp module for export.
#ifndef DllDecl
#define DllDecl __declspec( dllimport )
#endif
class CSQLERR: public CBaseErr
          public:
                    CSQLERR(void)
                               m_msgno = 0;
                               m_msgstate = 0;
                               m_severity = 0;
                               m msgtext = NULL;
                     };
                     ~CSQLERR()
                               delete [] m_msgtext;
                    };
```

```
int
                                          m_msgno;
                                          m msgstate;
                    int
                                          m_severity;
                    int
                    char *m_msgtext;
                    int ErrorType() {return ERR_TYPE_SQL;};
                    int ErrorNum() {return m msgno;};
                    char *ErrorText() {return m_msgtext;};
class CDBLIBERR: public CBaseErr
          public:
                    enum ACTION
                               eNone,
                               eUnknown,
                                                                         //
                               eLogin,
error from dblogin
                               eDbOpen,
                                                               // error from
dbopen
                               eDbUse,
                                                                         //
error from dbuse
                               eDbSqlExec,
                                                                         //
error from dbsqlexec
                               eDbSet,
                                                                         //
error from one of the dbset* routines
                                                                         //
                               eDbNextRow,
error from dbnextrow
                               eWrongRowCount,
                                                               // more or
less rows returned than expected
                               eWrongNumCols,
                                                               // more or
less columns returned than expected
                               eDbResults,
                                                                         //
error from dbresults
                               eDbRpcExec,
                                                                         //
error from dbrpcexec
                               eDbSetMaxProcs,
                                                               // error from
dbsetmaxprocs
                               eDbProcHandler
                                                               // error from
either dbprocerrhandle or dbprocmsghandle
                    CDBLIBERR(ACTION eAction, int severity = 0, int
dberror = 0, int oserr = 0)
                               m_eAction = eAction;
                               m severity = severity;
                               m_dberror = dberror;
                               m oserr = oserr;
                               m dberrstr = NULL;
                               m_oserrstr = NULL;
                    };
                    ~CDBLIBERR()
                     {
                               delete [] m_dberrstr;
                               delete [] m_oserrstr;
                    };
                     ACTION m_eAction;
                    int
                                          m_severity;
                                          m dberror;
                    int
                    int
                                          m_oserr;
                    char *m dberrstr;
```

```
~CTPCC_DBLIB(void);
                   char *m oserrstr;
                    int ErrorType() {return ERR TYPE DBLIB;};
                                                                                             inline PNEW ORDER DATA
                    int ErrorNum() {return m_dberror;};
                                                                         BuffAddr_NewOrder()
                                                                                                                 { return &m_txn.NewOrder; };
                   char *ErrorText() {return m_dberrstr;};
                                                                                             inline PPAYMENT DATA
                                                                         BuffAddr Payment()
                                                                                                       { return &m txn.Payment; };
                                                                                             inline PDELIVERY_DATA
};
                                                                         BuffAddr_Delivery()
                                                                                                       { return &m txn.Delivery; };
                                                                                             in line\ PSTOCK\_LEVEL\_DATA\ BuffAddr\_StockLevel()
class CTPCC DBLIB ERR: public CBaseErr
                                                                         { return &m_txn.StockLevel; };
                                                                                             inline PORDER_STATUS_DATA
         public:
                   enum CTPCC_DBLIB_ERRS
                                                                         BuffAddr_OrderStatus()
                                                                                                       { return &m_txn.OrderStatus; };
                             ERR_WRONG_SP_VERSION = 1,
                                                                     //
                                                                                             void NewOrder
                                                                                                                           ();
                                                                                             void Payment
"Wrong version of stored procs on database server"
                                                                                                                           ();
                             ERR INVALID CUST,
                                                                                             void Delivery
                                                                                                                           ();
// "Invalid Customer id,name."
                                                                                             void StockLevel
                                                                                                                           ();
                             ERR_NO_SUCH_ORDER,
                                                                                             void OrderStatus
                                                                                                                 ();
// "No orders found for customer."
                             ERR RETRIED TRANS,
// "Retries before transaction succeeded."
                                                                                             // these are public because they must be called from the
                                                                         dblib err handler and msg hangler
                   }:
                                                                                             // outside of the class
                   CTPCC DBLIB ERR( int iErr ) { m errno = iErr;
                                                                                             void SetDbLibError(int severity, int dberr, int oserr,
m_iTryCount = 0; };
                                                                         LPCSTR dberrstr, LPCSTR oserrstr);
                                                                                             void SetSqlError( int msgno, int msgstate, int severity,
                    CTPCC DBLIB ERR( int iErr, int iTryCount ) {
                                                                         LPCSTR msgtext );
m_errno = iErr; m_iTryCount = iTryCount; };
                                                                         };
                    int
                                        m errno:
                   int
                                       m iTryCount;
                                                                         extern "C" DllDecl CTPCC DBLIB* CTPCC DBLIB new
                                                                                   ( LPCSTR szServer, LPCSTR szUser, LPCSTR szPassword,
                   int ErrorType() {return ERR TYPE TPCC DBLIB;};
                                                                         LPCSTR szHost, LPCSTR szDatabase );
                   int ErrorNum() {return m errno;};
                                                                         typedef CTPCC_DBLIB* (TYPE_CTPCC_DBLIB)(LPCSTR, LPCSTR,
                                                                         LPCSTR, LPCSTR, LPCSTR);
                   char *ErrorText();
};
class DllDecl CTPCC_DBLIB: public CTPCC_BASE
                                                                         tpcc_enc.cpp
          private:
                                                                         // ctpcc_enc.cpp: implementation of the CTPCC_ENCINA class.
                    // declare variables and private functions here...
                   PDBPROCESS
                                        m dbproc;
                                                                         *m DbLibErr;
                   CDBLIBERR
not allocated until needed (maybe never)
                                                                         #include <windows.h>
                                        *m_SqlErr;
                    CSOLERR
                                                                         #include cess.h>
// not allocated until needed (maybe never)
                                                                         #include <stdio.h>
                                                 m MaxRetries;
                                                                         #include <stdarg.h>
// retry count on deadlock
                                                                         #include <malloc.h>
                                                                         #include <stdlib.h>
                    void DiscardNextRows(int iExpectedCount);
                                                                         #include <string.h>
                    void DiscardNextResults(int iExpectedCount);
                                                                         #include <time.h>
                    void ThrowError( CDBLIBERR::ACTION eAction );
                                                                         #include <sys\timeb.h>
                    void ResetError();
                                                                         #include <io.h>
                   union
                                                                         #ifdef ICECAP
                                                                         #include <icapexp.h>
                             NEW ORDER DATA
                                                                         #endif
NewOrder;
                             PAYMENT_DATA
                                                           Payment;
                                                                         // need to declare functions for export
                             DELIVERY DATA
                                                           Delivery;
                                                                         #define DllDecl __declspec( dllexport )
                             STOCK LEVEL DATA
                                                           StockLevel;
                             ORDER_STATUS_DATA
                                                           OrderStatus;
                                                                         #include "..\..\common\src\trans.h"
                                                                                                               //tpckit transaction header contains
                                                                         definitions of structures specific to TPC-C
                                                                         #include "..\..\common\src\error.h"
         public:
                                                                         #include "..\..\common\src\txn_base.h"
                   CTPCC_DBLIB(LPCSTR szServer, LPCSTR szUser,
                                                                         #include "tpcc_enc.h"
LPCSTR szPassword, LPCSTR szHost, LPCSTR szDatabase );
                                                                         #include "..\include\tpcc_type.h"
```

```
#include "mon client.h"
#include "client utils.h"
static CRITICAL SECTION
                              TpCriticalSection;
extern "C" char *errFile;
BOOL APIENTRY DllMain(HANDLE hModule, DWORD ul reason for call,
LPVOID lpReserved)
    switch(ul_reason_for_call)
        case DLL PROCESS ATTACH:
             DisableThreadLibraryCalls(hModule);
             InitializeCriticalSection(&TpCriticalSection);
             break;
        case DLL PROCESS DETACH:
             DeleteCriticalSection(&TpCriticalSection);
             break:
        default:
             /* nothing */;
    return TRUE;
// wrapper routine for class constructor
  _declspec(dllexport) CTPCC_ENCINA* CTPCC_ENCINA new()
          return new CTPCC ENCINA();
// wrapper routine for enroll client
  declspec(dllexport) CTPCC ENCINA* CTPCC ENCINA post init()
          enroll client();
          return NULL;
// constructor and destructor
CTPCC ENCINA::CTPCC ENCINA()
                   Add initialization of ENCINA Structures if any
         m_txn = (ENC_DATA *)malloc(sizeof(ENC_DATA));
         if (m txn == NULL)
                   throw new CENCERR(ERR TYPE MEMORY,
ERR FATAL_LEVEL);
CTPCC ENCINA::~CTPCC ENCINA()
          // free the data structure allocated with tpalloc
          free((char *)m txn);
void CTPCC_ENCINA::NewOrder()
         // question: if we need to prepare the data?
  if (send_new_order(sizeof(ENC_DATA), (unsigned char *)m_txn) ==
TRPC ERROR)
                    throw new CENCERR(TRPC_ERROR);
  if ( m txn->ErrorType != ERR SUCCESS )
                   throw new CENCERR( m_txn->ErrorType, m_txn->error
);
```

```
void CTPCC ENCINA::Payment()
  if (send_payment(sizeof(ENC_DATA), (unsigned char *)m_txn) ==
TRPC ERROR)
                    throw new CENCERR(TRPC ERROR);
  if ( m_txn->ErrorType != ERR_SUCCESS )
                    throw new CENCERR( m txn->ErrorType, m txn->error
void CTPCC_ENCINA::Delivery()
          // Note: Delivery txn code in the tuxedo server does not implement
logging of the delivery
         //
               txn results, so cannot be used as is to run an auditable TPC-C
result. For that
               reason, delivery txns should not be done via Tuxedo.
          //
               The code is included for completeness.
          //m_txn->u.Delivery.exec_status_code = eDeliveryFailed;
          //return:
          // Note: If we use the delivery thread in tpcc.dll, it is not possible to
get to this
               point for delivery txns. But if we use Encina delivery server,
the code is
               needed. It is suggested using the delivery thread in tpcc.dll
since it is
               convenient and provides best performance.
          GetLocalTime(&m_txn->u.Delivery.queue_time);
  if (send_delivery(sizeof(ENC_DATA), (unsigned char *)m_txn) ==
TRPC ERROR)
                    m_txn->u.Delivery.exec_status_code = eDeliveryFailed;
  else
                    m_txn->u.Delivery.exec_status_code = eOK;
void CTPCC_ENCINA::StockLevel()
  if (send stock level(sizeof(ENC DATA), (unsigned char *)m txn) ==
TRPC_ERROR)
                    throw new CENCERR(TRPC ERROR);
  if ( m_txn->ErrorType != ERR_SUCCESS )
                    throw new CENCERR( m txn->ErrorType, m txn->error
void CTPCC ENCINA::OrderStatus()
  if (send order status(sizeof(ENC DATA), (unsigned char *)m txn) ==
TRPC ERROR)
                    throw new CENCERR(TRPC ERROR);
  if ( m txn->ErrorType != ERR SUCCESS )
                    throw new CENCERR( m_txn->ErrorType, m_txn->error
char *CENCERR::ErrorText()
          if (m iErrorType == TRPC ERROR)
```

```
inline PNEW ORDER DATA
                    sprintf( m szErrorText, "Error: ENCINA TRPC error
                                                                          BuffAddr NewOrder()
                                                                                                                   { return &m txn->u.NewOrder;
(see log file %s for details)", errFile);
                                                                                               inline PPAYMENT DATA
                                                                                                         { return &m_txn->u.Payment;
                                                                          BuffAddr_Payment()
          else
                    sprintf( m szErrorText, "Error: Class %d, error # %d",
                                                                                               inline PDELIVERY DATA
m iErrorType, m iError);
                                                                          BuffAddr Delivery()
                                                                                                         { return &m_txn->u.Delivery; };
                                                                                               inline PSTOCK_LEVEL_DATA BuffAddr_StockLevel()
          return m szErrorText;
};
                                                                           { return &m_txn->u.StockLevel; };
                                                                                               inline PORDER STATUS DATA
                                                                           BuffAddr_OrderStatus()
                                                                                                         { return &m_txn->u.OrderStatus; };
tpcc enc.h
                                                                                               void NewOrder
                                                                                                                             ();
                                                                                               void Payment
                                                                                                                             ();
          FILE:
                              TPCC ENCINA.H
                                                                                               void Delivery
                                                                                                                             ();
                                         Microsoft TPC-C Kit Ver.
                                                                                               void StockLevel
                                                                                                                             ();
4.10.000
                                                                                               void OrderStatus
                                                                                                                   ();
                                        not yet audited
          PURPOSE:
                              Header file for TPC-C Encina class
implementation.
                                                                          class CENCERR: public CBaseErr
                                         Copyright Microsoft, 1999
*
         All Rights Reserved
                                                                                     private:
*/
                                                                                               char
                                                                                                         m_szErrorText[64];
                                                                                     public:
#if !defined(_TPCC_ENCINA_H_)
                                                                                                                                        //
                                                                                               int
                                                                                                                   m errno;
#define TPCC_ENCINA_H_
                                                                                                                   m_iErrorType; // match
                                                                                               int
                                                                          ErrorType in CTPCC ENCINA
#pragma once
                                                                                                                   m_iError;
                                                                                               int
                                                                                                                                        // machine
                                                                          error in CTPCC_ENCINA
// need to declare functions for import, unless define has already been created
// by the DLL's .cpp module for export.
                                                                                               // use this interface for genuine Encina errors
#ifndef DllDecl
                                                                                               CENCERR( int iErr )
#define DllDecl declspec( dllimport )
#endif
                                                                                                         m errno = iErr;
                                                                                                                             // ENCINA error
                                                                                                         m_iErrorType = ERR_TYPE_ENCINA;
class CTPCC_ENCINA: public CTPCC_BASE
                                                                                                         m iError = 0;
                                                                                                                             // only meaningful if
                                                                          m_errno == TPEOS
          private:
                                                                                               };
                    struct ENC_DATA
                                                                                               // use this interface to impersonate a non-Encina error
                              int
                                                                          type
ErrorType;
                                                                                               CENCERR( int iErrorType, int iError )
                              int
                                                                                               {
error;
                                                                                                         m_iErrorType = iErrorType;
                                                                                                         m iError = iError;
                              union
                                                                                                         m errno = iError; // ???
                                         NEW ORDER DATA
NewOrder;
                                                                                               // A CENCERR class can impersonate another class,
                                         PAYMENT_DATA
                                                                          which happens if the error
Payment;
                                                                                               // was not actually a Tuxedo error, but was simply
                                         DELIVERY_DATA
                                                                          transmitted back via Tuxedo.
Delivery;
                                                                                               int ErrorType()
                                         STOCK LEVEL DATA
StockLevel;
                                                                                                         return m iErrorType;
                                         ORDER_STATUS_DATA
OrderStatus;
                              } u;
                                                                                               int ErrorNum() {return m_errno;};
                    } *m_txn;
                                                                                               char *ErrorText();
          public:
                                                                          // wrapper routine for class constructor:
                    CTPCC ENCINA();
                                                                          extern "C" __declspec(dllexport) CTPCC_ENCINA* CTPCC_ENCINA_new();
                    virtual ~CTPCC_ENCINA();
                                                                          extern "C" declspec(dllexport) CTPCC ENCINA*
                                                                          CTPCC_ENCINA_post_init();
```

```
typedef CTPCC_ENCINA* (TYPE_CTPCC_ENCINA)();
                                                                           SQLAllocHandleStd(SQL_HANDLE_ENV, SQL_NULL_HANDLE, &henv)
                                                                           != SQL SUCCESS)
#endif // !defined(_TPCC_ENCINA_H_)
                                                                                                                    return FALSE;
                                                                                                         break;
tpcc odbc.cpp
                                                                                               case DLL PROCESS DETACH:
                                                                                                         if (henv != NULL)
          FILE:
                              TPCC_ODBC.CPP
                                                                                                                    SQLFreeEnv(henv);
                                         Microsoft TPC-C Kit Ver.
                                                                                                         break;
4.20.000
                                         Copyright Microsoft, 1999
                                                                                               default:
          All Rights Reserved
                                                                                                         /* nothing */;
                                         Version 4.10.000 audited by
                                                                                     return TRUE;
Richard Gimarc, Performance Metrics, 3/17/99
          PURPOSE:
                              Implements ODBC calls for TPC-C txns.
                                                                           /* FUNCTION: CTPCC_ODBC_ERR::ErrorText
          Contact: Charles Levine (clevine@microsoft.com)
  Change history:
                    4.20.000 - updated rev number to match kit
                                                                           char* CTPCC ODBC ERR::ErrorText(void)
                    4.10.001 - not deleting error class in catch handler on
deadlock retry;
                                                                                     int i:
                                          not a functional bug, but a
memory leak
                                                                                     static SERRORMSG errorMsgs[] =
                                                                                               { ERR_WRONG_SP_VERSION,
#include <windows.h>
                                                                           "Wrong version of stored procs on database server"
#include <stdio.h>
                                                                                               { ERR_INVALID_CUST,
#include <assert.h>
                                                                           "Invalid Customer id,name."
#define DBNTWIN32
                                                                                               { ERR_NO_SUCH_ORDER,
                                                                                                                                        "No orders
#include <sqltypes.h>
                                                                           found for customer.'
#include <sql.h>
#include <sqlext.h>
                                                                                               { ERR RETRIED TRANS,
                                                                                                                                        "Retries
#include <odbcss.h>
                                                                           before transaction succeeded."
#ifdef ICECAP
#include <icapexp.h>
                                                                                                                              }
#endif
                                                                                     };
// need to declare functions for export
                                                                                     static char szNotFound[] = "Unknown error number.";
#define DllDecl __declspec( dllexport )
                                                                                     for(i=0; errorMsgs[i].szMsg[0]; i++)
#include "..\..\common\src\error.h"
#include "..\..\common\src\trans.h"
                                                                                               if ( m_errno == errorMsgs[i].iError )
#include "..\..\common\src\txn_base.h"
                                                                                                         break:
#include "tpcc odbc.h"
                                                                                     if (!errorMsgs[i].szMsg[0])
// version string; must match return value from tpcc version stored proc
                                                                                               return szNotFound;
const char sVersion[] = "4.10.000";
                                                                                     else
                                                                                               return errorMsgs[i].szMsg;
const iMaxRetries = 10;
                                        // how many retries on deadlock
const int iErrOleDbProvider = 7312;
                                                                           // wrapper routine for class constructor
const char sErrTimeoutExpired[] = "Timeout expired";
                                                                             declspec(dllexport) CTPCC ODBC* CTPCC ODBC new(
                                                                                     LPCSTR szServer,
                                                                                                                    // name of SQL server
static SQLHENV henv = SQL_NULL_HENV;
                                                                                     LPCSTR szUser,
                                                                                                                              // user name for login
// ODBC environment handle
                                                                                     LPCSTR szPassword,
                                                                                                                    // password for login
                                                                                     LPCSTR szHost,
                                                                                                                              // not used
BOOL APIENTRY DllMain(HMODULE hModule, DWORD
                                                                                     LPCSTR szDatabase)
                                                                                                                    // name of database to use
ul reason for call, LPVOID lpReserved)
                                                                                     return new CTPCC_ODBC( szServer, szUser, szPassword, szHost,
          switch( ul_reason_for_call )
                                                                           szDatabase);
```

DisableThreadLibraryCalls(hModule);

case DLL PROCESS ATTACH:

```
CTPCC ODBC::CTPCC ODBC (
                                                                                       rc = SQLExecDirect(m hstmt, (unsigned char *)buffer,
                                                                     SQL NTS);
         LPCSTR szServer,
                                                        // name of
                                                                                       if (rc != SQL SUCCESS && rc !=
SQL server
         LPCSTR szUser,
                                                                     SQL_SUCCESS_WITH_INFO)
                                                                 //
user name for login
                                                                                                 ThrowError(CODBCERR::eExecDirect);
         LPCSTR szPassword,
                                                        // password
for login
                                                                                       // verify that version of stored procs on server is correct
         LPCSTR szHost,
                                                                                        char db sp version[10];
                                                                                        strcpy(buffer, "{call tpcc version}");
not used
                                                                                       rc = SQLExecDirect(m_hstmt, (unsigned char *)buffer,
         LPCSTR szDatabase
                                                        // name of
database to use
                                                                     SQL_NTS);
                                                                                       if (rc != SQL SUCCESS && rc !=
                                                                     SQL SUCCESS WITH INFO)
         RETCODE
                                                                                                 ThrowError(CODBCERR::eExecDirect);
                                     rc;
                                                                                       if (SQLBindCol(m hstmt, 1, SQL C CHAR,
                                                                     &db_sp_version, sizeof(db_sp_version), NULL) != SQL_SUCCESS)
         // initialization
         m hdbc = SQL NULL HDBC;
                                                                                                 ThrowError(CODBCERR::eBindCol);
         m hstmt = SQL NULL HSTMT;
                                                                                        if ( SQLFetch(m_hstmt) == SQL_ERROR )
                                                                                                 ThrowError(CODBCERR::eFetch);
         m hstmtNewOrder = SQL NULL HSTMT;
                                                                                        if (strcmp(db sp version,sVersion))
         m_hstmtPayment = SQL_NULL_HSTMT;
                                                                                                 throw new CTPCC_ODBC_ERR(
         m hstmtDelivery = SQL NULL HSTMT;
                                                                     CTPCC ODBC ERR::ERR WRONG SP VERSION);
         m hstmtOrderStatus = SQL NULL HSTMT;
         m hstmtStockLevel = SQL NULL HSTMT;
                                                                                       SQLFreeHandle(SQL HANDLE STMT, m hstmt);
                                                                              }
         m descNewOrderCols1 = SQL NULL HDESC;
         m descNewOrderCols2 = SQL NULL HDESC;
                                                                              // Bind parameters for each of the transactions
         m_descOrderStatusCols1 = SQL_NULL_HDESC;
                                                                              InitNewOrderParams();
         m descOrderStatusCols2 = SQL NULL HDESC;
                                                                              InitPaymentParams();
                                                                              InitOrderStatusParams();
                                                                              InitDeliveryParams();
         if ( SQLAllocHandle(SQL_HANDLE_DBC, henv, &m_hdbc) !=
                                                                              InitStockLevelParams();
SQL_SUCCESS)
                  ThrowError(CODBCERR::eAllocHandle);
                                                                     CTPCC_ODBC::~CTPCC_ODBC( void )
         if (SQLSetConnectOption(m hdbc, SQL PACKET SIZE, 4096)!=
SQL_SUCCESS)
                                                                              // note: descriptors are automatically released when the connection is
                   ThrowError(CODBCERR::eConnOption);
                                                                     dropped
                                                                              SQLFreeHandle(SQL_HANDLE_STMT, m_hstmtNewOrder);
                                                                              SQLFreeHandle(SQL HANDLE STMT, m hstmtPayment);
                                                                              SQLFreeHandle(SQL_HANDLE_STMT, m_hstmtDelivery);
                   char
                                               szConnectStr[256];
                                               szOutStr[1024];
                                                                              SOLFreeHandle(SQL HANDLE STMT, m_hstmtOrderStatus);
                  char
                  SQLSMALLINT
                                               iOutStrLen;
                                                                              SQLFreeHandle(SQL HANDLE STMT, m hstmtStockLevel);
                  sprintf( szConnectStr, "DRIVER=SQL
                                                                              SQLDisconnect(m hdbc);
                                                                              SQLFreeHandle(SQL_HANDLE_DBC, m_hdbc);
Server;SERVER=%s;UID=%s;PWD=%s;DATABASE=%s",
                            szServer, szUser, szPassword, szDatabase );
                  rc = SQLDriverConnect(m_hdbc, NULL,
                                                                     void CTPCC_ODBC::ThrowError( CODBCERR::ACTION eAction )
(SQLCHAR*)szConnectStr, sizeof(szConnectStr),
                            (SQLCHAR*)szOutStr, sizeof(szOutStr),
                                                                              RETCODE
&iOutStrLen, SQL DRIVER NOPROMPT);
                                                                              SDWORD
                                                                                                 lNativeError;
                                                                              char
                                                                                                 szState[6];
                  if (rc != SQL SUCCESS && rc !=
                                                                              char
                                                                                                 szMsg[SQL MAX MESSAGE LENGTH];
SQL SUCCESS_WITH_INFO)
                            ThrowError(CODBCERR::eConnect);
                                                                     szTmp[6*SQL_MAX_MESSAGE_LENGTH];
                                                                              CODBCERR *pODBCErr;
                                                                                                                             // not
                                                                     allocated until needed (maybe never)
         if (SQLAllocHandle(SQL HANDLE STMT, m hdbc, &m hstmt)
!= SQL_SUCCESS)
                                                                              pODBCErr = new CODBCERR();
                  ThrowError(CODBCERR::eAllocHandle);
                                                                              pODBCErr->m NativeError = 0;
                                                                              pODBCErr->m_eAction = eAction;
                                                                              pODBCErr->m_bDeadLock = FALSE;
                  char
                                     buffer[128];
                  // set some options affecting connection behavior
                                                                              szTmp[0] = 0;
                  strcpy(buffer, "set nocount on set XACT_ABORT ON");
                                                                              while (TRUE)
```

```
rc = SQLError(henv, m hdbc, m hstmt, (BYTE
*)&szState, &lNativeError,
                                                   (BYTE *)&szMsg,
sizeof(szMsg), NULL);
                    if (rc == SQL_NO_DATA)
                              break;
                    // check for deadlock
                    if (INativeError == 1205 || (INativeError ==
iErrOleDbProvider &&
                              strstr(szMsg, sErrTimeoutExpired) != NULL))
                              pODBCErr->m_bDeadLock = TRUE;
                   // capture the (first) database error
                    if (pODBCErr->m_NativeError == 0 && lNativeError !=
0)
                              pODBCErr->m NativeError = lNativeError;
                   // quit if there isn't enough room to concatenate error text
                    if ((strlen(szMsg) + 2) > (sizeof(szTmp) - strlen(szTmp))
)
                              break;
                   // include line break after first error msg
                    if (szTmp[0] != 0)
                              strcat( szTmp, "\n");
                    streat( szTmp, szMsg );
          if (pODBCErr->m_odbcerrstr != NULL)
                    delete [] pODBCErr->m_odbcerrstr;
                   pODBCErr->m_odbcerrstr = NULL;
          if (strlen(szTmp) > 0)
                   pODBCErr->m odbcerrstr = new char[ strlen(szTmp)+1
];
                    strcpy( pODBCErr->m_odbcerrstr, szTmp );
                                                                         period
          SQLFreeStmt(m hstmt, SQL CLOSE);
          throw pODBCErr;
void CTPCC ODBC::InitStockLevelParams()
          if ( SQLAllocHandle(SQL_HANDLE_STMT, m_hdbc,
&m hstmtStockLevel) != SQL SUCCESS )
                   ThrowError(CODBCERR::eAllocHandle);
          m_hstmt = m_hstmtStockLevel;
          if ( SQLBindParameter(m hstmt, ++i, SQL PARAM INPUT,
SQL C SSHORT, SQL SMALLINT, 0, 0, &m txn.StockLevel.w id, 0,
NULL) != SQL_SUCCESS
                    || SQLBindParameter(m hstmt, ++i,
SQL\_PARAM\_INPUT, SQL\_C\_UTINYINT, \overline{S}QL\_TINYINT, 0, 0,
&m txn.StockLevel.d id, 0, NULL) != SQL SUCCESS
                    SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT, SQL_C_SSHORT, SQL_SMALLINT, 0, 0,
&m_txn.StockLevel.threshold, 0, NULL) != SQL_SUCCESS
                    ThrowError(CODBCERR::eBindParam);
```

```
if (SQLBindCol(m hstmt, 1, SQL C SLONG,
&m_txn.StockLevel.low_stock, 0, NULL) != SQL_SUCCESS )
                  ThrowError(CODBCERR::eBindCol);
void CTPCC ODBC::StockLevel()
         RETCODE
         int
                                     iTryCount = 0;
         m_hstmt = m_hstmtStockLevel;
         while (TRUE)
                  trv
                            rc = SQLExecDirectW(m hstmt,
(SQLWCHAR*)L"{call tpcc_stocklevel(?,?,?)}", SQL_NTS);
                            if (rc != SQL_SUCCESS && rc !=
SQL SUCCESS WITH INFO)
ThrowError(CODBCERR::eExecDirect);
                            if (SQLFetch(m hstmt) == SQL ERROR)
ThrowError(CODBCERR::eFetch);
                            SQLFreeStmt(m_hstmt, SQL_CLOSE);
                            m_txn.StockLevel.exec_status_code = eOK;
                            break;
                  catch (CODBCERR *e)
                            if ((!e->m_bDeadLock) || (++iTryCount >
iMaxRetries))
                                     throw:
                            // hit deadlock; backoff for increasingly longer
                            Sleep(10 * iTryCount);
         }
         if (iTryCount)
                  throw new
CTPCC ODBC ERR(CTPCC ODBC ERR::ERR RETRIED TRANS,
iTryCount);
void CTPCC ODBC::InitNewOrderParams()
         if ( SQLAllocHandle(SQL HANDLE STMT, m hdbc,
&m hstmtNewOrder) != SQL SUCCESS
                  || SQLAllocHandle(SQL_HANDLE_DESC, m_hdbc,
&m descNewOrderCols1) != SQL SUCCESS
                  || SQLAllocHandle(SQL_HANDLE_DESC, m_hdbc,
&m descNewOrderCols2) != SQL SUCCESS
                  ThrowError(CODBCERR::eAllocHandle);
         m_hstmt = m_hstmtNewOrder;
         if (SQLSetStmtAttrW(m_hstmt, SQL_ATTR_APP_ROW_DESC,
m descNewOrderCols1, SQL IS POINTER) != SQL SUCCESS)
                  ThrowError(CODBCERR::eSetStmtAttr);
```

```
int i = 0;
         if ( SQLBindParameter(m hstmt, ++i, SQL PARAM INPUT,
SQL C SSHORT, SQL SMALLINT, 0, 0, &m txn.NewOrder.w id, 0,
NULL) != SQL_SUCCESS
                   || SOLBindParameter(m_hstmt, ++i,
SQL PARAM INPUT, SQL C UTINYINT, SQL TINYINT, 0, 0,
&m txn.NewOrder.d id, 0, NULL) != SQL SUCCESS
                    SQLBindParameter(m hstmt, ++i
SQL_PARAM_INPUT, SQL_C_SLONG, SQL_INTEGER, 0, 0,
&m_txn.NewOrder.c id, 0, NULL) != SQL SUCCESS
                   SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT, SQL_C_UTINYINT, SQL_TINYINT, 0, 0, &m_txn.NewOrder.o_ol_cnt, 0, NULL) != SQL_SUCCESS
                   || SQLBindParameter(m_hstmt, ++i,
SQL PARAM INPUT, SQL C UTINYINT, SQL TINYINT, 0, 0,
&m_txn.NewOrder.o_all_local, 0, NULL) != SQL_SUCCESS
                   ThrowError(CODBCERR::eBindParam);
         for (int j=0; j<MAX OL NEW ORDER ITEMS; j++)
                   if ( SQLBindParameter(m hstmt, ++i,
SQL_PARAM_INPUT, SQL_C_SLONG, SQL_INTEGER, 0, 0,
&m_txn.NewOrder.OL[j].ol_i_id, 0, NULL) != SQL_SUCCESS
                            || SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT, SQL_C_SSHORT, SQL_SMALLINT, 0, 0,
&m txn.NewOrder.OL[j].ol supply w id, 0, NULL) != SQL SUCCESS
                            || SQLBindParameter(m_hstmt, ++i,
SQL PARAM INPUT, SQL C SSHORT, SQL SMALLINT, 0, 0,
&m_txn.NewOrder.OL[j].ol_quantity, 0, NULL) != SQL_SUCCESS
                            ThrowError(CODBCERR::eBindParam);
         // set the bind offset pointer
         if (SQLSetStmtAttrW(m hstmt,
SQL_ATTR_ROW_BIND_OFFSET_PTR, &m_BindOffset,
SQL IS POINTER) != SQL SUCCESS)
                  ThrowError(CODBCERR::eSetStmtAttr);
         if ( SQLBindCol(m_hstmt, ++i, SQL_C_CHAR,
&m txn.NewOrder.OL[0].ol i name,
sizeof(m_txn.NewOrder.OL[0].ol_i_name), NULL) != SQL_SUCCESS
                   || SQLBindCol(m hstmt, ++i, SQL C SSHORT,
&m_txn.NewOrder.OL[0].ol_stock, 0, NULL) != SQL_SUCCESS
                   || SQLBindCol(m hstmt, ++i, SQL C CHAR,
&m txn.NewOrder.OL[0].ol brand generic,
sizeof(m_txn.NewOrder.OL[0].ol_brand_generic), NULL) != SQL_SUCCESS
                   || SQLBindCol(m hstmt, ++i, SQL C DOUBLE,
&m_txn.NewOrder.OL[0].ol_i_price, 0, NULL) != SQL_SUCCESS
                   || SQLBindCol(m hstmt, ++i, SQL C DOUBLE,
&m_txn.NewOrder.OL[0].ol_amount, 0, NULL) != SQL_SUCCESS
                  ThrowError(CODBCERR::eBindCol);
         // associate the column bindings for the second result set
         if (SQLSetStmtAttrW(m_hstmt, SQL_ATTR_APP_ROW_DESC,
m descNewOrderCols2, SQL IS POINTER) != SQL SUCCESS)
                   ThrowError(CODBCERR::eSetStmtAttr);
         if ( SQLBindCol(m_hstmt, ++i, SQL_C_DOUBLE,
&m_txn.NewOrder.w_tax, 0, NULL) != SQL_SUCCESS
                   || SQLBindCol(m_hstmt, ++i, SQL_C_DOUBLE,
&m_txn.NewOrder.d_tax, 0, NULL) != SQL SUCCESS
                   || SQLBindCol(m_hstmt, ++i, SQL_C_SLONG,
&m_txn.NewOrder.o_id, 0, NULL) != SQL_SUCCESS
```

```
|| SQLBindCol(m hstmt, ++i, SQL C CHAR,
&m txn.NewOrder.c last, sizeof(m txn.NewOrder.c last), NULL) !=
SQL SUCCESS
                    || SQLBindCol(m_hstmt, ++i, SQL_C_DOUBLE,
&m txn.NewOrder.c discount, 0, NULL) != SQL SUCCESS
                   || SQLBindCol(m hstmt, ++i, SQL C CHAR,
&m_txn.NewOrder.c_credit, sizeof(m_txn.NewOrder.c_credit), NULL) !=
SQL_SUCCESS
                    || SQLBindCol(m hstmt, ++i,
SQL C TYPE TIMESTAMP, &m txn.NewOrder.o entry d, 0, NULL) !=
SQL_SUCCESS
                    || SQLBindCol(m_hstmt, ++i, SQL_C_SLONG,
&m no commit flag, 0, NULL) != SQL SUCCESS
                    ThrowError(CODBCERR::eBindCol);
void CTPCC_ODBC::NewOrder()
                                                           i;
          RETCODE
                                                           rc;
                                                           iTryCount =
          int
0;
                    // 0
                                  2
                    // 012345678901234567890123456789
          wchar_t
                                                  szSqlTemplate[] =
L"{call tpcc neworder(?,?,?,?,","
                     L"?,?,?,?,?,?,?,?,?,?,?,?,"
                     L"?,?,?,?,?,?,?,?,?,?,?,?,"
                     L"?,?,?,?,?,?,?,?,?,?,?,?)}";
         m_hstmt = m_hstmtNewOrder;
         // associate the parameter and column bindings for this transaction
         if (SQLSetStmtAttrW(m hstmt, SQL ATTR APP ROW DESC,
m descNewOrderCols1, SQL IS POINTER) != SQL SUCCESS)
                    ThrowError(CODBCERR::eSetStmtAttr);
         // clip statement buffer based on number of parameters
          // fixed part is 29 chars and variable part is 6 chars per line item
          i = 29 + m_txn.NewOrder.o_ol_cnt*6;
          wcscpy(&szSqlTemplate[i], L")}");
         // check whether any order lines are for a remote warehouse
          m txn.NewOrder.o all local = 1;
          for (i = 0; i \le m_{txn.NewOrder.o_ol_cnt; i++)
                   if (m_txn.NewOrder.OL[i].ol_supply_w_id !=
m txn.NewOrder.w_id)
                              m_txn.NewOrder.o_all_local = 0; // at least
one remote warehouse
                              break:
          while (TRUE)
                    try
                              m BindOffset = 0;
                             rc = SQLExecDirectW(m_hstmt,
(SQLWCHAR*)szSqlTemplate, SQL NTS);
```

```
if (rc != SQL SUCCESS && rc !=
SQL SUCCESS WITH INFO)
                                                                                if (iTryCount)
                                                                                         throw new
                                                                      CTPCC_ODBC_ERR(CTPCC_ODBC_ERR::ERR_RETRIED_TRANS,
ThrowError(CODBCERR::eExecDirect);
                                                                      iTryCount);
                            // Get order line results
                            m txn.NewOrder.total amount = 0;
                            for (i = 0; i \le m \ txn. New Order.o \ ol \ cnt; i++)
                                                                      void CTPCC ODBC::InitPaymentParams()
                                                                                if ( SQLAllocHandle(SQL HANDLE STMT, m hdbc,
                                      // set the bind offset value...
                                                                      &m_hstmtPayment) != SQL_SUCCESS )
                                      m BindOffset = i *
sizeof(m txn.NewOrder.OL[0]);
                                                                                         ThrowError(CODBCERR::eAllocHandle);
                                                                                m_hstmt = m_hstmtPayment;
                                      if (SQLFetch(m hstmt) ==
SQL ERROR)
                                                                                int i = 0;
                                                                                if ( SQLBindParameter(m hstmt, ++i, SQL PARAM INPUT,
ThrowError(CODBCERR::eFetch);
                                                                      SQL_C_SSHORT, SQL_SMALLINT, 0, 0, &m_txn.Payment.w_id, 0, NULL)
                                                                      != SQL SUCCESS
                                      // move to the next resultset
                                                                                         || SQLBindParameter(m hstmt, ++i,
                                      if (SQLMoreResults(m hstmt) ==
                                                                      SQL_PARAM_INPUT, SQL_C_SSHORT, SQL_SMALLINT, 0, 0,
SQL ERROR)
                                                                      &m txn.Payment.c w id, 0, NULL) != SQL SUCCESS
                                                                                         || SQLBindParameter(m hstmt, ++i,
ThrowError(CODBCERR::eMoreResults);
                                                                      SQL PARAM INPUT, SQL C DOUBLE, SQL NUMERIC, 6, 2,
                                                                      &m_txn.Payment.h_amount, 0, NULL) != SQL_SUCCESS
                                      m txn.NewOrder.total amount +=
                                                                                          || SQLBindParameter(m_hstmt, ++i,
m_txn.NewOrder.OL[i].ol_amount;
                                                                      SQL PARAM INPUT, SQL C UTINYINT, SQL TINYINT, 0, 0,
                                                                      &m_txn.Payment.d_id, 0, NULL) != SQL_SUCCESS
                                                                                         || SQLBindParameter(m hstmt, ++i,
                            // associate the column bindings for the
                                                                      SQL_PARAM_INPUT, SQL_C_UTINYINT, SQL_TINYINT, 0, 0,
second result set
                                                                      &m_txn.Payment.c_d_id, 0, NULL) != SQL_SUCCESS
                            if (SQLSetStmtAttrW(m hstmt,
                                                                                         || SQLBindParameter(m_hstmt, ++i,
SQL ATTR APP ROW DESC, m descNewOrderCols2, SQL IS POINTER
                                                                      SQL PARAM INPUT, SQL C SLONG, SQL INTEGER, 0, 0,
) != SQL_SUCCESS )
                                                                      &m txn.Payment.c id, 0, NULL) != SQL SUCCESS
                                                                                          SQLBindParameter(m_hstmt, ++i,
ThrowError(CODBCERR::eSetStmtAttr);
                                                                      SQL PARAM INPUT, SQL C CHAR, SQL CHAR,
                                                                      sizeof(m_txn.Payment.c_last), 0, &m_txn.Payment.c_last,
                            if (SQLFetch(m hstmt) == SQL ERROR)
                                                                      sizeof(m txn.Payment.c last), NULL) != SQL SUCCESS
ThrowError(CODBCERR::eFetch);
                                                                                         ThrowError(CODBCERR::eBindParam);
                            SQLFreeStmt(m_hstmt, SQL_CLOSE);
                                                                                if ( SQLBindCol(m hstmt, ++i, SQL C SLONG,
                            if (m no commit flag == 1)
                                                                                                             0, NULL) != SQL SUCCESS
                                                                      &m_txn.Payment.c_id,
                                                                                          || SQLBindCol(m hstmt, ++i, SQL C CHAR,
                                      m txn.NewOrder.total amount *=
                                                                      &m_txn.Payment.c_last,
                                                                                                            sizeof(m_txn.Payment.c last),
((1 + m txn.NewOrder.w tax + m txn.NewOrder.d tax) * (1 -
                                                                      NULL) != SQL SUCCESS
m txn.NewOrder.c discount));
                                                                                          || SQLBindCol(m hstmt, ++i,
                                                                      SQL_C_TYPE_TIMESTAMP, &m_txn.Payment.h_date,
m txn.NewOrder.exec status code = eOK;
                                                                      0, NULL) != SQL SUCCESS
                                                                                          || SQLBindCol(m_hstmt, ++i, SQL_C_CHAR,
                            else
                                                                                                  sizeof(m_txn.Payment.w_street 1), NULL) !=
                                                                      &m txn.Payment.w street 1,
                                                                      SQL_SUCCESS
m_txn.NewOrder.exec_status_code = eInvalidItem;
                                                                                         || SQLBindCol(m hstmt, ++i, SQL C CHAR,
                                                                                                   sizeof(m_txn.Payment.w_street_2), NULL) !=
                                                                      &m_txn.Payment.w_street_2,
                            break:
                                                                      SQL SUCCESS
                                                                                         || SQLBindCol(m hstmt, ++i, SQL C CHAR,
                   catch (CODBCERR *e)
                                                                      &m_txn.Payment.w_city,
                                                                                                             sizeof(m_txn.Payment.w_city),
                                                                      NULL) != SQL SUCCESS
                            if ((!e->m bDeadLock) || (++iTryCount >
                                                                                         || SQLBindCol(m_hstmt, ++i, SQL_C_CHAR,
iMaxRetries))
                                                                      &m txn.Payment.w state,
                                                                                                            sizeof(m txn.Payment.w state),
                                      throw;
                                                                      NULL) != SQL SUCCESS
                                                                                          || SQLBindCol(m_hstmt, ++i, SQL_C_CHAR,
                            // hit deadlock; backoff for increasingly longer
                                                                      &m_txn.Payment.w_zip,
                                                                                                             sizeof(m_txn.Payment.w_zip),
period
                                                                      NULL) != SQL_SUCCESS
                            delete e:
                                                                                         || SQLBindCol(m hstmt, ++i, SQL C CHAR,
                            Sleep(10 * iTryCount);
                                                                      &m_txn.Payment.d_street_1,
                                                                                                   sizeof(m_txn.Payment.d_street_1), NULL) !=
                   }
                                                                      SQL_SUCCESS
```

}

```
|| SQLBindCol(m hstmt, ++i, SQL C CHAR,
&m txn.Payment.d street 2,
                            sizeof(m txn.Payment.d street 2), NULL) !=
                                                                                                   rc = SQLExecDirectW(m hstmt,
                                                                      (SQLWCHAR*)L"{call tpcc_payment(?,?,?,?,?,?,?)}", SQL NTS);
SQL SUCCESS
                   || SQLBindCol(m_hstmt, ++i, SQL_C_CHAR,
                                                                                                   if (rc != SQL_SUCCESS && rc !=
&m txn.Payment.d city,
                                      sizeof(m_txn.Payment.d_city),
                                                                      SQL_SUCCESS_WITH_INFO)
NULL) != SQL SUCCESS
                   || SQLBindCol(m hstmt, ++i, SQL C CHAR,
                                                                      ThrowError(CODBCERR::eExecDirect);
                                      sizeof(m_txn.Payment.d_state),
&m txn.Payment.d state,
NULL) != SQL SUCCESS
                                                                                                   if (SQLFetch(m hstmt) == SQL ERROR)
                   || SQLBindCol(m hstmt, ++i, SQL C CHAR,
                                                                      ThrowError(CODBCERR::eFetch);
&m_txn.Payment.d_zip,
                                      sizeof(m_txn.Payment.d_zip),
NULL) != SQL_SUCCESS
                   || SQLBindCol(m hstmt, ++i, SQL C CHAR,
                                                                                                   SQLFreeStmt(m hstmt, SQL CLOSE);
&m_txn.Payment.c_first,
                                      sizeof(m_txn.Payment.c_first),
NULL) != SQL_SUCCESS
                                                                                                   if (m_txn.Payment.c_id == 0)
                   || SQLBindCol(m hstmt, ++i, SQL C CHAR,
                                                                                                            throw new CTPCC ODBC ERR(
                                                                      CTPCC ODBC_ERR::ERR_INVALID_CUST );
&m txn.Payment.c middle,
                            sizeof(m txn.Payment.c middle), NULL) !=
SQL SUCCESS
                   || SQLBindCol(m hstmt, ++i, SQL C CHAR,
                                                                                                            m_txn.Payment.exec_status_code
&m txn.Payment.c street 1,
                            sizeof(m txn.Payment.c street 1), NULL) !=
                                                                      = eOK;
SQL_SUCCESS
                   || SQLBindCol(m hstmt, ++i, SQL C CHAR,
                                                                                                   break;
&m_txn.Payment.c_street_2,
                            sizeof(m_txn.Payment.c_street_2), NULL) !=
SQL_SUCCESS
                                                                                         catch (CODBCERR *e)
                   || SQLBindCol(m_hstmt, ++i, SQL_C_CHAR,
&m txn.Payment.c city,
                                      sizeof(m_txn.Payment.c_city),
                                                                                                   if ((!e->m_bDeadLock) || (++iTryCount >
NULL) != SQL SUCCESS
                                                                      iMaxRetries))
                   || SQLBindCol(m_hstmt, ++i, SQL_C_CHAR,
                                                                                                            throw:
&m txn.Payment.c state,
                                      sizeof(m txn.Payment.c state),
NULL) != SQL_SUCCESS
                                                                                                   // hit deadlock; backoff for increasingly longer
                   || SQLBindCol(m hstmt, ++i, SQL C CHAR,
                                                                      period
&m_txn.Payment.c_zip,
                                      sizeof(m_txn.Payment.c_zip),
                                                                                                   delete e;
NULL) != SQL_SUCCESS
                                                                                                   Sleep(10 * iTryCount);
                   || SQLBindCol(m hstmt, ++i, SQL C CHAR,
&m_txn.Payment.c_phone,
                                      sizeof(m_txn.Payment.c_phone),
                                                                                }
NULL) != SQL SUCCESS
                   || SQLBindCol(m_hstmt, ++i,
                                                                               if (iTryCount)
SQL C TYPE TIMESTAMP, &m txn.Payment.c since,
                                                                                         throw new
0, NULL) != SQL_SUCCESS
                                                                      CTPCC_ODBC_ERR(CTPCC_ODBC_ERR::ERR_RETRIED_TRANS,
                   || SQLBindCol(m hstmt, ++i, SQL C CHAR,
                                                                      iTryCount);
&m txn.Payment.c credit,
                            sizeof(m txn.Payment.c credit), NULL) !=
SQL_SUCCESS
                   || SQLBindCol(m hstmt, ++i, SQL C DOUBLE,
                                                                      void CTPCC ODBC::InitOrderStatusParams()
&m_txn.Payment.c_credit_lim,0, NULL) != SQL_SUCCESS
                   || SQLBindCol(m hstmt, ++i, SQL C DOUBLE,
                                                                                if ( SQLAllocHandle(SQL HANDLE STMT, m hdbc,
                            0, NULL) != SQL_SUCCESS
                                                                      &m_hstmtOrderStatus) != SQL_SUCCESS
&m_txn.Payment.c_discount,
                   || SQLBindCol(m hstmt, ++i, SQL C DOUBLE,
                                                                                         || SQLAllocHandle(SQL_HANDLE_DESC, m_hdbc,
                            0, NULL) != SQL SUCCESS
&m txn.Payment.c balance,
                                                                      &m descOrderStatusCols1) != SQL SUCCESS
                                                                                         \parallel SQLAllocHandle(SQL\_HANDLE\_DESC, m\_hdbc,
                   || SQLBindCol(m_hstmt, ++i, SQL_C_CHAR,
&m txn.Payment.c data,
                                      sizeof(m txn.Payment.c data),
                                                                      &m descOrderStatusCols2) != SQL SUCCESS
NULL) != SQL_SUCCESS
                                                                                         ThrowError(CODBCERR::eAllocHandle);
                   ThrowError(CODBCERR::eBindCol);
                                                                               m hstmt = m hstmtOrderStatus;
}
void CTPCC_ODBC::Payment()
                                                                                if ( SQLSetStmtAttrW( m_hstmt, SQL_ATTR_APP_ROW_DESC,
                                                                      m descOrderStatusCols1, SQL IS POINTER) != SQL SUCCESS)
         RETCODE
                                                                                         ThrowError(CODBCERR::eSetStmtAttr);
                                      iTryCount = 0;
                                                                                int i = 0;
         m hstmt = m hstmtPayment;
                                                                                if ( SQLBindParameter(m hstmt, ++i, SQL PARAM INPUT,
                                                                      SQL C SSHORT, SQL SMALLINT, 0, 0, &m txn.OrderStatus.w id, 0,
                                                                      NULL) != SQL_SUCCESS
         if (m txn.Payment.c id != 0)
                   m_txn.Payment.c_last[0] = 0;
                                                                                          SQLBindParameter(m_hstmt, ++i,
                                                                      SQL_PARAM_INPUT, SQL_C_UTINYINT, SQL_TINYINT, 0, 0,
         while (TRUE)
                                                                      &m txn.OrderStatus.d id, 0, NULL) != SQL SUCCESS
```

```
|| SQLBindParameter(m hstmt, ++i,
                                                                                int
SQL PARAM INPUT, SQL C SLONG, SQL INTEGER, 0, 0,
                                                                       iTryCount = 0;
&m txn.OrderStatus.c id, 0, NULL) != SQL SUCCESS
                                                                                 RETCODE
                   SQLBindParameter(m_hstmt, ++i,
SQL PARAM INPUT, SQL C CHAR, SQL CHAR,
sizeof(m txn.OrderStatus.c last), 0, &m txn.OrderStatus.c last,
                                                                                 m hstmt = m hstmtOrderStatus;
sizeof(m txn.OrderStatus.c last), NULL) != SQL SUCCESS
                                                                                 if ( SQLSetStmtAttrW( m_hstmt, SQL_ATTR_APP_ROW_DESC,
                   ThrowError(CODBCERR::eBindParam);
                                                                       m descOrderStatusCols1, SQL IS POINTER) != SQL SUCCESS)
                                                                                          ThrowError(CODBCERR::eSetStmtAttr);
         // configure block cursor
         if ( SQLSetStmtAttrW(m hstmt,
                                                                                 if (m txn.OrderStatus.c id != 0)
SQL ATTR ROW BIND TYPE,
                                                                                          m txn.OrderStatus.c last[0] = 0;
(SQLPOINTER)sizeof(m_txn.OrderStatus.OL[0]), 0) != SQL_SUCCESS
                   || SQLSetStmtAttrW(m hstmt,
                                                                                 while (TRUE)
SQL_ATTR_ROWS_FETCHED_PTR, &m_RowsFetched, 0) !=
SQL SUCCESS
                                                                                          try
                   ThrowError(CODBCERR::eSetStmtAttr);
                                                                                                    // configure block cursor
                                                                                                    if (SQLSetStmtAttrW(m hstmt,
                                                                       SQL_ATTR_ROW_ARRAY_SIZE, (SQLPOINTER)1, 0) != SQL_SUCCESS)
         i = 0;
         if ( SQLBindCol(m hstmt, ++i, SQL C SSHORT,
&m_txn.OrderStatus.OL[0].ol_supply_w_id, 0, NULL) != SQL_SUCCESS
                                                                       ThrowError(CODBCERR::eSetStmtAttr);
                   || SQLBindCol(m hstmt, ++i, SQL C SLONG,
&m_txn.OrderStatus.OL[0].ol_i_id, 0, NULL) != SQL_SUCCESS
                                                                                                    rc = SQLExecDirectW(m_hstmt,
                                                                        \begin{aligned} &(SQLWCHAR*)L"\{call\ tpcc\_orderstatus(?,?,?,?)\}",\ SQL\_NTS);\\ &\quad if\ (\ ((rc == SQL\_SUCCESS\_WITH\_INFO)); \end{aligned} 
                   || SQLBindCol(m_hstmt, ++i, SQL_C_SSHORT,
&m txn.OrderStatus.OL[0].ol quantity, 0, NULL) != SQL SUCCESS
                   || SQLBindCol(m_hstmt, ++i, SQL_C_DOUBLE,
                                                                       && (m_RowsFetched != 0)) \parallel (rc == SQL_ERROR))
&m txn.OrderStatus.OL[0].ol amount, 0, NULL) != SQL SUCCESS
                   || SQLBindCol(m_hstmt, ++i,
                                                                       ThrowError(CODBCERR::eExecDirect);
SQL C TYPE TIMESTAMP, &m txn.OrderStatus.OL[0].ol delivery d, 0,
NULL) != SQL_SUCCESS
                                                                                                    // configure block cursor
                                                                                                    if (SQLSetStmtAttrW(m hstmt,
                   ThrowError(CODBCERR::eBindCol);
                                                                       SOL ATTR ROW ARRAY SIZE.
                                                                       (SQLPOINTER)MAX_OL_ORDER_STATUS_ITEMS, 0) != SQL_SUCCESS
         if (SQLSetStmtAttrW(m_hstmt, SQL_ATTR_APP_ROW_DESC,
m descOrderStatusCols2, SQL IS POINTER) != SQL SUCCESS)
                                                                       ThrowError(CODBCERR::eSetStmtAttr);
                   ThrowError(CODBCERR::eSetStmtAttr);
                                                                                                    rc = SQLFetchScroll( m hstmt,
                                                                       SQL FETCH NEXT, 0);
         i = 0;
         if ( SQLBindCol(m_hstmt, ++i, SQL_C_SLONG,
                                                                                                    if ( ((rc == SQL_SUCCESS_WITH_INFO)
&m txn.OrderStatus.c id, 0, NULL) != SQL SUCCESS
                                                                       && (m RowsFetched != 0)) \parallel (rc == SQL ERROR))
                    SQLBindCol(m_hstmt, ++i, SQL_C_CHAR,
&m txn.OrderStatus.c last, sizeof(m txn.OrderStatus.c last), NULL) !=
                                                                       ThrowError(CODBCERR::eFetchScroll);
SQL_SUCCESS
                   || SQLBindCol(m hstmt, ++i, SQL C CHAR,
                                                                                                    m txn.OrderStatus.o ol cnt =
&m txn.OrderStatus.c first, sizeof(m txn.OrderStatus.c first), NULL) !=
                                                                       (short)m RowsFetched;
SQL_SUCCESS
                   || SQLBindCol(m hstmt, ++i, SQL C CHAR,
                                                                                                    if (m txn.OrderStatus.o ol cnt != 0)
&m_txn.OrderStatus.c_middle, sizeof(m_txn.OrderStatus.c_middle), NULL) !=
SQL SUCCESS
                                                                                                              if (SQLSetStmtAttrW(m hstmt,
                                                                       SQL_ATTR_APP_ROW_DESC, m_descOrderStatusCols2,
                   || SQLBindCol(m_hstmt, ++i,
SQL C TYPE TIMESTAMP, &m txn.OrderStatus.o entry d, 0, NULL) !=
                                                                       SQL IS POINTER) != SQL SUCCESS)
SQL SUCCESS
                                                                       ThrowError(CODBCERR::eSetStmtAttr);
                   \parallel SQLBindCol(m_hstmt, ++i, SQL_C_SSHORT,
&m txn.OrderStatus.o carrier id, 0, NULL) != SQL SUCCESS
                    SQLBindCol(m_hstmt, ++i, SQL_C_DOUBLE,
                                                                                                              if ( SQLMoreResults(m_hstmt) ==
&m txn.OrderStatus.c balance, 0, NULL) != SQL SUCCESS
                                                                       SQL ERROR)
                   SQLBindCol(m_hstmt, ++i, SQL_C_SLONG,
&m_txn.OrderStatus.o_id, 0, NULL) != SQL_SUCCESS
                                                                       ThrowError(CODBCERR::eMoreResults);
                   ThrowError(CODBCERR::eBindCol);
                                                                                                              if ( (rc = SQLFetch(m hstmt)) ==
                                                                       SQL ERROR)
void CTPCC ODBC::OrderStatus()
                                                                       ThrowError(CODBCERR::eFetch);
```

```
SQLFreeStmt(m_hstmt, SQL_CLOSE);
                                                                                           try
                             if (m txn.OrderStatus.o ol cnt == 0)
                                       throw new CTPCC_ODBC_ERR(
                                                                                                    rc = SQLExecDirectW(m hstmt,
CTPCC ODBC ERR::ERR NO SUCH ORDER );
                                                                       (SQLWCHAR*)L"{call tpcc_delivery(?,?)}", SQL_NTS);
                             else if (m txn.OrderStatus.c id == 0 &&
                                                                                                    if (rc != SQL SUCCESS && rc !=
m txn.OrderStatus.c last[0] == 0
                                                                       SQL SUCCESS WITH INFO)
                                       throw new CTPCC ODBC ERR(
                                                                       ThrowError(CODBCERR::eExecDirect);
CTPCC ODBC ERR::ERR INVALID CUST );
                                                                                                    if ( SQLFetch(m_hstmt) == SQL_ERROR )
m_txn.OrderStatus.exec_status_code = eOK;
                                                                       ThrowError(CODBCERR::eFetch);
                             break:
                                                                                                    SQLFreeStmt(m hstmt, SQL CLOSE);
                   catch (CODBCERR *e)
                                                                                                    m_txn.Delivery.exec_status_code = eOK;
                                                                                                    break;
                             if ((!e->m_bDeadLock) || (++iTryCount >
iMaxRetries))
                                                                                           catch (CODBCERR *e)
                                       throw;
                                                                                                    if ((!e->m_bDeadLock) || (++iTryCount >
                             // hit deadlock; backoff for increasingly longer
                                                                       iMaxRetries))
period
                                                                                                              throw;
                             delete e:
                             Sleep(10 * iTryCount);
                                                                                                    // hit deadlock; backoff for increasingly longer
                                                                       period
                   }
                                                                                                    delete e;
                                                                                                    Sleep(10 * iTryCount);
         if (iTryCount)
                   throw new
CTPCC ODBC ERR(CTPCC ODBC ERR::ERR RETRIED TRANS,
iTryCount);
                                                                                 if (iTryCount)
                                                                                           throw new
                                                                       CTPCC ODBC ERR(CTPCC ODBC ERR::ERR RETRIED TRANS,
void CTPCC_ODBC::InitDeliveryParams()
                                                                       iTryCount);
         if ( SQLAllocHandle(SQL_HANDLE_STMT, m_hdbc,
&m hstmtDelivery) != SQL SUCCESS)
                   ThrowError(CODBCERR::eAllocHandle);
                                                                       tpcc odbc.h
         m hstmt = m hstmtDelivery;
                                                                                 FILE:
                                                                                                    TPCC ODBC.H
                                                                                                              Microsoft TPC-C Kit Ver.
                                                                       4.20.000
         if ( SQLBindParameter(m_hstmt, ++i, SQL_PARAM_INPUT,
                                                                                                              Copyright Microsoft, 1999
SQL C SSHORT, SQL SMALLINT, 0, 0, &m txn.Delivery.w id, 0, NULL)
                                                                                All Rights Reserved
!= SQL_SUCCESS
                   || SQLBindParameter(m hstmt, ++i,
                                                                                                              Version 4.10.000 audited by
SQL_PARAM_INPUT, SQL_C_SSHORT, SQL_SMALLINT, 0, 0,
                                                                       Richard Gimarc, Performance Metrics, 3/17/99
&m_txn.Delivery.o_carrier_id, 0, NULL) != SQL_SUCCESS
                                                                                 PURPOSE:
                                                                                                    Header file for TPC-C txn class
                   ThrowError(CODBCERR::eBindParam);
                                                                       implementation.
         for (i=0;i<10;i++)
                                                                          Change history:
                                                                                           4.20.000 - updated rev number to match kit
                   if ( SQLBindCol(m_hstmt, (UWORD)(i+1),
SQL_C_SLONG, &m_txn.Delivery.o_id[i], 0, NULL) != SQL_SUCCESS)
                                                                       #pragma once
                             ThrowError(CODBCERR::eBindCol);
         }
                                                                       // need to declare functions for import, unless define has already been created
                                                                       // by the DLL's .cpp module for export.
                                                                       #ifndef DllDecl
void CTPCC_ODBC::Delivery()
                                                                       #define DllDecl declspec( dllimport )
                                                                       #endif
         RETCODE
                                       rc;
                                       iTryCount = 0;
                                                                       class CODBCERR: public CBaseErr
         m hstmt = m hstmtDelivery;
                                                                                 public:
                                                                                           enum ACTION
         while (TRUE)
```

```
ERR_NO_SUCH_ORDER,
                             eNone,
                             eUnknown,
                                                                          "No orders found for customer."
                                                                                                      ERR RETRIED TRANS.
                             eAllocConn,
                                                                     //
error from SQLAllocConnect
                                                                        // "Retries before transaction succeeded."
                             eAllocHandle,
                                                           // error from
SQLAllocHandle
                                                                                            CTPCC ODBC ERR( int iErr ) { m errno = iErr;
                             eConnOption,
                                                           // error from
SQLSetConnectOption
                                                                        m iTryCount = 0; };
                             eConnect,
                                                           // error from
                                                                                            CTPCC ODBC ERR( int iErr, int iTryCount ) {
SQLConnect
                                                                        m_errno = iErr; m_iTryCount = iTryCount; };
                             eAllocStmt,
                                                                     //
error from SQLAllocStmt
                             eExecDirect,
                                                           // error from
                                                                                            int
                                                                                                                m errno;
SQLExecDirect
                                                                                                                m_iTryCount;
                                                                                            int
                             eBindParam,
                                                                     //
error from SQLBindParameter
                                                                                            int ErrorType() {return ERR TYPE TPCC ODBC;};
                             eBindCol,
                                                           // error from
                                                                                            int ErrorNum() {return m errno;};
SQLBindCol
                             eFetch,
                                                                                            char *ErrorText();
error from SQLFetch
                             eFetchScroll,
                                                           // error from
SQLFetchScroll
                                                                        class DllDecl CTPCC ODBC: public CTPCC BASE
                             eMoreResults,
                                                           // error from
SQLMoreResults
                                                                                  private:
                             ePrepare,
                                                           // error from
                                                                                            // declare variables and private functions here...
                                                                                            BOOL
                                                                                                                m bDeadlock;
                                                                                                                                             //
SQLPrepare
                             eExecute,
                                                           // error from
                                                                        transaction was selected as deadlock victim
SQLExecute
                                                                                                                          m_MaxRetries;
                                                                                            int
                             eSetEnvAttr,
                                                           // error from
                                                                        // retry count on deadlock
SQLSetEnvAttr
                             eSetStmtAttr
                                                           // error from
                                                                                            SQLHENV
                                                                                                                          m henv;
SQLSetStmtAttr
                                                                                  // ODBC environment handle
                                                                                            SOLHDBC
                                                                                                                          m_hdbc;
                   };
                                                                                            SQLHSTMT
                                                                                                                m hstmt;
                                                                                                                                             //
                   CODBCERR(void)
                                                                        the current hstmt
                             m_eAction = eNone;
                                                                                            SQLHSTMT
                                                                                                                m_hstmtNewOrder;
                             m NativeError = 0;
                                                                                            SQLHSTMT
                                                                                                                m hstmtPayment;
                             m_bDeadLock = FALSE;
                                                                                            SQLHSTMT
                                                                                                                m_hstmtDelivery;
                             m odbcerrstr = NULL;
                                                                                            SQLHSTMT
                                                                                                                m hstmtOrderStatus;
                                                                                            SQLHSTMT
                                                                                                                m hstmtStockLevel;
                   };
                   ~CODBCERR()
                                                                                            SQLHDESC
                                                                                                                m descNewOrderCols1;
                                                                                            SQLHDESC
                                                                                                                m descNewOrderCols2;
                             if (m odbcerrstr!= NULL)
                                                                                            SQLHDESC
                                                                                                                m descOrderStatusCols1;
                                       delete [] m_odbcerrstr;
                                                                                            SQLHDESC
                                                                                                                m_descOrderStatusCols2;
                   };
                                                                                            // new-order specific fields
                   ACTION m_eAction;
                                                                                            SQLUINTEGER
                                                                                                                m BindOffset;
                                       m NativeError;
                                                                                            SQLUINTEGER
                                                                                                                m RowsFetched;
                   BOOL
                             m bDeadLock;
                                                                                                                          m_no_commit_flag;
                   char *m odbcerrstr;
                                                                                            void ThrowError( CODBCERR::ACTION eAction );
                   int ErrorType() {return ERR TYPE ODBC;};
                   int ErrorNum() {return m NativeError;};
                                                                                            void InitNewOrderParams();
                   char *ErrorText() {return m_odbcerrstr;};
                                                                                            void InitPaymentParams();
                                                                                            void InitDeliveryParams();
};
                                                                                            void InitStockLevelParams();
                                                                                            void InitOrderStatusParams();
class CTPCC_ODBC_ERR: public CBaseErr
                                                                                            union
          public:
                   enum TPCC_ODBC_ERRS
                                                                                                      NEW_ORDER_DATA
                                                                        NewOrder;
                             ERR_WRONG_SP_VERSION = 1,
                                                                     //
                                                                                                      PAYMENT_DATA
                                                                                                                                    Payment;
"Wrong version of stored procs on database server"
                                                                                                      DELIVERY DATA
                                                                                                                                    Delivery;
                             ERR_INVALID_CUST,
                                                                                                      STOCK_LEVEL_DATA
                                                                                                                                    StockLevel;
// "Invalid Customer id,name."
                                                                                                      ORDER STATUS DATA
                                                                                                                                    OrderStatus;
```

```
}
                                                                            #include <atmi.h>
                                                   m_txn;
          public:
                                                                            #ifdef ICECAP
                    CTPCC_ODBC(LPCSTR szServer, LPCSTR szUser,
                                                                            // for IceCAP profiling
LPCSTR szPassword, LPCSTR szHost, LPCSTR szDatabase);
                                                                            #include <icapexp.h>
                     ~CTPCC ODBC(void);
                                                                            #endif
                    inline PNEW_ORDER_DATA
                                                                            // need to declare functions for export
BuffAddr NewOrder()
                                         { return &m_txn.NewOrder; };
                                                                            #define DllDecl __declspec( dllexport )
                     inline PPAYMENT DATA
BuffAddr_Payment()
                               { return &m_txn.Payment; };
                                                                            #include "..\..\common\src\trans.h"
                                                                                                                               //tpckit transaction
                    inline PDELIVERY_DATA
                                                                            header contains definations of structures specific to TPC-C
                                                                            #include "..\..\common\src\error.h"
BuffAddr Delivery()
                               { return &m txn.Delivery; };
                     inline PSTOCK_LEVEL_DATA BuffAddr_StockLevel()
                                                                            #include "..\..\common\src\txn_base.h"
                                                                            #include "tpcc tux.h"
{ return &m_txn.StockLevel; };
                                                                                                                                         // interface
                                                                            to Tuxedo libraries
                     inline PORDER STATUS DATA
BuffAddr OrderStatus()
                               { return &m txn.OrderStatus; };
                                                                            static TPINIT
                                                                                                                               *tpinf;
                    void NewOrder
                                                                            static DWORD
                                                                                                                               TLSIsTpInitedKey;
                                                   ();
                    void Payment
                                                                            static CRITICAL SECTION
                                                                                                                     TpCriticalSection;
                                                   ();
                     void Delivery
                                                   ();
                     void StockLevel
                                                   ():
                     void OrderStatus
                                         ();
                                                                            BOOL APIENTRY DllMain(HMODULE hModule, DWORD
                                                                            ul_reason_for_call, LPVOID lpReserved)
};
                                                                                      switch( ul_reason_for_call )
                                                                                                case DLL_PROCESS_ATTACH:
// wrapper routine for class constructor
extern "C" DllDecl CTPCC ODBC* CTPCC ODBC new
                                                                                                           DisableThreadLibraryCalls(hModule);
          ( LPCSTR szServer, LPCSTR szUser, LPCSTR szPassword,
LPCSTR szHost, LPCSTR szDatabase );
                                                                                                           // create thread local storage to determine
                                                                            Tuxedo initialization per thread.
typedef CTPCC ODBC* (TYPE CTPCC ODBC)(LPCSTR, LPCSTR,
                                                                                                           // it really should be possible to do this in the
LPCSTR, LPCSTR, LPCSTR);
                                                                            DLL THREAD ATTACH call, but
                                                                                                           // Ed says he could not get it to work.
                                                                                                           // assumption:value init'd to 0
                                                                                                           TLSIsTpInitedKey = TlsAlloc();
tpcc tux.cpp
                                                                                                           if ((tpinf = (TPINIT *)tpalloc("TPINIT",
          FILE:
                               TPCC TUX.CPP
                                                                            NULL, sizeof(TPINIT))) == NULL)
                                         Microsoft TPC-C Kit Ver.
4.20.000
                                                                                                                     // int TpRc = tperrno;
                                         Copyright Microsoft, 1999
                                                                                                                     return FALSE;
          All Rights Reserved
                                                                                                           tpinf->flags |= TPMULTICONTEXTS;
                                         Version 4.10.000 audited by
Richard Gimarc, Performance Metrics, 3/17/99
                                                                                                           InitializeCriticalSection(&TpCriticalSection);
                                                                                                           break;
                               Implementation for TPC-C Tuxedo class.
          Contact:
                    Charles Levine (clevine@microsoft.com)
                                                                                                case DLL PROCESS DETACH:
                                                                                                           TlsFree(TLSIsTpInitedKey);
   Change history:
                                                                                                           DeleteCriticalSection(&TpCriticalSection);
                    4.20.000 - updated rev number to match kit
                                                                                                           break:
                                                                                                default:
#include <windows.h>
                                                                                                           /* nothing */;
#include cess.h>
#include <stdio.h>
                                                                                      return TRUE;
#include <stdarg.h>
#include <malloc.h>
#include <stdlib.h>
                                                                            static void ThrTpInit()
#include <string.h>
#include <time.h>
                                                                                      static int num_tpinits=0;
#include <sys\timeb.h>
                                                                                      int iRc, TpRc;
#include <io.h>
#include <assert.h>
                                                                                      // has this thread been initialized? check thread local storage
                                                                                      if(!TlsGetValue(TLSIsTpInitedKey))
#include <tmenv.h>
#include <xa.h>
```

```
EnterCriticalSection(&TpCriticalSection);
                    itoa(++num tpinits, tpinf->cltname, 10);
                    iRc = tpinit(tpinf);
                    TpRc = tperrno;
                    LeaveCriticalSection(&TpCriticalSection);
                    if (iRc < 0)
                              throw new CTUXERR( tperrno );
                    int value = 1;
                    TlsSetValue(TLSIsTpInitedKey,&value);
          }
// wrapper routine for class constructor
  declspec(dllexport) CTPCC TUXEDO* CTPCC TUXEDO new()
          return new CTPCC_TUXEDO();
}
CTPCC TUXEDO::CTPCC TUXEDO()
                    Add initialization of Tuxedo Structures
          m_txn = (TUX_DATA *)tpalloc("CARRAY", NULL,
sizeof(TUX DATA));
          if (m txn == NULL)
                    throw new CTUXERR( tperrno );
}
CTPCC TUXEDO::~CTPCC TUXEDO()
          // free the data structure allocated with tpalloc
          tpfree((char *)m txn);
void CTPCC_TUXEDO::NewOrder()
                    ilen, *olen;
          long
          ThrTpInit();
          ilen = sizeof(TUX DATA);
          olen = &ilen;
          if (tpcall("NEWORDER", (char *)m_txn, ilen, (char **)&m_txn,
(long *)olen, TPSIGRSTRT) == -1)
                    throw new CTUXERR( tperrno );
          if ( m txn->ErrorType != ERR SUCCESS )
                    throw new CTUXERR( m_txn->ErrorType, m_txn->error
);
void CTPCC_TUXEDO::Payment()
                    ilen, *olen;
          long
          ThrTpInit();
          ilen = sizeof(TUX DATA);
          olen = &ilen;
          if (tpcall("PAYMENT", (char *)m_txn, ilen, (char **)&m_txn, (long
*)olen, TPSIGRSTRT) == -1)
                    throw new CTUXERR( tperrno );
          if ( m txn->ErrorType != ERR SUCCESS )
```

```
throw new CTUXERR( m_txn->ErrorType, m_txn->error
void CTPCC_TUXEDO::Delivery()
          int
                               iRc;
          long
                    ilen, *olen;
          // Note: Delivery txn code in the tuxedo server does not implement
logging of the delivery
               txn results, so cannot be used as is to run an auditable TPC-C
         //
result. For that
               reason, delivery txns should not be done via tuxedo.
         //
               The code is included for completeness.
          m_txn->u.Delivery.exec_status_code = eDeliveryFailed;
          return;
          // normal path...
          ThrTpInit();
          GetLocalTime(&m txn->u.Delivery.queue time);
          ilen = sizeof(TUX_DATA);
          olen = &ilen;
          if ((iRc = tpacall("DELIVERY", (char *)m_txn, ilen,
TPNOREPLY) = -1
                    int TpRc = tperrno;
                    m_txn->u.Delivery.exec_status_code = eDeliveryFailed;
          else
                    m_txn->u.Delivery.exec_status_code = eOK;
void CTPCC_TUXEDO::StockLevel()
          long
                    ilen, *olen;
          ThrTpInit();
          ilen = sizeof(TUX DATA);
          olen = &ilen;
          if (tpcall("STOCKLEVEL", (char *)m txn, ilen, (char **)&m txn,
(long *)olen, TPSIGRSTRT) == -1)
                    throw new CTUXERR( tperrno );
          if ( m_txn->ErrorType != ERR SUCCESS )
                    throw new CTUXERR( m_txn->ErrorType, m_txn->error
void CTPCC TUXEDO::OrderStatus()
          long
                    ilen, *olen;
          ThrTpInit();
          ilen = sizeof(TUX DATA);
          olen = &ilen;
          if (tpcall("ORDERSTATUS", (char *)m_txn, ilen, (char **)&m_txn,
(long *)olen, TPSIGRSTRT) == -1)
                    throw new CTUXERR( tperrno );
```

```
STOCK_LEVEL_DATA
          if ( m txn->ErrorType != ERR SUCCESS )
                                                                           StockLevel;
                    throw new CTUXERR( m txn->ErrorType, m txn->error
                                                                                                                    ORDER STATUS DATA
                                                                           OrderStatus;
);
}
                                                                                                          } u;
                                                                                               } *m txn;
char *CTUXERR::ErrorText()
                                                                                     public:
          if (m iErrorType == 0)
                                                                                               CTPCC TUXEDO();
                                                                                               ~CTPCC_TUXEDO(void);
                    if (m_errno == TPEOS)
                              sprintf( m_szErrorText, "Error: TUXEDO
                                                                                               inline PNEW ORDER DATA
error # %d, OS error # %d", m errno, m iError );
                                                                           BuffAddr NewOrder()
                                                                                                                    { return &m txn->u.NewOrder;
                    else
                              sprintf( m_szErrorText, "Error: TUXEDO
                                                                                               inline PPAYMENT DATA
error # %d", m_errno );
                                                                                                          { return &m_txn->u.Payment; };
                                                                           BuffAddr_Payment()
                                                                                               inline PDELIVERY_DATA
          else
                                                                           BuffAddr_Delivery()
                                                                                                          { return &m_txn->u.Delivery; };
                    sprintf( m_szErrorText, "Error: Class %d, error # %d",
                                                                                               inline PSTOCK_LEVEL_DATA BuffAddr_StockLevel()
m iErrorType, m iError);
                                                                           { return &m txn->u.StockLevel; };
                                                                                               inline PORDER_STATUS_DATA
          return m_szErrorText;
                                                                           BuffAddr OrderStatus()
                                                                                                          { return &m txn->u.OrderStatus; };
};
                                                                                               void NewOrder
                                                                                                                              ();
                                                                                               void Payment
                                                                                                                              ();
tpcc tux.h
                                                                                               void Delivery
                                                                                                                              ();
                                                                                               void StockLevel
                                                                                                                              ();
          FILE:
                              TPCC TUX.H
                                                                                               void OrderStatus
                                                                                                                    ();
                                         Microsoft TPC-C Kit Ver.
4.20.000
                                         Copyright Microsoft, 1999
          All Rights Reserved
                                                                           class CTUXERR: public CBaseErr
                                         Version 4.10.000 audited by
                                                                                     private:
Richard Gimarc, Performance Metrics, 3/17/99
                                                                                               // TODO: should use the sz Msg field of the base class
          PURPOSE:
                              Header file for TPC-C Tuxedo class
                                                                           instead
implementation.
                                                                                               char m szErrorText[64];
  Change history:
                                                                                     public:
                    4.20.000 - updated rev number to match kit
                                                                                               // use this interface for genuine Tuxedo errors
                                                                                               CTUXERR( int iErr )
#pragma once
                                                                                                          m_errno = iErr;
                                                                                                          m iErrorType = 0;
// need to declare functions for import, unless define has already been created
                                                                                                          m_iError = GetLastError();
                                                                                                                                        // only
// by the DLL's .cpp module for export.
                                                                           meaningful if m_errno == TPEOS
#ifndef DllDecl
                                                                                               };
#define DllDecl declspec( dllimport )
#endif
                                                                                               // use this interface to impersonate a non-Tuxedo error
                                                                           type
class DllDecl CTPCC_TUXEDO: public CTPCC_BASE
                                                                                               CTUXERR( int iErrorType, int iError )
          private:
                                                                                                          m iErrorType = iErrorType;
                    struct TUX_DATA
                                                                                                          m iError = iError;
                                                                                                          m = 0;
                              int
ErrorType;
                              int
                                                                                                                    m errno;
                                                                                               int
error;
                                                                                               int
                                                                                                                    m_iErrorType;
                                                                                                                    m iError;
                                                                                               int
                              union
                                                                                               // A CTUXERR class can impersonate another class,
                                         NEW ORDER DATA
                                                                           which happens if the error
NewOrder;
                                                                                               // was not actually a Tuxedo error, but was simply
                                         PAYMENT_DATA
                                                                           transmitted back via Tuxedo.
Payment;
                                                                                               int ErrorType()
                                         DELIVERY_DATA
Delivery;
```

```
if (m_iErrorType == 0)
                                                                        // when compiling with dblib, so redefined here. Note: we are using the symbol
                                       return ERR TYPE TUXEDO;
                                                                          SQLTYPES'
                             else
                                                                        // (declared in sqltypes.h) as a way to determine if TIMESTAMP STRUCT has
                                       return m_iErrorType;
                                                                        been declared.
                                                                        #ifndef SQLTYPES
                   }
                                                                                  typedef struct
                   int ErrorNum() {return m errno;};
                   char *ErrorText();
                                                                                                     short
};
                                                                        SQLSMALLINT */ year;
                                                                                                     unsigned short
                                                                                                                         /* SQLUSMALLINT
// wrapper routine for class constructor
                                                                                  month;
extern "C" __declspec(dllexport) CTPCC_TUXEDO*
                                                                                                                         /* SQLUSMALLINT
                                                                                                     unsigned short
CTPCC TUXEDO new();
                                                                                  day;
                                                                                                     unsigned short
                                                                                                                         /* SQLUSMALLINT
                                                                                  hour;
typedef CTPCC TUXEDO* (TYPE CTPCC TUXEDO)();
                                                                                                     unsigned short
                                                                                                                         /* SQLUSMALLINT
                                                                                 minute:
                                                                                                     unsigned short
                                                                                                                         /* SQLUSMALLINT
trans.h
                                                                                  second;
                                                                                                     unsigned long
                                                                                                                         /* SQLUINTEGER */
                                                                        fraction:
          FILE:
                             TRANS.H
                                                                                  } TIMESTAMP STRUCT;
                                       Microsoft TPC-C Kit Ver.
                                                                        #endif
4.20.000
                                       Copyright Microsoft, 1999
                                                                        // possible values for exec_status_code after transaction completes
         All Rights Reserved
                                                                        enum EXEC_STATUS
                                       Version 4.10.000 audited by
                                                                                                               // 0
                                                                                  eOK,
                                                                                                                         "Transaction
Richard Gimarc, Performance Metrics, 3/17/99
                                                                        commited."
                                                                                  eInvalidItem,
                                                                                                     // 1
                                                                                                               "Item number is not valid."
          PURPOSE:
                             Header file for TPC-C structure templates.
                                                                                  eDeliveryFailed
                                                                                                     //2
                                                                                                               "Delivery Post Failed."
  Change history:
                    4.20.000 - updated rev number to match kit
                                                                        // transaction structures
                                                                        typedef struct
#pragma once
                                                                                  // input params
// String length constants
                                                                                  short
                                                                                                                         ol_supply_w_id;
#define SERVER_NAME_LEN
                                  20
                                                                                  long
                                                                                                                         ol_i_id;
#define DATABASE NAME LEN
                                                                                  short
                                                                                                                         ol_quantity;
#define USER NAME LEN
                                 20
#define PASSWORD LEN
                                20
                                                                                 // output params
#define TABLE_NAME_LEN
                                  20
                                                                                  char
#define I_DATA_LEN
                             50
                                                                        ol_i_name[I_NAME_LEN+1];
#define I NAME LEN
                              24
                                                                                  char
#define BRAND_LEN
                              1
                                                                        ol_brand_generic[BRAND_LEN+1];
#define LAST NAME LEN
                                 16
                                                                                  double
                                                                                                                         ol i price;
#define W NAME LEN
                               10
                                                                                  double
                                                                                                                         ol amount;
#define ADDRESS LEN
                               20
                                                                                                                         ol_stock;
                                                                                  short
#define STATE_LEN
                             2
                                                                        } OL_NEW_ORDER_DATA;
#define ZIP LEN
#define S DIST LEN
                            24
                                                                        typedef struct
#define S_DATA_LEN
                              50
#define D NAME LEN
                               10
                                                                                  // input params
#define FIRST NAME LEN
                                 16
                                                                                  short
                                                                                                     w_id;
#define MIDDLE NAME LEN
                                                                                  short
                                                                                                     d_id;
#define PHONE LEN
                             16
                                                                                  long
                                                                                                     c id;
#define DATETIME_LEN
                               30
                                                                                  short
                                                                                                     o_ol_cnt;
#define CREDIT LEN
#define C_DATA_LEN
                              250
                                                                                  // output params
#define H DATA LEN
                              24
                                                                                  EXEC STATUS
                                                                                                                         exec status_code;
#define DIST_INFO_LEN
                               24
                                                                                  char
                                                                                                     c last[LAST NAME LEN+1];
#define MAX_OL_NEW_ORDER_ITEMS
                                                                                                     c_credit[CREDIT_LEN+1];
                                                                                  char
#define MAX_OL_ORDER_STATUS_ITEMS 15
                                                                                  double
                                                                                                     c_discount;
#define STATUS LEN
                              25
                                                                                  double
                                                                                                     w_tax;
#define OL DIST INFO LEN
                                         24
                                                                                  double
                                                                                                     d tax;
                                                                                  long
                                                                                                               o_id;
// TIMESTAMP STRUCT is provided by the ODBC header file sqltypes.h, but
                                                                                  short
                                                                                                                         o_commit_flag;
is not available
```

```
TIMESTAMP_STRUCT
                                      o_entry_d;
                                                                                 long
                                                                                              o id:
         short
                             o_all_local;
                                                                                 TIMESTAMP STRUCT
                                                                                                               o_entry_d;
         double
                             total amount;
                                                                                              o carrier id;
                                                                                 short
         OL NEW ORDER DATA
                                                                                 OL_ORDER_STATUS_DATA
OL[MAX OL NEW ORDER ITEMS];
                                                                       OL[MAX_OL_ORDER_STATUS_ITEMS];
NEW ORDER DATA, *PNEW ORDER DATA;
                                                                                 short
                                                                                              o ol cnt;
                                                                        ORDER STATUS DATA, *PORDER STATUS DATA;
typedef struct
         // input params
                                                                       typedef struct
         short
                                       w id;
         short
                                       d_id;
                                                                                 // input params
                                                                                                     w_id;
         long
                                       c_id;
                                                                                 short
         short
                                       c d id;
                                                                                 short
                                                                                                     o carrier id;
         short
                                       c_w_id;
         double
                                       h amount;
                                                                                 // output params
                             c\_last[LAST\_NAME\_LEN+1];
         char
                                                                                 EXEC STATUS
                                                                                                                        exec_status_code;
                                                                                 SYSTEMTIME
                                                                                                              queue_time;
         // output params
                                                                                                                        o_id[10];
                                                                                                                                            //
         EXEC STATUS
                                                                       id's of delivered orders for districts 1 to 10
                                                 exec_status_code;
         TIMESTAMP STRUCT
                                                                        } DELIVERY DATA, *PDELIVERY DATA;
         char
                                       w_street_1[ADDRESS_LEN+1];
                                       w street 2[ADDRESS LEN+1];
                                                                       //This structure is used for posting delivery transactions and for writing them to
         char
         char
                                       w_city[ADDRESS_LEN+1];
                                                                       the delivery server.
                                       w state[STATE LEN+1];
                                                                       typedef struct DELIVERY TRANSACTION
         char
         char
                                       w_zip[ZIP_LEN+1];
                                       d_street_1[ADDRESS_LEN+1];
                                                                                 SYSTEMTIME
                                                                                                                                  //time
         char
                                                                                                     queue;
         char
                                       d street 2[ADDRESS LEN+1];
                                                                       delivery transaction queued
                                       d_city[ADDRESS_LEN+1];
         char
                                                                                                     w_id;
                                                                                                                                  //delivery
                                                                                 short
         char
                                       d state[STATE LEN+1];
                                                                       warehouse
                               d_zip[ZIP_LEN+1];
         char
                                                                                 short
                                                                                                     o_carrier_id;
                                                                                                                        //carrier id
         char
                                       c first[FIRST NAME LEN+1];
                                                                        } DELIVERY_TRANSACTION;
                                       c_middle[MIDDLE_NAME_LEN
         char
+ 1];
                                                                       typedef struct
                                       c street 1[ADDRESS LEN+1];
                                                                                 // input params
                                       c_street_2[ADDRESS_LEN+1];
                                                                                 short
                                                                                                                        w_id;
         char
                                       c city[ADDRESS LEN+1];
                                                                                                                        d id;
         char
                                                                                 short
                                       c_state[STATE_LEN+1];
         char
                                                                                 short
                                                                                                                        threshold;
                                       c_zip[ZIP_LEN+1];
         char
                                       c_phone[PHONE_LEN+1];
         char
                                                                                 // output params
         TIMESTAMP_STRUCT
                                       c_since;
                                                                                 EXEC_STATUS
                                                                                                                        exec_status_code;
                                       c credit[CREDIT LEN+1];
                                                                                 long
                                                                                                                        low stock;
                                                                        } STOCK_LEVEL_DATA, *PSTOCK_LEVEL_DATA;
         double
                                       c_credit_lim;
         double
                                       c discount;
         double
                                       c balance;
         char
                                       c data[200+1];
                                                                        tuxapp.cpp
} PAYMENT_DATA, *PPAYMENT_DATA;
                                                                                 FILE:
                                                                                                     TUXAPP.CPP
typedef struct
                                                                                                              Microsoft TPC-C Kit Ver.
                                                                       4.20.000
         long
                                       ol i id;
                                                                                                              Copyright Microsoft, 1999
         short
                                       ol_supply_w_id;
                                                                                 All Rights Reserved
         short
                                       ol quantity;
         double
                                       ol_amount;
                                                                                                               Version 4.10.000 audited by
         TIMESTAMP STRUCT
                                       ol delivery d;
                                                                       Richard Gimarc, Performance Metrics, 3/17/99
} OL_ORDER_STATUS_DATA;
                                                                                 PURPOSE:
                                                                                                     Implementation for TPC-C Tuxedo server.
typedef struct
                                                                                 Contact: Charles Levine (clevine@microsoft.com)
         // input params
         short
                      w id;
                                                                          Change history:
         short
                      d_id;
                                                                                           4.20.000 - updated rev number to match kit
                      c_id;
         long
         char
                      c_last[LAST_NAME_LEN+1];
                                                                       #include <windows.h>
         // output params
                                                                       #include cess.h>
         EXEC_STATUS
                                                 exec_status_code;
                                                                       #include <tchar.h>
         char
                      c first[FIRST NAME LEN+1];
                                                                       #include <stdio.h>
                      c_middle[MIDDLE_NAME_LEN+1];
         char
                                                                       #include <stdarg.h>
         double
                       c balance;
                                                                       #include <iostream.h>
```

```
#include <malloc.h>
#include <stdlib.h>
#include <string.h>
#include <time.h>
#include <sys\timeb.h>
#include <io.h>
#include <assert.h>
#include <sqltypes.h>
#include <sql.h>
#include <sqlext.h>
#include <tmenv.h>
#include <xa h>
#include <atmi.h>
                                                   //tpckit transaction
#include "..\..\common\src\trans.h"
header contains definations of structures specific to TPC-C
#include "..\..\common\src\error.h"
#include "..\..\common\src\txn base.h"
#include "..\..\common\src\ReadRegistry.h"
#include "..\..\db dblib dll\src\tpcc dblib.h"
                                                   // DBLIB
implementation of TPC-C txns
#include "..\..\db odbc dll\src\tpcc odbc.h"
                                                   // ODBC
implementation of TPC-C txns
#include "tuxapp.h"
          szMyComputerName[MAX_COMPUTERNAME_LENGTH+1];
char
// configuration settings from registry
TPCCREGISTRYDATA
                                         Reg;
CTPCC_BASE
                              *pTxn = NULL;
#include "..\..\common\src\ReadRegistry.cpp"
/* FUNCTION: tpsvrinit ( int argc, char *argv[])
* PURPOSE:
                    Initialize the Server to Database connection.
* RETURNS:
                              int
                                                   Success
                                                             Failure
int tpsvrinit ( int argc, char *argv[] )
          try
                    DWORD dwSize =
MAX_COMPUTERNAME_LENGTH+1;
                    GetComputerName(szMyComputerName, &dwSize);
                    szMyComputerName[dwSize] = 0;
                    if ( ReadTPCCRegistrySettings( &Reg ) )
                              throw new CTUXAPP_ERR(
ERR MISSING REGISTRY ENTRIES);
                    GetParameters(argc, argv);
                    switch (Reg.eDB_Protocol)
                    case ODBC:
                              pTxn = new CTPCC_ODBC(
Reg.szDbServer, Reg.szDbUser, Reg.szDbPassword, szMyComputerName,
Reg.szDbName);
                              break:
                    case DBLIB:
```

```
pTxn = new CTPCC DBLIB(
Reg.szDbServer,\,Reg.szDbUser,\,Reg.szDbPassword,\,szMyComputerName,\,
Reg.szDbName);
                                break;
           }
           catch (CBaseErr *e)
                     WriteMessageToEventLog(e->ErrorText());
                     delete e;
          catch (...)
                     WriteMessageToEventLog(TEXT("Unhandled
exception."));
           return 0;
/* FUNCTION: tpsvrdone ( void )
*/
void tpsvrdone (void)
           delete pTxn;
          pTxn = NULL;
/* FUNCTION: BOOL GetParameters(int argc, char *argv[])
* PURPOSE:
                     This function parses the command line passed in to the
delivery executable, initializing
                                and filling in global variable parameters.
 * ARGUMENTS:
                    int
                                           argc
                                                      number of command
line arguments passed to delivery
                                           char
                                                      *argv[]
                                                                 array of
command line argument pointers
static void GetParameters(int argc, char *argv[])
           // advance through args until "--" is found
           for(int j=0; j<argc; j++)
                     if (strcmp(argv[j],"--") == 0)
                                break:
           for(int i=j+1; i<argc; i++)
                     if ( argv[i][0] == '-' || argv[i][0] == '/' )
                                switch(argv[i][1])
                                           case 'S':
                                                      strcpy(Reg.szDbServer,
argv[i]+2);
                                                      break;
                                           case 'D':
                                                      strcpy(Reg.szDbName,
argv[i]+2);
                                                      break;
                                           case 'P':
```

```
strcpy(Reg.szDbPassword, argv[i]+2);
                                                   break:
                                         case 'U':
                                                   strcpy(Reg.szDbUser,
argv[i]+2);
                                                   break;
                                         default:
                                                   cout << "Microsoft
TPC-C Kit" << endl;
                                                   cout << "Tuxedo
Server" << endl << endl:
                                                   cout << "Usage:" <<
endl;
                                                   cout << " tuxapp
[<tuxedo-args>] -- -S<sql-server> [-D<database>] [-U<user>] [-P<password>]"
<< endl << endl;
                                                   cout << "All parameters
default to values in registry." << endl;
                                                   throw new
CTUXAPP_ERR( ERR_BAD_SYNTAX );
          }
static void WriteMessageToEventLog(LPTSTR lpszMsg)
  TCHAR szMsg[256];
  HANDLE hEventSource;
  LPTSTR lpszStrings[2];
  // Use event logging to log the error.
  hEventSource = RegisterEventSource(NULL, TEXT("TUXAPP.EXE"));
   _stprintf(szMsg, TEXT("Error in TUXAPP.EXE: "));
  lpszStrings[0] = szMsg;
  lpszStrings[1] = lpszMsg;
  if (hEventSource != NULL)
    ReportEvent(hEventSource, // handle of event source
      EVENTLOG_ERROR_TYPE, // event type
                   // event category
      0,
                   // event ID
      0,
      NULL,
                       // current user's SID
                   // strings in lpszStrings
                   // no bytes of raw data
      (LPCTSTR *)lpszStrings, // array of error strings
      NULL):
                       // no raw data
    (VOID) DeregisterEventSource(hEventSource);
void NEWORDER( TPSVCINFO *rqst )
          PNEW ORDER DATA
                                         pNewOrder;
          TUX_DATA
                                         *pData;
          const int
                              iSize = sizeof(pData->u.NewOrder);
          try
                    pData = (TUX_DATA*)rqst->data;
                    pData->retval = ERR SUCCESS;
                    pData->error = 0;
```

```
pNewOrder = pTxn->BuffAddr NewOrder();
                    assert( rqst->len == sizeof(TUX DATA) );
                    memcpy(pNewOrder, &pData->u.NewOrder, iSize);
                    pTxn->NewOrder();
                    memcpy( &pData->u.NewOrder, pNewOrder, iSize );
                    tpreturn( TPSUCCESS, 0, rqst->data,
sizeof(TUX DATA), 0);
          catch (CBaseErr *e)
                    pData->retval = e->ErrorType();
                    pData->error = e->ErrorNum();
                    memcpy( &pData->u.NewOrder, pNewOrder, iSize );
                    tpreturn( TPSUCCESS, 0, rqst->data,
sizeof(TUX DATA), 0);
                    delete e;
          catch (...)
                    WriteMessageToEventLog(TEXT("Unhandled
exception."));
                    pData->retval = ERR TYPE LOGIC;
                    pData - > error = 0;
                    memcpy( &pData->u.NewOrder, pNewOrder, iSize );
                    tpreturn( TPSUCCESS, 0, rqst->data,
sizeof(TUX DATA), 0);
void PAYMENT( TPSVCINFO *rqst )
          PPAYMENT_DATA pPayment;
          TUX DATA
                                        *pData;
          const int
                              iSize = sizeof(pData->u.Payment);
          try
                    pData = (TUX_DATA*)rqst->data;
                    pData->retval = ERR_SUCCESS;
                    pData - > error = 0;
                    pPayment = pTxn->BuffAddr Payment();
                    assert( rqst->len == sizeof(TUX_DATA) );
                    memcpy(pPayment, &pData->u.Payment, iSize);
                    pTxn->Payment();
                    memcpy(&pData->u.Payment, pPayment, iSize);
                    tpreturn( TPSUCCESS, 0, rqst->data,
sizeof(TUX DATA), 0);
          catch (CBaseErr *e)
                    pData->retval = e->ErrorType();
                    pData - error = e - ErrorNum();
                    memcpy( &pData->u.Payment, pPayment, iSize );
                    tpreturn( TPSUCCESS, 0, rqst->data,
sizeof(TUX_DATA), 0);
                    delete e;
          catch (...)
                    WriteMessageToEventLog(TEXT("Unhandled
exception."));
                    pData->retval = ERR_TYPE_LOGIC;
                    pData - > error = 0;
                    memcpy( &pData->u.Payment, pPayment, iSize );
```

```
tpreturn( TPSUCCESS, 0, rqst->data,
sizeof(TUX_DATA), 0);
// Note: Delivery txn code below does not implement logging of the delivery
     txn results, so cannot be used as is to run an auditable TPC-C result.
     The code is included for completeness.
void DELIVERY( TPSVCINFO *rqst )
          PDELIVERY_DATA pDelivery;
          TUX DATA
                                         *pData;
          const int
                               iSize = sizeof(pData->u.Delivery);
          try
          {
                     pData = (TUX DATA*)rqst->data;
                    pData->retval = ERR_SUCCESS;
                    pData->error = 0;
                    pDelivery = pTxn->BuffAddr_Delivery();
                    assert( rqst->len == sizeof(TUX DATA) );
                    memcpy (pDelivery, \&pData->u.Delivery, iSize~);\\
                    pTxn->Delivery();
                    memcpy(&pData->u.Delivery, pDelivery, iSize);
                    tpreturn( TPSUCCESS, 0, rqst->data,
sizeof(TUX DATA), 0);
          catch (CBaseErr *e)
                    pData->retval = e->ErrorType();
                    pData->error = e->ErrorNum();
                    memcpy( &pData->u.Delivery, pDelivery, iSize );
                     tpreturn( TPSUCCESS, 0, rqst->data,
sizeof(TUX_DATA), 0);
                    delete e;
          catch (...)
          {
                     WriteMessageToEventLog(TEXT("Unhandled
exception."));
                     pData->retval = ERR_TYPE_LOGIC;
                    pData - > error = 0;
                    memcpy( &pData->u.Delivery, pDelivery, iSize );
                     tpreturn( TPSUCCESS, 0, rqst->data,
sizeof(TUX DATA), 0);
          }
void STOCKLEVEL( TPSVCINFO *rqst )
          PSTOCK LEVEL DATA
                                         pStockLevel;
          TUX DATA
                                                    *pData;
          const int
                                         iSize =
sizeof(pData->u.StockLevel);
          try
                    pData = (TUX DATA*)rgst->data;
                    pData->retval = ERR SUCCESS;
                    pData->error=0;
                    pStockLevel = pTxn->BuffAddr_StockLevel();
                    assert( rqst->len == sizeof(TUX DATA) );
                    memcpy(pStockLevel, &pData->u.StockLevel, iSize );
```

```
pTxn->StockLevel();
                    memcpy( &pData->u.StockLevel, pStockLevel, iSize );
                    tpreturn( TPSUCCESS, 0, rqst->data,
sizeof(TUX DATA), 0);
          catch (CBaseErr *e)
                    pData->retval = e->ErrorType();
                    pData->error = e->ErrorNum();
                    memcpy(&pData->u.StockLevel, pStockLevel, iSize);
                    tpreturn( TPSUCCESS, 0, rqst->data,
sizeof(TUX_DATA), 0);
                    delete e;
          catch (...)
                    WriteMessageToEventLog(TEXT("Unhandled
exception."));
                    pData->retval = ERR_TYPE_LOGIC;
                    pData - > error = 0;
                    memcpy( &pData->u.StockLevel, pStockLevel, iSize );
                    tpreturn( TPSUCCESS, 0, rqst->data,
sizeof(TUX DATA), 0);
          }
void ORDERSTATUS( TPSVCINFO *rqst )
          PORDER STATUS DATA
                                         pOrderStatus;
          TUX DATA
                                         *pData;
          const int iSize = sizeof(pData->u.OrderStatus);
          try
                    pData = (TUX_DATA*)rqst->data;
                    pData->retval = ERR SUCCESS;
                    pData - error = 0;
                    pOrderStatus = pTxn->BuffAddr_OrderStatus();
                    assert( rqst->len == sizeof(TUX DATA) );
                    memcpy(pOrderStatus, &pData->u.OrderStatus, iSize);
                    pTxn->OrderStatus();
                    memcpy( &pData->u.OrderStatus, pOrderStatus, iSize );
                    tpreturn( TPSUCCESS, 0, rqst->data,
sizeof(TUX_DATA), 0);
          catch (CBaseErr *e)
                    pData->retval = e->ErrorType();
                    pData->error = e->ErrorNum();
                    memcpy(&pData->u.OrderStatus, pOrderStatus, iSize);
                    tpreturn( TPSUCCESS, 0, rqst->data,
sizeof(TUX DATA), 0);
                    delete e;
          catch (...)
                    WriteMessageToEventLog(TEXT("Unhandled
exception."));
                    pData->retval = ERR TYPE LOGIC;
                    pData - > error = 0;
                    memcpy(&pData->u.OrderStatus, pOrderStatus, iSize);
                    tpreturn( TPSUCCESS, 0, rqst->data,
sizeof(TUX_DATA), 0);
          }
```

```
/* FUNCTION: CTUXAPP_ERR::ErrorText
*/
char* CTUXAPP_ERR::ErrorText(void)
          int i;
          static SERRORMSG errorMsgs[] =
                    { ERR_MISSING_REGISTRY_ENTRIES, "Required
entries missing from registry.
                    { ERR BAD SYNTAX,
"Syntax error in input parameters."
                    { ERR_UNKNOWN_DB_PROTOCOL,
"Unknown database protocol specified in registry."
                                                            }
          };
          static char szNotFound[] = "Unknown error number.";
          for(i=0; errorMsgs[i].szMsg[0]; i++)
                    if ( m Error == errorMsgs[i].iError )
                              break:
          if ( !errorMsgs[i].szMsg[0] )
                   return szNotFound;
          else
                    return errorMsgs[i].szMsg;
}
tuxapp.h
```

```
FILE:
                            TUXAPP.H
                                      Microsoft TPC-C Kit Ver.
4.20.000
                                      Copyright Microsoft, 1999
         All Rights Reserved
                                      Version 4.10.000 audited by
Richard Gimarc, Performance Metrics, 3/17/99
         PURPOSE:
                            Header file for TPC-C Tuxedo server.
  Change history:
                   4.20.000 - updated rev number to match kit
enum TUXERROR
         ERR MISSING REGISTRY ENTRIES = 1,
         ERR BAD SYNTAX,
         ERR_UNKNOWN_DB_PROTOCOL
};
class CTUXAPP_ERR: public CBaseErr
         public:
                   TUXERROR m_Error;
                  CTUXAPP ERR(TUXERROR Err) {
                                                         m Error =
Err; };
                  ~CTUXAPP_ERR() {};
```

```
int ErrorType() {return ERR_TYPE_TUXEDO;};
                   int ErrorNum() {return m Error;};
                   char *ErrorText();
};
struct TUX DATA
          int
                                                           retval;
         int
                                                           error:
          union
                   NEW_ORDER_DATA
                                                           NewOrder;
                   PAYMENT DATA
                                                 Payment;
                   \overline{DELIVERY\_DATA}
                                                 Delivery;
                   STOCK LEVEL DATA
                                                 StockLevel;
                   ORDER_STATUS_DATA
                                                 OrderStatus;
         } u;
};
static void GetParameters(int argc, char *argv[]);
static void WriteMessageToEventLog(LPTSTR lpszMsg);
#if defined(__cplusplus)
extern "C" {
#endif
void NEWORDER( TPSVCINFO *rqst );
void PAYMENT( TPSVCINFO *rqst );
void DELIVERY( TPSVCINFO *rqst );
void STOCKLEVEL( TPSVCINFO *rqst );
void ORDERSTATUS( TPSVCINFO *rqst );
#if defined(__cplusplus)
#endif
```

tuxmain.c

```
TUXMAIN.C
          FILE:
                                         Microsoft TPC-C Kit Ver.
4.20.000
                                         Copyright Microsoft, 1999
          All Rights Reserved
                                         Version 4.10.000 audited by
Richard Gimarc, Performance Metrics, 3/17/99
          PURPOSE:
                               Implementation for TPC-C Tuxedo server.
          Contact: Charles Levine (clevine@microsoft.com)
   Change history:
                    4.20.000 - updated rev number to match kit
#include <stdio.h>
#include <xa.h>
#include <atmi.h>
#if defined( cplusplus)
extern "C" {
#endif
extern int _tmrunserver _((int));
extern void DELIVERY ((TPSVCINFO *));
extern void NEWORDER _((TPSVCINFO *));
extern void ORDERSTATUS _((TPSVCINFO *));
```

```
extern void PAYMENT ((TPSVCINFO *));
                                                                                   FILE:
                                                                                                        TXN BASE.H
extern void STOCKLEVEL ((TPSVCINFO *));
                                                                                                                  Microsoft TPC-C Kit Ver.
                                                                          4.20.000
#if defined(__cplusplus)
                                                                                                                  Copyright Microsoft, 1999
#endif
                                                                                   All Rights Reserved
static struct tmdsptchtbl_t _tmdsptchtbl[] = {
                                                                                                                  Version 4.10.000 audited by
          { "DELIVERY", "DELIVERY", (void (*) _((TPSVCINFO *)))
                                                                          Richard Gimarc, Performance Metrics, 3/17/99
DELIVERY, 0, 0 },
          { "NEWORDER", "NEWORDER", (void (*) ((TPSVCINFO *)))
                                                                                    PURPOSE:
                                                                                                        Header file for TPC-C txn class
                                                                          implementation.
NEWORDER, 1, 0 },
          { "ORDERSTATUS", "ORDERSTATUS", (void (*)
((TPSVCINFO *))) ORDERSTATUS, 2, 0 },
                                                                            Change history:
          { "PAYMENT", "PAYMENT", (void (*) _((TPSVCINFO *)))
                                                                                              4.20.000 - updated rev number to match kit
PAYMENT, 3, 0 },
          { "STOCKLEVEL", "STOCKLEVEL", (void (*) _((TPSVCINFO
*))) STOCKLEVEL, 4, 0 },
                                                                          #pragma once
          { NULL, NULL, NULL, 0, 0 }
                                                                          // need to declare functions for import, unless define has already been created
                                                                          // by the DLL's .cpp module for export.
#ifndef _TMDLLIMPORT
                                                                          #ifndef DllDecl
#define _TMDLLIMPORT
                                                                          #define DllDecl declspec( dllimport )
#endif
                                                                          #endif
_TMDLLIMPORT extern struct xa_switch_t tmnull_switch;
                                                                          class DllDecl CTPCC_BASE
struct tmsvrargs t tmsvrargs = {
                                                                                    public:
          NULL,
                                                                                             CTPCC_BASE(void) {};
          & tmdsptchtbl[0],
                                                                                             virtual ~CTPCC BASE(void) {};
          tpsvrinit,
                                                                                             virtual PNEW ORDER DATA
          tpsvrdone,
                                                                          BuffAddr_NewOrder()
                                                                                              virtual PPAYMENT_DATA
                              /* PRIVATE */
          tmrunserver.
          NULL,
                                                                                                       = 0;
                                        /* RESERVED */
                                                                          BuffAddr Payment()
          NULL,
                                        /* RESERVED */
                                                                                              virtual PDELIVERY_DATA
          NULL,
                                        /* RESERVED */
                                                                          BuffAddr_Delivery()
                                        /* RESERVED */
                                                                                              virtual PSTOCK_LEVEL_DATA
          NULL
                                                                          BuffAddr StockLevel()
};
                                                                                                        = 0;
                                                                                              virtual PORDER_STATUS_DATA
struct tmsvrargs t*
                                                                          BuffAddr_OrderStatus()
                                                                                                        = 0:
#ifdef TMPROTOTYPES
                                                                                                                                      () = 0;
_tmgetsvrargs(void)
                                                                                              virtual void NewOrder
                                                                                              virtual void Payment
                                                                                                                            () = 0;
#else
 _tmgetsvrargs()
                                                                                             virtual void Delivery
                                                                                                                            () = 0;
                                                                                             virtual void StockLevel
#endif
                                                                                                                                      () = 0;
                                                                                             virtual void OrderStatus
                                                                                                                            () = 0:
          tmsvrargs.xa switch = &tmnull switch;
                                                                          };
          return(&tmsvrargs);
}
                                                                          txnlog.h
#ifdef TMPROTOTYPES
                                                                                    FILE:
                                                                                                        TXNLOG.H
main(int argc, char **argv)
                                                                                                                  Microsoft TPC-C Kit Ver.
#else
                                                                          4.10.000
main(argc,argv)
int argc;
                                                                                                                  NOTE: this file is RTE specific
char **argv;
                                                                          and should not be included
#endif
                                                                                                                  in Full Disclosure Reports.
#ifdef TMMAINEXIT
                                                                                                                  Copyright Microsoft, 1999
#include "mainexit.h"
#endif
                                                                                   PURPOSE:
                                                                                                        Structure definitions for logging delivery txn
                                                                          completion stats.
          return( _tmstartserver( argc, argv, _tmgetsvrargs()));
                                                                                    Contact: Charles Levine (clevine@microsoft.com)
txn base.h
                                                                            typedef struct _TXN_NEWORDER
```

```
BYTE
                  OL Count;
                                  //range 0 to 31
    BYTE
                  OL Remote Count; //range 0 to 31
    WORD
                                                                        //Graphically:
                  c_id;
    int
                  o id:
         } TXN NEWORDER;
                                                                        // time -->
                                                                        //
 typedef struct _TXN_PAYMENT
                                                                        //
    BYTE
                   CustByName;
                                                                        //
    BYTE
                   IsRemote;
                                                                        //
         } TXN PAYMENT;
                                                                        //
 typedef struct _TXN_ORDERSTATUS
                   BYTE
                            CustByName;
         } TXN ORDERSTATUS;
         typedef union _TXN_DETAILS
                                                                        //reflect this.
                   TXN NEWORDER NewOrder;
                   TXN PAYMENT
                                               Payment;
                  TXN ORDERSTATUS
                                               OrderStatus;
         } TXN DETAILS;
                                                                      TXN RECORD HEADER
                                                                      start of txn
         // Common header for all records in txn log. The TxnType field is
                                                                                         BYTE
         // a switch which identifies the particular variant.
                                                                      TXN_REC_TYPE_TPCC
         #define TXN REC TYPE CONTROL
                                                                                         BYTE
                                                                      depends on TxnType
         #define TXN_REC_TYPE_TPCC
                                                                   2
// replaces TRANSACTION_TYPE_TPCC
         #define TXN_REC_TYPE_TPCC_DELIV_DEF
                                                         3
                                                                                         int
                                                                                         int
         typedef struct _TXN_RECORD_HEADER
                                                                                         int
                                                                                         int
                  JULIAN TIME
                                                                  //
                                      TxnStartT0;
                                                                                         int
start of txn
                                                                      (ms)
                   BYTE
                            TxnType;
                                                         // one of
                                                                                         int
TXN_REC_TYPE_*
                            TxnSubType;
depends on TxnType
         TXN RECORD HEADER, *PTXN RECORD HEADER;
                                                                                         BYTE
                                                                      district ID for this thread
                                                                                         BYTE
         typedef struct _TXN_RECORD_CONTROL
                                                                      this particular
                                                                                         BYTE
                   // common header; must exactly match
                                                                      txn to indicate errors
TXN_RECORD_HEADER
                                                                                         BYTE
                  JULIAN TIME
                                                                  //
                                      TxnStartT0;
start of txn
                            TxnType;
                   BYTE
                                                         // =
TXN_REC_TYPE_CONTROL
                   BYTE
                            TxnSubType;
                                                                   //
depends on TxnType
                  // end of common header
                                                                      bytes.
                                                                   //
                  DWORD Len:
number of bytes after this field
         } TXN_RECORD_CONTROL, *PTXN_RECORD_CONTROL;
                                                                      TXN RECORD HEADER
         // TPC-C Txn Record Layout:
 //'TxnStartT0' is a Julian timestamp corresponding to the moment the
                                                                      start of txn
 //txn is sent to the SUT, i.e., beginning of response time. Deltas
                                                                                         BYTE
 //are in milliseconds. Note that if RTDelay > 0, then the txn was
 //delayed by this amount. The delay occurs at the beginning of the
```

```
//response time. So if RTDelay > 0, then the txn was actually sent
  //at TxnStartT0 + RTDelay.
      |--- Menu --- |-- Keying -- |-- Response -- |--- Think -- |
      <- DeltaT1 -> <- DeltaT2 -> <- DeltaT4 -> <- DeltaT3 ->
                      ^{\wedge}\,TxnStartT0
  //RTDelay is the amount of response time delay included in DeltaT4.
  //RTDelay is recorded per txn because this value can be changed on
  //the fly, and so may vary from txn to txn.
  //TxnStatus is the txn completion code. It is used to indicate errors.
  //For example, in the New Order txn, 1% of txns abort. TxnStatus will
          typedef struct _TXN_RECORD_TPCC
                     // common header; must exactly match
                     JULIAN TIME
                                           TxnStartT0;
                                                                          //
                                TxnType;
                                                                //=
                                TxnSubType;
                                                                          //
                     // end of common header
                          DeltaT1;
                                                     // menu time (ms)
                          DeltaT2;
                                                     // keying time (ms)
                          DeltaT3;
                                                     // think time (ms)
                                                     // response time (ms)
                          DeltaT4;
                                                     // response time delay
                          RTDelay:
                                          TxnError:
                                                                // error code
providing more detail for TxnStatus
                     WORD
                                                                // warehouse
                                w id;
                                                                // assigned
                                d id;
                                d_id_ThisTxn;
                                                     // district ID chosen for
                                TxnStatus;
                                                     // completion status for
                                                     // for word alignment
                               reserved:
                     TXN DETAILS
                                          TxnDetails;
          } TXN_RECORD_TPCC, *PTXN_RECORD_TPCC;
          // TPC-C Deferred Delivery Txn Record Layout:
  //Incorporating delivery transaction information into the above
          //structure would increase the size of TXN_DETAILS from 8 to 42
  //Hence, we store delivery transaction details in a separate structure.
          typedef struct _TXN_RECORD_TPCC_DELIV_DEF
                     // common header; must exactly match
                     JULIAN_TIME
                                          TxnStartT0;
                                                                          //
                                TxnType;
                                                                //=
TXN_REC_TYPE_TPCC_DELIV_DEF
```

```
BYTE
                               TxnSubType;
                                                                                     MAX_NUM_BUFFERS
                                                                           #define
                                                                                                                                         2
=0
                    // end of common header
                                                                           // flags passed in to the constructor
                                                                                      TXN LOG WRITE
                                                                           #define
                                                                                                                     0x01
                                                   // response time (ms)
                                                                           #define
                                                                                      TXN LOG READ
                                                                                                                    0x02
                     int
                         DeltaT4:
                                         DeltaTxnExec;
                                                              // execution
                                                                           #define
                                                                                      TXN LOG SORTED
                                                                                                                     0x04
time (ms)
                                                                                      TXN LOG OS ERROR
                     WORD
                               w id;
                                                              // warehouse
                                                                           #define
                                                                                                                     1
ID
                                                                           #define
                                                                                      TXN LOG NOT SORTED
                                                                                                                    2
                     BYTE
                               TxnStatus;
                                                   // completion status for
                                                                           #define SKIP_CTRL_RECS
txn to indicate errors
                     BYTE
                                                   // for word alignment
                               reserved:
                     short o carrier id;
                                        // carrier id
                                                                           class CTxnLog
                               o_id[10];
                                                   // returned delivery
                     long
transaction ids
                                                                                      private:
          } TXN_RECORD_TPCC_DELIV_DEF,
                                                                                                DWORD
                                                                                                                     iBufferSize;
*PTXN RECORD TPCC DELIV DEF;
                                                                            //buffer allocated size
                                                                                                DWORD
                                                                                                                     iBytesFreeInBuffer;
                                                                           //total bytes available for use in buffer
                                                                                                                               iNumBuffers;
#define
                    TXN_LOG_VERSION
                                                                                      //buffers in use
#define
                    TXN DATA START
                                                   4096
                                                              // offset in
                                                                                                                               iActiveBuffer;
log file where log records start
                                                                                      //indicates which buffer is active: 0 or 1
#define
                    TXN LOG EYE CATCHER
                                                   "BC"
                                                              // signature
                                                                                                                               iIoBuffer;
                                                                                                int
bytes at the start of log file
                                                                                      //buffer for any pending IO operation
                                                                                                                               iFilePointer:
                                                                                                int
                                                                                      //position in file.
          iNextRec;
                                                                                                int
          // The transaction log has a header as the first 4K block.
                                                                                      //when reading, ordinal value of next record
          typedef struct _TXN_LOG HEADER
                                                                                                // A "save point" is remembered each time
                                                                            GetNextRecord is called with a start time specified.
                                                   EyeCatcher[2];
                                                                                                // The next time it is called, if start time is after the save
                    char
signature bytes; should always be "BC"
                                                                           point, we start scanning from the
                                                              LogVersion;
                                                                                                // save point. This is particularly useful in
                    int
          // set to TXN_LOG_VERSION
                                                                           FindBestInterval, where the log is scanned repeatedly.
                     JULIAN_ TIME
                                                   BeginTxnTS;
                                                                                                JULIAN TIME
                                                                                                                               SavePtTime;
// timestamp of first (lowest) txn start
                                                                                                int
                    JULIAN_TIME
                                                   EndTxnTS;
                                                                            iSavePtFilePointer;
// timestamp of last (highest) txn completion time
                                                                                                int
                                                              iRecCount;
                                                                           iSavePtNextRec;
          // number of records in log file
                     BOOL
                                                   bLogSorted;
                                                                                                JULIAN TIME
                                                                                                                    lastTS;
                                                              iFileSize;
                                                                                      //when writing sorted output, used to verify records are sorted
                     int
// file size in bytes
                                                                                                                     bWrite;
                                                                                      //writing log file
                    // the record map provides a fast way to get close to a
particular timestamp in a sorted log file.
                                                                                                BOOL
                                                                                                                               bLogSorted;
                                                                                      // is log file sorted? applies to both input and output
                     struct
                                                                                                JULIAN TIME
//
                                                                                                                               BeginTxnTS;
                                                                                      // timestamp of first (lowest) txn start
//
                               JULIAN TIME
                                                              TS;
                                                                                                JULIAN TIME
                                                                                                                               EndTxnTS;
          // timestamp of record
//
                                                                                      // timestamp of last (highest) txn completion time
iPos;
                     // byte position in file
                                                                                                int
                                                                                                                                         iRecCount;
                                                                                                // number of records in log file
RecMap[RecMapSize];
//#define
                     RecMapSize
                                                              200
                                                                                                                     *pCurrent;
                                                                            //ptr to current buffer
          } TXN LOG HEADER, *PTXN LOG HEADER;
                                                                            *pBuffer[MAX_NUM_BUFFERS];
                                                                                                PTXN RECORD HEADER
                                                                                                                               *TxnArray;
                                                                           //transaction record pointer array for sort
#define
          READ_BUFFER_SIZE
                                                   64*1024
          WRITE_BUFFER_SIZE
                                                                                                DWORD
                                                    8*1024
                                                                                                                     dwError:
#define
                                                                                                HANDLE
                                                                                                                     hTxnFile;
#define
          NUM_READ_BUFFERS
                                                                           //handle to log file
                                                   1
#define
          NUM WRITE BUFFERS
                                                   2
```

```
hMapFile;
                   HANDLE
//map file used when sorting the log
                                                                                          int ErrorType() {return ERR TYPE TXNLOG;};
                                      hIoComplete;
                   HANDLE
                                                                                          int ErrorNum() {return m errno;};
//event to signify that there are no pending IOs
                   HANDLE
                                      hLogFileIo;
                                                                                          // TODO: need to complete...
         //event to signal the IO thread to write the inactive buffer
                                                                                          char *ErrorText() {return "";};
                   Spinlock Spin;
//spin lock to protect the txn log file buffers
                                                                       webcInt.dsp
                   int Write(BYTE *ptr, DWORD Size);
                   static void LogFileIO(CTxnLog *);
                                                                       # Microsoft Developer Studio Project File - Name="webclnt" - Package
                                                                       Owner=<4>
         public:
                                                                       # Microsoft Developer Studio Generated Build File, Format Version 5.00
                   CTxnLog::CTxnLog(LPCTSTR szFileName, DWORD
                                                                       # ** DO NOT EDIT **
dwOpts);
                   ~CTxnLog(void);
                                                                       #TARGTYPE "Win32 (x86) Application" 0x0101
                   int WriteToLog(PTXN RECORD TPCC pTxnRcrd);
                                                                       CFG=webclnt - Win32 Release
                   int WriteToLog(PTXN RECORD TPCC DELIV DEF
                                                                       !MESSAGE This is not a valid makefile. To build this project using NMAKE,
pTxnRcrd);
                                                                       !MESSAGE use the Export Makefile command and run
                   int WriteToLog(PTXN RECORD CONTROL
                                                                       !MESSAGE
pCtrlRec);
                                                                       !MESSAGE NMAKE /f "Webclnt.mak".
                   int WriteToLog(PTXN RECORD HEADER pCtrlRec);
                                                                       !MESSAGE
                                                                       !MESSAGE You can specify a configuration when running NMAKE
                   int WriteCtrlRecToLog(BYTE SubType, LPTSTR lpStr,
                                                                       !MESSAGE by defining the macro CFG on the command line. For example:
DWORD dwLen);
                                                                       !MESSAGE
                                                                       !MESSAGE NMAKE /f "Webclnt.mak" CFG="webclnt - Win32 Release"
                   void CloseTransactionLogFile(void);
                                                                       !MESSAGE
                                                                       !MESSAGE Possible choices for configuration are:
                   PTXN RECORD HEADER GetNextRecord(BOOL
                                                                       !MESSAGE
bSkipCtrlRecs = FALSE);
                                                                       !MESSAGE "webclnt - Win32 Release" (based on "Win32 (x86) Application")
                   PTXN RECORD HEADER
                                                                       !MESSAGE "webclnt - Win32 Debug" (based on "Win32 (x86) Application")
GetNextRecord(JULIAN TIME SeekTimeT0, BOOL bSkipCtrlRecs =
                                                                       !MESSAGE
FALSE);
                                                                       # Begin Project
                   int Sort(void);
                                                                       # PROP Scc_ProjName ""
                   PTXN RECORD HEADER GetSortedRecord(int
                                                                       # PROP Scc LocalPath ""
index);
                                                                       CPP=cl.exe
                                                                       MTL=midl.exe
                   inline BOOL IsSorted(void) { return bLogSorted; };
                                                                       RSC=rc.exe
                   inline JULIAN_TIME BeginTS(void) { return
BeginTxnTS; };
                                                                       !IF "$(CFG)" == "webclnt - Win32 Release"
                   inline JULIAN TIME EndTS(void) { return EndTxnTS;
};
                                                                       # PROP BASE Use MFC 0
                   inline int RecordCount(void) { return iRecCount; };
                                                                       # PROP BASE Use Debug Libraries 0
}:
                                                                       # PROP BASE Output_Dir ".\Release"
                                                                       # PROP BASE Intermediate_Dir ".\Release"
class CTXNLOG_ERR: public CBaseErr
                                                                       # PROP BASE Target Dir ""
                                                                       # PROP Use MFC 0
         public:
                                                                       # PROP Use_Debug_Libraries 0
                   enum CTPCC DBLIB ERRS
                                                                       # PROP Output Dir ".\Release"
                                                                       # PROP Intermediate_Dir ".\Release"
                             ERR BAD FILE FORMAT = 1,
                                                                       # PROP Target Dir ""
"File format is invalid."
                                                                       # ADD BASE CPP /nologo /W3 /GX /O2 /D "WIN32" /D "NDEBUG" /D
                             ERR_UNKNOWN_LOG_VERSION,
                                                                         WINDOWS" /YX /c
"Log file version is unknown."
                                                                       #ADD CPP /nologo /W3 /GX /O2 /D "WIN32" /D "NDEBUG" /D
                             ERR_BROKEN_LOG_FILE,
                                                                        WINDOWS" /YX /FD /c
"Log file is broken."
                                                                       # ADD BASE MTL /nologo /D "NDEBUG" /win32
                             ERR_LOG_NOT_SORTED,
                                                                       # ADD MTL /nologo /D "NDEBUG" /mktyplib203 /win32
// "Log file is not sorted"
                                                                       # ADD BASE RSC /I 0x409 /d "NDEBUG"
                             ERR INVALID TIME SEQ,
                                                                       # ADD RSC /I 0x409 /d "NDEBUG"
"Internal Error: Record Time Sequence invalid."
                                                                       BSC32=bscmake.exe
                                                                       # ADD BASE BSC32 /nologo
                                                                       # ADD BSC32 /nologo
                   CTXNLOG ERR( int iErr ) { m errno = iErr; };
                                                                       LINK32=link.exe
                                       m errno;
```

```
# ADD BASE LINK32 kernel32.lib user32.lib gdi32.lib winspool.lib
comdlg32.lib advapi32.lib shell32.lib ole32.lib oleaut32.lib uuid.lib odbc32.lib
                                                                  Project: "db odbc dll"=.\db odbc dll\db odbc dll.dsp - Package Owner=<4>
odbccp32.lib /nologo /subsystem:windows /machine:I386
# ADD LINK32 kernel32.lib user32.lib gdi32.lib winspool.lib comdlg32.lib
                                                                  Package=<5>
advapi32.lib shell32.lib ole32.lib oleaut32.lib uuid.lib odbc32.lib odbccp32.lib
                                                                  -{{{
/nologo/subsystem:windows/machine:I386
                                                                  }}}
!ELSEIF "$(CFG)" == "webclnt - Win32 Debug"
                                                                  Package=<4>
                                                                  {{{
# PROP BASE Use MFC 0
                                                                  }}}
# PROP BASE Use_Debug_Libraries 1
# PROP BASE Output_Dir ".\Debug"
                                                                  # PROP BASE Intermediate Dir ".\Debug"
                                                                  # PROP BASE Target_Dir ""
# PROP Use MFC 0
                                                                  Project: "install"=\install\install.dsp - Package Owner=<4>
# PROP Use_Debug_Libraries 1
# PROP Output Dir ".\Debug"
                                                                  Package=<5>
# PROP Intermediate Dir ".\Debug"
                                                                  {{{
# PROP Target Dir ""
                                                                  }}}
# ADD BASE CPP /nologo /W3 /Gm /GX /Zi /Od /D "WIN32" /D " DEBUG"
/D "_WINDOWS" /YX /c
                                                                  Package=<4>
# ADD CPP /nologo /W3 /Gm /GX /Zi /Od /D "WIN32" /D " DEBUG" /D
 WINDOWS" /YX /FD /c
                                                                    Begin Project Dependency
# ADD BASE MTL /nologo /D "_DEBUG" /win32
                                                                    Project Dep Name isapi_dll
# ADD MTL /nologo /D "_DEBUG" /mktyplib203 /win32
                                                                    End Project Dependency
# ADD BASE RSC /I 0x409 /d " DEBUG'
                                                                    Begin Project Dependency
# ADD RSC /l 0x409 /d " DEBUG"
                                                                    Project Dep Name tuxapp
                                                                    End Project Dependency
BSC32=bscmake.exe
# ADD BASE BSC32 /nologo
                                                                    Begin Project Dependency
                                                                    Project_Dep_Name db_dblib_dll
# ADD BSC32 /nologo
LINK32=link.exe
                                                                    End Project Dependency
# ADD BASE LINK32 kernel32.lib user32.lib gdi32.lib winspool.lib
                                                                    Begin Project Dependency
comdlg32.lib advapi32.lib shell32.lib ole32.lib oleaut32.lib uuid.lib odbc32.lib
                                                                    Project Dep Name db odbc dll
odbccp32.lib /nologo /subsystem:windows /debug /machine:I386
                                                                    End Project Dependency
# ADD LINK32 kernel32.lib user32.lib gdi32.lib winspool.lib comdlg32.lib
                                                                    Begin Project Dependency
advapi32.lib shell32.lib ole32.lib oleaut32.lib uuid.lib odbc32.lib odbccp32.lib
                                                                    Project Dep Name tm com dll
/nologo/subsystem:windows/debug/machine:I386
                                                                    End Project Dependency
                                                                    Begin Project Dependency
!ENDIF
                                                                    Project_Dep_Name tm_tuxedo_dll
                                                                    End Project Dependency
# Begin Target
                                                                    Begin Project Dependency
                                                                    Project_Dep_Name tpcc_com_all
# Name "webclnt - Win32 Release"
                                                                    End Project Dependency
# Name "webclnt - Win32 Debug"
                                                                    Begin Project Dependency
# End Target
                                                                    Project Dep Name tpcc com ps
# End Project
                                                                    End Project Dependency
                                                                  webclnt.dsw
                                                                  Microsoft Developer Studio Workspace File, Format Version 6.00
                                                                  Project: "isapi_dll"=\isapi_dll\isapi_dll.dsp - Package Owner=<4>
# WARNING: DO NOT EDIT OR DELETE THIS WORKSPACE FILE!
                                                                  Package=<5>
}}}
Project: "db dblib dll"=.\db dblib dll\db dblib dll.dsp - Package Owner=<4>
                                                                  Package=<4>
Package=<5>
                                                                    Begin Project Dependency
{{{
                                                                    Project Dep Name db dblib dll
}}}
                                                                    End Project Dependency
                                                                    Begin Project Dependency
Package=<4>
                                                                    Project_Dep_Name db_odbc_dll
End Project Dependency
}}}
                                                                    Begin Project Dependency
                                                                    Project_Dep_Name tm_tuxedo_dll
End Project Dependency
```

```
Begin Project Dependency
                                                       Project Dep Name tpcc com ps
 Project Dep Name tm com dll
                                                       End Project Dependency
 End Project Dependency
 Begin Project Dependency
 Project Dep Name tm encina dll
                                                     ###################
 End Project Dependency
}}}
                                                     Project: "tpcc_com_ps"=.\tpcc_com_ps\tpcc_com_ps.dsp - Package
Owner=<4>
                                                     Package=<5>
Project: "tm_com_dll"=.\tm_com_dll\tm_com_dll.dsp - Package Owner=<4>
                                                     -{{{
                                                     }}}
Package=<5>
                                                     Package=<4>
}}}
                                                     }}}
Package=<4>
                                                     ###################
 Begin Project Dependency
 Project_Dep_Name tpcc_com_ps
 End Project Dependency
                                                     Project: "tuxapp"=.\tuxapp\tuxapp.dsp - Package Owner=<4>
 Begin Project Dependency
 Project Dep Name tpcc com all
                                                     Package=<5>
 End Project Dependency
                                                     {{{
                                                     }}}
Package=<4>
Begin Project Dependency
Project: "tm encina dll"=.\tm encina dll\tm encina dll.dsp - Package
                                                       Project Dep Name db dblib dll
                                                       End Project Dependency
Owner=<4>
                                                       Begin Project Dependency
                                                       Project Dep Name db odbc dll
Package=<5>
                                                       End Project Dependency
{{{
}}}
Package=<4>
                                                     ###############################
}}}
                                                     Global:
Package=<5>
                                                     {{{
Project: "tm tuxedo dll"=.\tm tuxedo dll\tm tuxedo dll.dsp - Package
                                                     }}}
Owner=<4>
                                                     Package=<3>
Package=<5>
                                                     {{{
-{{{
                                                     }}}
}}}
                                                     Package=<4>
                                                     ###############################
}}}
                                                     Stored Procedures
neword.sql
Project: "tpcc com all"=.\tpcc com all\tpcc com all.dsp - Package
Owner=<4>
                                                     -- File: NEWORD.SQL
                                                           Microsoft TPC-C Benchmark Kit Ver. 4.22
Package=<5>
                                                           Copyright Microsoft, 2001
-{{{
                                                     -- Purpose: Creates new order transaction stored procedure
}}}
                                                            Interface Level: 4.10.000
Package=<4>
                                                     use tpcc
 Begin Project Dependency
```

```
begin transaction n
if exists ( select name from sysobjects where name = "tpcc neworder" )
          drop procedure tpcc_neworder
                                                                              -- get district tax and next availible order id and update
                                                                                plus initialize local variables
create proc tpcc neworder
                                                                                        update
                                                                                                   district
                                                                  smallint,
                                                     @w_id
                                                                                        set
                                                                                                   @d tax
                                                                                                              = d tax,
                                                     @d id
                                                                  tinyint,
                                                                                                   @o id
                                                                                                              = d next o id,
                                                                                                   d_{next_o_id} = d_{next_o_id} + 1,
                                                     @c id
                                                                 int,
                                                                                                   @o_{entry_d} = getdate(),
                                                     @o ol cnt
                                                                   tinvint.
                                                     @o_all_local tinyint,
                                                                                                   @li_no
                                                                                                              = 0,
                                                     @i_id1 int = 0,
                                                                                                   @commit_flag = 1
@s w id1 smallint = 0, @ol qty1 smallint = 0,
                                                                                        where
                                                                                                   d w id
                                                                                                              = @w id and
                                                     @i_id2 int = 0,
                                                                                                            = @d_id
                                                                                                   d_id
@s_w_id2 smallint = 0, @ol_qty2 smallint = 0,
                                                     (a)i id3 int = 0,
                                                                              -- process orderlines
@s w id3 smallint = 0, @ol gty3 smallint = 0,
                                                     @i_id4 \text{ int} = 0,
                                                                                        while (@li_no < @o_ol_cnt)
@s_w_id4 smallint = 0, @ol_qty4 smallint = 0,
                                                                                        begin
                                                     (a)i id5 int = 0,
@s w_id5 smallint = 0, @ol_qty5 smallint = 0,
                                                                                                   select @li_no = @li_no + 1
                                                     (a)i id6 int = 0,
@s_w_id6 smallint = 0, @ol_qty6 smallint = 0,
                                                                              -- set i_id, s_w_id, and qty for this lineitem
                                                     (a)i id7 int = 0,
@s_w_id7 smallint = 0, @ol_qty7 smallint = 0,
                                                                                                   select
                                                                                                             @li_id = case @li_no
                                                     @i_id8 int = 0,
                                                                                                                         when 1 then @i id1
@s w id8 smallint = 0, @ol qty8 smallint = 0,
                                                                                                                         when 2 then @i id2
                                                     @i_id9 int = 0,
                                                                                                                        when 3 then @i_id3
@s_w_id9 smallint = 0, @ol_qty9 smallint = 0,
                                                                                                                         when 4 then @i id4
                                                     @i_id10 int = 0,
                                                                                                                         when 5 then @i_id5
@s w id10 smallint = 0, @ol qty10 smallint = 0,
                                                                                                                         when 6 then @i id6
                                                     @i_id11 int = 0,
                                                                                                                         when 7 then @i_id7
@s_w_id11 smallint = 0, @ol_qty11 smallint = 0,
                                                                                                                         when 8 then @i id8
                                                     @i id12 int = 0,
                                                                                                                         when 9 then @i id9
@s_w_id12 smallint = 0, @ol_qty12 smallint = 0,
                                                                                                                         when 10 then @i_id10
                                                     @i id13 int = 0,
                                                                                                                         when 11 then @i id11
@s_w_id13 \text{ smallint} = 0, @ol_qty13 \text{ smallint} = 0,
                                                                                                                        when 12 then @i_id12
                                                     @i id14 int = 0,
                                                                                                                         when 13 then @i id13
@s_w_id14 smallint = 0, @ol_qty14 smallint = 0,
                                                                                                                         when 14 then @i_id14
                                                     @i id15 int = 0,
                                                                                                                         when 15 then @i_id15
@s w id15 smallint = 0, @ol qty15 smallint = 0
                                                                                                                         end,
                                                                                                             @li_s_w_id = case @li_no
                                                                                                                           when 1 then @s_w_id1
as
declare
                        numeric(4,4),
                                                                                                                           when 2 then @s w id2
          @w tax
    @d_tax
                 numeric(4,4),
                                                                                                                           when 3 then @s_w_id3
    @c last
                 char(16),
                                                                                                                           when 4 then @s w id4
    @c credit
                  char(2),
                                                                                                                           when 5 then @s w id5
    @c_discount
                                                                                                                           when 6 then @s_w_id6
                   numeric(4,4),
    @i price
                                                                                                                           when 7 then @s w id7
                 numeric(5,2),
    @i_name
                  char(24),
                                                                                                                           when 8 then @s_w_id8
                                                                                                                           when 9 then @s w id9
    @i data
                 char(50),
                                                                                                                           when 10 then @s_w_id10
    @o_entry_d
                   datetime,
    @remote_flag int,
                                                                                                                           when 11 then @s w id11
    @s_quantity
                   smallint,
                                                                                                                           when 12 then @s_w_id12
    @s_data
                 char(50),
                                                                                                                           when 13 then @s_w_id13
    @s dist
                 char(24),
                                                                                                                           when 14 then @s w id14
          @li_no
                       int
                                                                                                                           when 15 then @s_w_id15
          @o id
                               int,
                                                                                                                           end,
          @commit_flag tinyint,
    @li id
                                                                                                             @li_qty = case @li_no
                int,
    @li_s_w_id
                  smallint,
                                                                                                                         when 1 then @ol qty1
                 smallint,
                                                                                                                         when 2 then @ol_qty2
    @li_qty
          @ol number
                               int,
                                                                                                                         when 3 then @ol_qty3
          @c_id_local
                                                                                                                         when 4 then @ol_qty4
                               int
                                                                                                                         when 5 then @ol qty5
begin
                                                                                                                         when 6 then @ol_qty6
                                                                                                                         when 7 then @ol_qty7
```

```
when 8 then @ol_qty8
                                                                                                                                                   @li_no,
                                            when 9 then @ol_qty9
                                            when 10 then @ol qty10
                                                                                                                                                   @li_id,
                                            when 11 then @ol_qty11
                                            when 12 then @ol_qty12
                                            when 13 then @ol qty13
                                                                               @li_s_w_id,
                                            when 14 then @ol qty14
                                                                                                                                                   "dec 31,
                                                                               1899",
                                            when 15 then @ol_qty15
                                            end
                                                                                                                                                   @li_qty,
-- get item data (no one updates item)
                                                                                                                                                   @i_price
                                                                               * @li_qty,
                     select
                                @i price = i price,
                                                                                                                                                   @s dist)
                                @i_name = i_name,
                                (a)i data = i data
                     from
                                item (tablock repeatableread)
                                                                                 send line-item data to client
                     where
                                i_id = @li_id
                                                                                                               select
                                                                                                                          @i_name,
-- update stock values
                                                                                                                          @s_quantity,
                                                                                                                          b g = case when (
                                                                               (patindex("%ORIGINAL%",@i_data) > 0) and
                     update
                                stock
                                                     = s_ytd + @li_qty,
                     set
                                s ytd
                                @s_quantity
                                                     = s_quantity =
                                                                               (patindex("%ORIGINAL%",@s_data) > 0))
                                                                                                                             then "B" else "G" end,
s_quantity - @li_qty +
                                                                                                                          @i_price,
when (s_quantity - @li_qty < 10) then 91 else 0 end,
                                                                                                                          @i_price * @li_qty
                                s order ent
                                                     = s order cnt + 1,
                                s_remote_cnt
                                                     = s_remote_cnt + case
                                                                                                    end
when (@li_s_w_id = @w_id) then 0 else 1 end,
                                                                                                    else
                                @s\_data\\
                                                     = s_{data}
                                                                                                    begin
                                @s dist
                                                      = case @d id
                                                                                - no item (or stock) found - triggers rollback condition
                                                      when 1 then
s_dist_01
                                                                                                               select "",0,"",0,0
                                                      when 2 then
                                                                                                               select @commit_flag = 0
s_dist_02
                                                      when 3 then
s_dist_03
                                                                                                    end
                                                       when 4 then
                                                                                         end
s_dist_04
                                                      when 5 then
                                                                               -- get customer last name, discount, and credit rating
s dist 05
                                                      when 6 then
                                                                                                    @c_{last} = c_{last}
                                                                                         select
                                                                                                    @c discount = c discount,
s dist 06
                                                      when 7 then
                                                                                                    @c\_credit = c\_credit,
s dist 07
                                                                                                    (a)c id local = c id
                                                      when 8 then
                                                                                         from
                                                                                                    customer (repeatableread)
s dist 08
                                                                                         where
                                                                                                                          = (a)c id and
                                                                                                    c_id
                                                       when 9 then
                                                                                                    c w id
                                                                                                                          = @w id and
s_dist_09
                                                                                                    c_d_id
                                                                                                                          = @d_id
                                                       when 10 then
s_dist_10
                                                                               -- insert fresh row into orders table
                                                      end
                                                     = @li_id and
                     where
                                s_i_id
                                                                                         insert into orders values (
                                                                                                                          @o_id,
                                                     = @li_s_w_id
                                                                                                                                     @d id,
                                s\_w\_id
                                                                                                                                     @w_id,
-- if there actually is a stock (and item) with these ids, go to work
                                                                                                                                     @c_id_local,
                                                                                                                                     @o_entry_d,
                     if (@@rowcount > 0)
                                                                                                                                     0.
                                                                                                                                     @o ol cnt,
                     begin
                                                                                                                                     @o_all_local)
-- insert order_line data (using data from item and stock)
                                                                               -- insert corresponding row into new-order table
                                insert into order_line values(@o_id,
                                                                    @d_id,
                                                                                         insert into new_order values (
                                                                                                                          @o_id,
                                                                                                                                     @d_id,
                                                                    @w_id,
                                                                                                                                     @w_id)
                                                                                 select warehouse tax
```

```
@c first
                                                                                               char(16),
                                                                                  @c_middle
          select
                    @w_tax = w_tax
                                                                                                 char(2),
          from
                    warehouse (repeatableread)
                                                                                  @c_street_1
                                                                                                char(20),
          where
                    w_id
                               = @w_id
                                                                                  @c_street_2
                                                                                                char(20),
                                                                                  @c_city
                                                                                               char(20),
          if (@commit flag = 1)
                                                                                  @c state
                                                                                               char(2),
                                                                                  @c_zip
                                                                                               char(9),
                    commit transaction n
                                                                                  @c_phone
                                                                                                char(16),
          else
                                                                                  @c since
                                                                                               datetime,
                                                                                  @c credit
                                                                                               char(2),
-- all that work for nuthin!!!
                                                                                  @c_credit_lim numeric(12,2),
                                                                                                numeric(12,2),
                                                                                  @c_balance
                    rollback transaction n
                                                                                  @c discount
                                                                                                numeric(4,4),
                                                                                  @data
                                                                                              char(500),
                                                                                               char(500),
-- return order data to client
                                                                                  @c data
                                                                                  @datetime
                                                                                                datetime,
          select
                    @w tax,
                                                                                  @w ytd
                                                                                               numeric(12,2),
                     @d_tax,
                                                                                  @d_ytd
                                                                                               numeric(12,2),
                                                                                              smallint,
                    @o_id,
                                                                                  @cnt
                     @c last,
                                                                                  @val
                                                                                              smallint,
                                                                                  @screen_data char(200),
                     @c_discount,
                     @c credit,
                                                                                        @d id local
                                                                                                             tinyint,
                     @o_entry_d,
                                                                                        @w_id_local
                                                                                                             smallint,
                    @commit flag
                                                                                        @c_id_local
                                                                                                             int
                                                                             select @screen_data = ""
end
                                                                             begin tran p
go
                                                                             -- get payment date
payment.sql
                                                                                       select
                                                                                                  @datetime = getdate()
-- File: PAYMENT.SQL
                                                                                       if (@c id = 0)
        Microsoft TPC-C Benchmark Kit Ver. 4.22
                                                                                       begin
        Copyright Microsoft, 2001
-- Purpose: Creates payment transaction stored procedure
                                                                             -- get customer id and info using last name
          Interface Level: 4.10.000
                                                                                                 select
                                                                                                            @cnt
                                                                                                                      = count(*)
                                                                                                            customer (repeatableread)
                                                                                                 from
use tpcc
                                                                                                  where
                                                                                                            c last
                                                                                                                       = @c last and
                                                                                                                      = @c_w_id and
                                                                                                            c_w_id
                                                                                                                      = @c_d_id
                                                                                                            c_d_id
if exists (select name from sysobjects where name = "tpcc_payment")
          drop procedure tpcc_payment
go
                                                                                                            @val = (@cnt + 1) / 2
                                                                                                  select
                                                                                                            rowcount @val
                                                                                                 set
create proc tpcc_payment
                               @w_id
                                            smallint,
                                          @c_w_id
                                                        smallint,
                                                                                                 select
                                                                                                            @c_id
                                                                                                                      = c_id
                                          @h amount
                                                         numeric(6,2),
                                                                                                            customer (repeatableread)
                                                                                                 from
                                          @d_id
                                                      tinyint,
                                                                                                 where
                                                                                                            c last
                                                                                                                      = @c_last and
                                          @c_d_id
                                                       tinyint,
                                                                                                                      = (\overline{a})c w id and
                                                                                                            c w id
                                          @c id
                                                      int,
                                                                                                                      = @c_d_{id}
                                                                                                            c_d_id
                                          @c_last
                                                      char(16) = ""
                                                                                                            by c_last, c_first
                                                                                                 order
                                                                                                            rowcount 0
                                                                                                 set
                                                                                       end
declare @w_street_1 char(20),
     @w_street_2
                   char(20),
                                                                               get customer info and update balances
     @w_city
                  char(20),
     @w_state
                   char(2),
                                                                                       update
                                                                                                 customer
     @w_zip
                  char(9),
                                                                                                  @c_balance
                                                                                                                      = c_balance
                                                                                                                                            = c balance
     @w name
                    char(10),
                                                                             - @h_amount,
     @d_street_1
                   char(20),
                                                                                                  c_payment_cnt
                                                                                                                       = c_payment_cnt + 1,
     @d_street_2
                   char(20),
                                                                                                 c_ytd_payment
                                                                                                                       = c_ytd_payment + @h_amount,
     @d_city
                  char(20),
                                                                                                  (a)c first = c first,
     @d_state
                  char(2),
                                                                                                  @c\_middle
                                                                                                                      = c_middle,
     @d_zip
                  char(9),
                                                                                                  @c_last
                                                                                                                       = c_{last}
     @d_name
                   char(10),
```

```
= c \text{ street } 1,
                     @c street 1
                                                                                                  @w name
                                                                                                                       = w_name,
                     @c_street_2
                                          = c \text{ street } 2,
                                                                                                  @w_id_local
                                                                                                                       = w_id
                     @c_city
                                          = c_{city}
                                                                                        where
                                                                                                  w_id
                                                                                                                       = (a) w id
                     @c_state = c_state,
                                                                              -- create history record
                                          = c_zip,
                     @c_zip
                     (a)c phone = c phone,
                     @c credit = c credit,
                                                                                        insert into history values (
                                                                                                                        @c_id_local,
                     @c_credit_lim
                                                                                                                                  @c d id,
                                          = c credit lim,
                     @c_discount
                                          = c discount,
                                                                                                                                  @c_w_id,
                     (a)c since = c since,
                     @data
                                          = c_data,
                                                                                                                                  @d_id_local,
                     @c_id_local
                                                                                                                                  @w_id_local,
                                          = c id
          where
                     c id
                                          = (a)c id and
                                                                                                                                  @datetime,
                                          = @c_w_{id} and
                                                                                                                                  @h_amount,
                    c_w_id
                                                                                                                                  @w_name + "
                                          = @c_d_id
                    c_d_{id}
                                                                             @d_name)
-- if customer has bad credit get some more info
                                                                             commit tran p
          if(@c\_credit = "BC")
                                                                             -- return data to client
          begin
                                                                             select
                                                                                        @c_id,
          compute new info
                                                                                        @c last,
                                                                                        @datetime,
                     select @c data
                                          = convert(char(5),@c id) +
                                                                                        @w street 1,
                                           convert(char(4),@c_d_id) +
                                                                                        @w_street_2,
                                           convert(char(5),@c_w_id) +
                                                                                        @w_city,
                                           convert(char(4),@d id) +
                                                                                        @w_state,
                                           convert(char(5),@w_id) +
                                                                                        @w_zip,
                                           convert(char(19),@h amount) +
                                                                                        @d street 1,
                                           substring(@data, 1, 458)
                                                                                        @d_street_2,
                                                                                        @d_city,
                                                                                        @d_state,
-- update customer info
                                                                                        @d_zip,
                                                                                        @c first,
                     update
                                                                                        @c_middle,
                               customer
                                          = @c data
                                                                                        @c street 1,
                    set
                               c data
                                                                                        @c_street_2,
                                                                                        @c city,
                     where
                               c id
                                          = (a)c id and
                                                                                        @c_state,
                               c_w_id
                                          = @c_w_id and
                               c_d_id
                                          = @c_d_id
                                                                                        @c_zip,
                                                                                        @c phone,
                                                                                        @c_since,
                               @screen_data = substring (@c_data,1,200)
                     select
                                                                                        @c credit,
                                                                                        @c_credit_lim,
          end
                                                                                        @c_discount,
-- get district data and update year-to-date
                                                                                        @c_balance,
                                                                                        @screen data
          update
                     district
                                          = d_ytd + @h_amount,
                     d_ytd
          set
                                                                             go
                     ad street 1
                                          = d street 1,
                     @d_street_2
                                          = d_street_2,
                                          = d city,
                     @d city
                                                                             ordstat.sql
                     (a)d_state = d_state,
                     @d zip
                                          = d zip,
                                                                              -- File: ORDSTAT.SQL
                     @d name
                                          = d name,
                                                                                      Microsoft TPC-C Benchmark Kit Ver. 4.22
                     @d_id_local
                                          = d id
                                                                                      Copyright Microsoft, 2001
          where
                                          = @w id and
                     d w id
                                                                             -- Purpose: Creates order status transaction stored procedure
                     d_id
                                          = @d_id
                                                                                        Interface Level: 4.10.000
-- get warehouse data and update year-to-date
                                                                             use tpcc
          update
                     warehouse
                                                                             go
                                          = w_ytd + @h_amount,
                     w_ytd
          set
                     @w_street_1
                                          = w_street_1,
                                                                             if exists ( select name from sysobjects where name = "tpcc orderstatus" )
                                          = w_street_2,
                     @w_street_2
                                                                                        drop procedure tpcc_orderstatus
                     @w city
                                          = w city,
                                                                             go
                     @w_state = w_state,
                     @w_zip
                                          = w_zip,
                                                                            create proc tpcc_orderstatus
                                                                                                             @w_id
                                                                                                                       smallint,
```

```
@d_id
                                                         tinyint,
                                                                                                          goto custnotfound
                                                                                               end
                                              @c id
                                                         int.
                                                         char(16) = ""
                                              @c_last
                                                                                      get order info
                                                                                               select
                                                                                                           @o id
                                                                                                                                 = o id,
                                                                                                          @o_entry_d
                                                                                                                                  = o_entry_d,
declare @c_balance
                                  numeric(12,2),
                                                                                                           @o_carrier_id
                                                                                                                                  = o_carrier_id
                                  char(16),
     @c first
                                                                                               from
                                                                                                          orders (serializable)
     @c middle
                                  char(2),
                                                                                               where
                                                                                                          o c id
                                                                                                                                 = (a)c id and
     @o_id
                                              int,
                                                                                                          o_d_id
                                                                                                                                 = @d_id and
     @o_entry_d
                                                                                                                                 = (\widehat{a}) \mathbf{w}_{-} \mathbf{i} \mathbf{d}
                                  datetime,
                                                                                                          o_w_id
     @o carrier id
                                  smallint,
                                                                                               order
                                                                                                          by o_id asc
                                  smallint
     @cnt
                                                                                     - select order lines for the current order
begin tran o
                                                                                               select
                                                                                                          ol_supply_w_id,
if(@c_id = 0)
                                                                                                          ol_i_id,
                                                                                                          ol_quantity,
                                                                                                          ol amount,
                                                                                                          ol_delivery_d
-- get customer id and info using last name
                                                                                                          order line (repeatableread)
                                                                                               from
                       select
                                  @cnt
                                             = (count(*)+1)/2
                                                                                               where
                                                                                                          ol_o_id = @o_id and
                                  customer (repeatableread)
                                                                                                          of \overrightarrow{d} id = (\overrightarrow{a})\overrightarrow{d} id and
                      from
                       where
                                  c_last
                                              = @c_last and
                                                                                                          ol_w_id = @w_id
                                  c\_w\_id
                                             = \overline{@}w_id and
                                  c_d_id
                                             = (a)d id
                                                                                    custnotfound:
                      set
                                  rowcount @cnt
                                                                                    commit tran o
                      select
                                  @c id
                                                         = c id,
                                                                                    -- return data to client
                                  @c_balance
                                                         = c_balance,
                                    \overline{@}c_first = c_first,
                                                                                    select
                                                                                               @c_id,
                                    (a)c last = c last,
                                                                                               @c last,
                                    @c_middle
                                                         = c_{middle}
                                                                                               @c_first,
                       from
                                  customer (repeatableread)
                                                                                               @c middle,
                       where
                                  c_last
                                                         = @c_last and
                                                                                               @o_entry_d,
                                                                                               @o_carrier id,
                                                         = \overline{(a)} w id and
                                  c w id
                                                         = @d_id
                                                                                               @c_balance,
                                  c_d_id
                                  by c_w_id, c_d_id, c_last, c_first
                      order
                                                                                               @o_id
                                  rowcount 0
                       set
                                                                                    go
           end
           else
                                                                                    delivery.sql
           begin
                                                                                    -- File: DELIVERY.SQL
                                                                                             Microsoft TPC-C Benchmark Kit Ver. 4.22
-- get customer info if by id
                                                                                             Copyright Microsoft, 2001
                                                                                    -- Purpose: Creates delivery transaction stored procedure
                       select
                                  @c_balance
                                                         = c_balance,
                                  (a)c first = c first,
                                                                                               Interface Level: 4.10.000
                                  @c_middle
                                                         = c_{middle}
                                  @c_last
                                                         = c^{-} last
                                                                                    use tpcc
                       from
                                  customer (repeatableread)
                                                                                    go
                                                         = @c_id and
                                  c\_id
                       where
                                  c d id
                                                         = (a)d id and
                                                                                    if exists (select name from sysobjects where name = "tpcc_delivery")
                                  c_w_id
                                                         = @w_id
                                                                                               drop procedure tpcc_delivery
                       select
                                  @cnt
                                              = @@rowcount
                                                                                    create proc tpcc_delivery
                                                                                                                      @w_id
                                                                                                                                    smallint,
           end
                                                                                                                                  @o_carrier_id smallint
-- if no such customer
                                                                                    as
           if (@cnt = 0)
                                                                                    declare @d_id
                                                                                                          tinyint,
           begin
                                                                                         @o_id
                                                                                                          int,
                      raiserror("Customer not found",18,1)
                                                                                         @c_id
                                                                                                          int.
```

```
@total
                     numeric(12,2),
                                                                                                                            c_delivery_cnt
    @oid1
                      int,
                                                                                c delivery cnt + 1
                                                                                                                                                  = @w_id
    @oid2
                      int,
                                                                                                                 where
                                                                                                                            c_w_id
    @oid3
                      int,
    @oid4
                                                                                                                                                  = @d_id
                     int
                                                                                                                            c_d_id
    @oid5
                                                                                and
                      int,
    @oid6
                      int,
                                                                                                                            c id
                                                                                                                                                  = (a)c id
    @oid7
                      int,
    @oid8
                      int,
                                                                                                      end
    @oid9
                     int.
    @oid10
                                                                                           select @oid1 = case @d_id when 1 then @o_id else @oid1 end,
                                                                                         @oid2 = case @d_id when 2 then @o_id else @oid2 end,
select @d id = 0
                                                                                         @oid3 = case @d id when 3 then @o id else @oid3 end,
                                                                                         @oid4 = case @d_id when 4 then @o_id else @oid4 end,
begin tran d
                                                                                         @oid5 = case @d_id when 5 then @o_id else @oid5 end,
                                                                                         @oid6 = case @d_id when 6 then @o_id else @oid6 end,
                                                                                         @oid7 = case @d id when 7 then @o id else @oid7 end,
           while (@d id < 10)
          begin
                                                                                         @oid8 = case @d_id when 8 then @o_id else @oid8 end,
                                                                                         @oid9 = case @d_id when 9 then @o_id else @oid9 end,
               select @d id = @d id + 1,
                                                                                         @oid10 = case @d id when 10 then @o id else @oid10 end
                                \textcircled{a}total = 0,
                                \bar{(a)}o id = 0
                                                                                  end
                      select
                                                                                commit tran d
                                top 1
                                @o_id
                                           = no_o_id
                                                                                -- return delivery data to client
                      from
                                new_order (serializable updlock)
                                no w id = @w id and
                      where
                                no_d^-id = @d_i^-id
                                                                                select @oid1,
                     order
                                by no o id asc
                                                                                    @oid2,
                                                                                    @oid3,
               if (@@rowcount \Leftrightarrow 0)
                                                                                    @oid4,
                                                                                    @oid5,
                                                                                    @oid6,
-- claim the order for this district
                                                                                    @oid7,
                                                                                    @oid8,
                                                                                    @oid9,
                                delete
                                           new order
                                           no_w_id = @w_id and
                                where
                                                                                    @oid10
                                           no d id = \overline{(a)}d id and
                                           no_o_id = @o_id
                                                                                go
-- set carrier id on this order (and get customer id)
                                                                                stocklev.sql
                                update
                                           orders
                                set
                                           o_carrier_id
                                                                 =
                                                                                         STOCKLEV.SQL
@o carrier id,
                                                                                         Microsoft TPC-C Benchmark Kit Ver. 4.22
                                           @c_id
                                                                 = o_c_id
                                                                                         Copyright Microsoft, 2001
                                where
                                                                 = (\overline{a}) \mathbf{w}^{\mathsf{T}} \mathbf{id}
                                           o_w_id
                                                                                  Purpose: Creates stock level transaction stored procedure
and
                                           o_d_id
                                                                 = @d_id
                                                                                           Interface Level: 4.10.000
and
                                           o_id
                                                                 = @o_id
                                                                                use tpcc
-- set date in all lineitems for this order (and sum amounts)
                                                                                if exists (select name from sysobjects where name = "tpcc_stocklevel")
                                update
                                           order line
                                                                                           drop procedure tpcc_stocklevel
                                                                 = getdate(),
                                           ol_delivery_d
                                set
                                                                 = (a)total +
                                           @total
ol_amount
                                                                                create proc tpcc_stocklevel
                                                                                                                 @w_id
                                                                                                                                       smallint,
                                where
                                           ol_w_id
                                                                 = @w id
                                                                                                                            @d id
                                                                                                                                                  tinvint,
and
                                                                                                                            @threshhold
                                                                                                                                                  smallint
                                                                 = (a)d id
                                           ol_d_id
                                                                                as
and
                                                                 = @o_id
                                           ol_o_id
                                                                                declare
                                                                                           @o_id_low int,
                                                                                           @o_id_high int
-- accummulate lineitem amounts for this order into customer
                                                                                select
                                                                                           @o id low
                                                                                                                 = (d next o id - 20),
                                update
                                           customer
                                                                                           @o_id_high
                                                                                                                 = (d_next_o_id - 1)
                                           c balance = c balance + @total,
                                set
```

from

district

```
where
           d w id
                                 = @w id and
                                                                                drop proc tpcc delivery
           d id
                                 = (a)d id
                                                                                drop proc tpcc_neworder
                                                                                drop proc tpcc_orderstatus
select
           count(distinct(s_i_id))
                                                                                drop proc tpcc_payment
           stock, order_line
                                                                                drop proc tpcc_stocklevel
from
where
           ol w id
                                 = @w id and
                                                                                drop proc tpcc version
           ol_d_id
                                 = (a)d id and
                                                                                drop table order line null
           ol_o_id
                                  between @o_id_low and
                                  @o_id_high and
           s_w_id
                                 = ol w id and
                                                                                create proc tpcc_delivery
                                                                                                                               smallint,
                                                                                                                 @w_id
           s_i_id
                                 = ol_i_id and
           s\_quantity < @threshhold
                                                                                @o carrier id smallint
go
                                                                                declare @d_id tinyint,
                                                                                     @o_id int,
version.sql
                                                                                     @c_id int,
                                                                                     atotal numeric(12,2),
-- File: VERSION.SQL
                                                                                     @oid1 int,
        Microsoft TPC-C Benchmark Kit Ver. 4.22
                                                                                     @oid2 int,
         Copyright Microsoft, 2001
                                                                                     @oid3 int,
-- Purpose: Returns version level of TPC-C stored procs
                                                                                     @oid4 int,
-- Note: Always update the return value of this proc for
                                                                                     @oid5 int,
         any interface changes or "must have" bug fixes.
                                                                                     @oid6 int,
                                                                                     @oid7 int,
-- The value returned by this SP defines the "interface level",
                                                                                     @oid8 int,
-- which must match between the stored procs and the client code.
                                                                                     @oid9 int,
-- The interface level may be down rev from the current kit. This
                                                                                     @oid10 int
-- indicates that the interface hasn't changed since that version.
                                                                                declare @delaytime varchar(30)
use tpcc
                                                                                -- uniform random delay of 0 - 1 second; avg = 0.50
                                                                                select @delaytime = '00:00:0' + cast(cast((rand()*1.00) as decimal(4,3))) as
if exists ( select name from sysobjects where name = "tpcc version" )
                                                                                char(5)
           drop procedure tpcc_version
                                                                                waitfor delay @delaytime
go
                                                                                select 3001, 3001, 3001, 3001, 3001, 3001, 3001, 3001, 3001, 3001
create proc tpcc_version
declare
           @version char(8)
                                                                                GO
begin
                                                                                create proc tpcc neworder
           select @version = "4.10.000"
                                                                                                                                                    smallint,
                                                                                                                                       @w id
           select @version as "Version"
                                                                                                                                       @d id
                                                                                                                                                    tinyint,
end
                                                                                                                                       @c_id
                                                                                                                                                    int,
                                                                                                                                       @o ol cnt
                                                                                                                                                     tinyint,
go
                                                                                                                                       @o_all_local tinyint,
                                                                                                                                       @i_id1 int = 0,
                                                                                @s w id1 smallint = 0, @ol qty1 smallint = 0,
                                                                                                                                       @i_id2 int = 0,
null-txn.sql
                                                                                @s w id2 smallint = 0, @ol qty2 smallint = 0,
                                                                                                                                       @i_id3 int = 0,
-- TPC-C Null Txn Stored Procs
                                                                                @s w id3 smallint = 0, @ol qty3 smallint = 0,
-- Microsoft TPC-C Kit
                                                                                                                                       @i_id4 int = 0,
-- 8/17/99
                                                                                @s_w_id4 smallint = 0, @ol_qty4 smallint = 0,
                                                                                                                                       @i id5 int = 0,
-- This script will create stored procs which accept the same parameters and
                                                                                @s_w_id5 smallint = 0, @ol_qty5 smallint = 0,
return correctly formed
                                                                                                                                       @i id6 int = 0,
-- results sets to match the standard TPC-C stored procs. Of course, the advan-
                                                                                @s_w_id6 smallint = 0, @ol_qty6 smallint = 0,
tage is that these
                                                                                                                                       @i id7 int = 0,
-- stored procs place almost no load on SQL Server and do not require a
                                                                                @s_w_id7 \text{ smallint} = 0, @ol_qty7 \text{ smallint} = 0,
database.
                                                                                                                                       @i id8 int = 0,
                                                                                @s_w_id8 smallint = 0, @ol_qty8 smallint = 0,
-- The purpose of these stored procs is to size and test the web client without the
                                                                                                                                       @i_id9 int = 0,
need of a fully
                                                                                @s w id9 smallint = 0, @ol qty9 smallint = 0,
-- scaled database.
                                                                                                                                       @i_id10 int = 0,
                                                                                @s_w_id10 smallint = 0, @ol_qty10 smallint = 0,
```

```
@i_id11 \text{ int} = 0,
                                                                                                    end
@s_w_id11 \text{ smallint} = 0, @ol_qty11 \text{ smallint} = 0,
                                                                                          else
                                                     (a)i id12 int = 0,
                                                                                                    begin
@s_w_id12 \text{ smallint} = 0, @ol_qty12 \text{ smallint} = 0,
                                                                                                    select 'Item Name blah',17,'G', @i_price, @i_price *
                                                     @i_id13 int = 0,
                                                                               @li_qty
@s w id13 smallint = 0, @ol qty13 smallint = 0,
                                                                                                    end
                                                     @i id14 int = 0,
@s_w_id14 smallint = 0, @ol_qty14 smallint = 0,
                                                                                          end
                                                     @i id15 int = 0,
@s_w_id15 smallint = 0, @ol_qty15 smallint = 0
                                                                               -- return order data to client
                                                                                                    @w \tan = 0.1234,
                                                                                          select
                                                                                                    (a)d tax = 0.0987,
                                                                                                    @o_{id} = 3001,
declare
          @w_tax
                        numeric(4,4),
    @d tax
                                                                                                    @c last = 'BAROUGHTABLE',
                 numeric(4,4),
                                                                                                    @c_discount = 0.2198,
    @c_last
                 char(16),
    @c credit
                  char(2),
                                                                                                    @c credit = 'GC',
    @c_discount
                    numeric(4,4),
                                                                                                    @o_entry_d = getdate()
    @i_price
                 numeric(5,2),
    @i name
                   char(24),
                                                                                          select
                                                                                                    @w tax,
    @o_entry_d
                    datetime,
                                                                                                    @d_tax,
           @li_no
                       int,
                                                                                                    @o id,
           @o_id
                                int.
                                                                                                    @c_last,
          @commit_flag tinyint,
                                                                                                    @c discount,
    @li id
                 int,
                                                                                                    @c_credit,
                                                                                                    @o_entry_d,
                 smallint
    @li_qty
                                                                                                    @commit flag
declare @delaytime varchar(30)
                                                                               end
begin
  -- uniform random delay of 0 - 0.6 second; avg = 0.3
                                                                               GO
  select @delaytime = '00:00:0' + cast(cast((rand()*0.60) as decimal(4,3))) as
char(5))
                                                                               create proc tpcc_orderstatus @w_id
                                                                                                                                     smallint,
  waitfor delay @delaytime
                                                                               @d_id
                                                                                                    tinyint,
  -- process orderlines
           select @commit flag = 1, @li no = 0
                                                                               @c id
                                                                                                    int,
  while (@li_no < @o_ol_cnt)
                                                                                         char(16) = "
                                                                               @c_last
           begin
                                                                               as
           select @li id = case @li no
                                                                               declare @c_balance
                                                                                                               numeric(12,2),
                                when 1 then @i_id1
                                                                                                               char(16),
                                  when 2 then @i id2
                                                                                    @c first
                                  when 3 then @i_id3
                                                                                    @c_middle
                                                                                                               char(2),
                                  when 4 then @i id4
                                                                                    @o id
                                                                                                                          int.
                                  when 5 then @i id5
                                                                                    @o entry d
                                                                                                               datetime,
                                  when 6 then @i_id6
                                                                                                               smallint,
                                                                                    @o_carrier_id
                                  when 7 then @i id7
                                                                                         @ol cnt
                                                                                                                          smallint
                                  when 8 then @i_id8
                                  when 9 then @i id9
                                                                               declare @delaytime varchar(30)
                                  when 10 then @i_id10
                                  when 11 then @i id11
                                                                                 -- uniform random delay of 0 - 0.2 second; avg = 0.1
                                  when 12 then @i_id12
                                  when 13 then @i_id13
                                                                                 select @delaytime = '00:00:0' + cast(cast((rand()*0.20) as decimal(4,3)) as
                                  when 14 then @i id14
                                  when 15 then @i_id15
                                                                                 waitfor delay @delaytime
                                  end
                                                                                          select
                                                                                              @c id = 113,
    select @li no = @li no + 1
                                                                                              @c_balance = -10.00,
           select @i_price = 23.45, @li_qty = @li_no
                                                                                              @c_first = '8YCodgytqCj8',
                                                                                              @c_{middle} = 'OE'
          if (@li_id = 999999)
                                                                                              @c_last = 'OUGHTOUGHTABLE',
                     begin
                                                                                              @o id = 3456,
                     select ",0,",0,0
                                                                                              @o_entry_d = getdate(),
                     select @commit flag = 0
                                                                                              @o_carrier_id = 1
```

```
@d ytd
                                                                                              numeric(12,2),
          select @ol cnt = (rand() * 11) + 5
                                                                                 @cnt
                                                                                            smallint,
          SET ROWCOUNT @ol cnt
                                                                                 @val
                                                                                            smallint.
                                                                                                 char(200),
                                                                                 @screen_data
          select
                                                                                                 @d id local
                                                                                                                     tinyint,
                    ol supply w id,
                                                                                                 @w id local
                                                                                                                      smallint,
                    ol_i_id,
                                                                                                 @c id local
                                                                                                                     int
              ol quantity,
              ol amount,
                                                                            declare @delaytime varchar(30)
              ol delivery d
                                                                              -- uniform random delay of 0 - 0.3 second; avg = 0.15
          from order_line_null
                                                                              select @delaytime = '00:00:0' + cast(cast((rand()*0.30) as decimal(4,3)) as
          select @c id,
           @c_last,
                                                                              waitfor delay @delaytime
            @c first,
            @c_middle,
                                                                              select @screen data = "
            @o_entry_d,
            @o_carrier_id,
            @c_balance,
                                                                            -- get customer info and update balances
            @o id
                                                                                      select
                                                                                                @d street 1 = 'rqSHHakqyV',
GO
                                                                                                \textcircled{a}d street 2 = 'zZ98nW3BR2s',
                                                                                                @d city = 'ArNr4GNFV9',
create proc tpcc_payment @w_id
                                     smallint,
                                                                                                @d_state = 'aV',
                                                              @c\_w\_id
                                                                                                @d_{zip} = '453511111'
  smallint,
                                                              @h_amount
                                                                            -- get warehouse data and update year-to-date
   numeric(6,2),
                                                              @d_id
                                                                                      select
tinyint,
                                                                                                @w street 1 = \text{'rqSHHakqyV'},
                                                                                                @w_street_2 = 'zZ98nW3BR2s',
                                                              @c_d_id
 tinyint,
                                                                                                @w city = 'ArNr4GNFV9',
                                                                                                \overline{(a)}w state = 'aV',
                                                              @c id
                                                                                                @w_zip = '453511111'
int,
                                                              @c last
 char(16) = "
                                                                                      select
                                                                                                @c id
                                                                                                               = 123,
                                                                                                @c_balance = -10000.00,
                                                                                                @c_first
                                                                                                           = 'KmR03Xureb',
declare @w_street_1 char(20),
     @w street 2
                   char(20),
                                                                                                @c middle = 'OE',
                                                                                                          = 'BAROUGHTBAR',
     @w_city
                  char(20),
                                                                                                @c_last
     @w state
                  char(2),
                                                                                                @c street 1 = 'QpGdOHjv8mR9vNI8V',
                  char(9),
                                                                                                @c_street_2 = 'dzKoCObBqbC3yu',
     @w_zip
                   char(10),
                                                                                                            = 'zAKZXdC037FQxq',
     @w name
                                                                                                @c city
                                                                                                            = 'QA',
     @d_street_1
                                                                                                @c_state
                   char(20),
     @d street 2
                   char(20),
                                                                                                @c_zip
                                                                                                            = '7003111111',
     @d city
                  char(20),
                                                                                                @c phone
                                                                                                              = '2967264064528555',
     @d_state
                                                                                                @c_credit
                                                                                                             = 'GC',
                  char(2),
     @d zip
                  char(9),
                                                                                                @c credit lim = 50000.00,
                                                                                                @c_{discount} = 0.3069,
     @d_name
                   char(10),
     @c first
                                                                                                             = getdate(),
                 char(16),
                                                                                                @c since
     @c_middle
                                                                                                @datetime
                   char(2),
                                                                                                             = getdate()
     @c_street_1
                   char(20),
     @c_street_2
                   char(20),
                 char(20),
     @c_city
     @c state
                  char(2),
                                                                            -- return data to client
     @c_zip
                 char(9),
     @c phone
                   char(16),
                                                                            select @c id,
     @c_since
                  datetime,
                                                                                                @c_last,
     @c credit
                  char(2),
                                                                                                @datetime,
     @c credit lim numeric(12,2),
                                                                                                @w_street_1,
     @c_balance
                   numeric(12,2),
                                                                                                @w_street_2,
     @c_discount
                   numeric(4,4),
                                                                                                @w_city,
     @data
                 char(500),
                                                                                                @w_state,
     @c data
                  char(500),
                                                                                                @w zip,
                  datetime,
     @datetime
                                                                                                @d_street_1,
     @w_ytd
                  numeric(12,2),
                                                                                                @d_street_2,
```

```
@d city,
                                                                                  insert into order line null values (110, 1, getdate(), 5, 123.45)
                      @d_state,
                                                                                  insert into order_line_null values (111, 1, getdate(), 1, 123.45)
                      @d_zip,
                                                                                  insert into order line null values (112, 1, getdate(), 2, 123.45)
                      @c_first,
                                                                                  insert into order_line_null values (113, 1, getdate(), 3, 123.45)
                      @c_middle,
                                                                                  insert into order_line_null values (114, 1, getdate(), 4, 123.45)
                      @c street 1,
                                                                                  insert into order line null values (115, 1, getdate(), 5, 123.45)
                      @c_street_2,
                                                                                  GO
                      @c_city,
                      @c_state,
                      @c zip,
                      @c_phone,
                      @c_since,
                      @c credit,
                      @c_credit_lim,
                      @c discount,
                      @c_balance,
                      @screen_data
GO
                                                       smallint,
create proc tpcc stocklevel
                                 @w id
                                                                   @d_id
tinyint,
                                                                   @threshhold
           smallint
declare @delaytime varchar(30)
  -- uniform random delay of 0 - 3.6 second; avg = 1.8
  select @delaytime = '00:00:0' + cast(cast((rand()*3.60) as decimal(4,3))) as
char(5))
  waitfor delay @delaytime
  select 49
GO
create proc tpcc_version
declare @version
                      char(8)
begin
           select @version = '4.10.000'
           select @version as 'Version'
end
GO
CREATE TABLE order_line_null (
           [ol i id] [int] NOT NULL,
           [ol\_supply\_w\_id] \ [smallint] \ NOT \ NULL \ ,
           [ol\_delivery\_d] \ [datetime] \ NOT \ NULL \ ,
           [ol_quantity] [smallint] NOT NULL,
           [ol_amount] [numeric](6, 2) NOT NULL
) ON [PRIMARY]
GO
insert into order_line_null values (101, 1, getdate(), 1, 123.45)
insert into order line null values (102, 1, getdate(), 2, 123.45)
insert into order_line_null values (103, 1, getdate(), 3, 123.45)
insert into order_line_null values (104, 1, getdate(), 4, 123.45)
insert into order_line_null values (105, 1, getdate(), 5, 123.45)
insert into order_line_null values (106, 1, getdate(), 1, 123.45)
insert into order line null values (107, 1, getdate(), 2, 123.45)
insert into order_line_null values (108, 1, getdate(), 3, 123.45)
insert into order_line_null values (109, 1, getdate(), 4, 123.45)
```

Appendix B: Database Design

Database Build

backup.sql

```
-- File: BACKUP.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.22
-- Copyright Microsoft, 2001
-- Purpose: Creates backup of tpcc database

declare @startdate datetime
declare @enddate datetime
select @startdate = getdate()
select "Start date:", convert(varchar(30),@startdate,9)

dump database tpcc to tpccback1, tpccback2, tpccback3, tpccback4 with init, stats = 1

select @enddate = getdate()
select "End date: ", convert(varchar(30),@enddate,9)
select "Elapsed time (in seconds): ", datediff(second, @startdate, @enddate)

Go
```

backupdev.sql

-- File: BACKUPDEVB.SQL

```
-- Microsoft TPC-C Benchmark Kit Ver. 4.22
-- Copyright Microsoft, 2001
-- Purpose: Creates tpcc database Backup Devices

use master
go
-- create backup devices

exec sp_addumpdevice 'disk','tpccback1','Y:\tpccback1.dmp'
go
exec sp_addumpdevice 'disk','tpccback2','Y:\tpccback2.dmp'
go
exec sp_addumpdevice 'disk','tpccback3','Y:\tpccback3.dmp'
go
exec sp_addumpdevice 'disk','tpccback4','Y:\tpccback3.dmp'
go
exec sp_addumpdevice 'disk','tpccback4','Y:\tpccback4.dmp'
```

createdb.sql

go

```
-- File: CREATEDB.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.22
-- Copyright Microsoft, 2001
-- Purpose: Creates tpcc database and backup files

use master
go
-- Create temporary table for timing
if exists ( select name from sysobjects where name = 'tpcc_timer' )
drop table tpcc timer
```

```
create table tpcc_timer
          start_date
                                                char(30),
         end_date
                                                char(30)
         into tpcc timer values (0,0)
insert
go
         Store starting time
update
         tpcc timer
         start_date = (select convert(char(30), getdate(),9))
set
go
-- create main database files
CREATE DATABASE tpcc
ON PRIMARY
                             = MSSQL_tpcc_root,
         NAME
         FILENAME
                             = "Y:\MSSQL tpcc root.mdf",
         SIZE
                             = 8MB
         FILEGROWTH
                             =0).
FILEGROUP
                   MSSQL_misc_fg
                             = MSSQL misc1,
         NAME
         FILENAME
                             = "G:"
                             = 14000MB,
         SIZE
         FILEGROWTH
                             = 0),
                             = MSSQL_misc2,
         NAME
         FILENAME
                             = "I:",
                             = 14000MB,
         SIZE
         FILEGROWTH
                             = 0
                             = MSSQL misc3,
         NAME
         FILENAME
                            = "K:".
                             = 14000MB,
         FILEGROWTH
                             = 0),
FILEGROUP
                   MSSQL_cs_fg
                             = MSSQL_cs1,
         NAME
         FILENAME
                             = "F:".
         SIZE
                             = 29200MB,
         FILEGROWTH
                             = 0),
                             = MSSQL cs2,
         NAME
         FILENAME
                             = "H·"
                             = 29200MB,
         SIZE
         FILEGROWTH
                             = 0),
         NAME
                             = MSSQL_{cs3},
         FILENAME
                             = "J:"
                             = 29200MB,
         SIZE
         FILEGROWTH
LOG ON
                             =MSSQL_tpcc_log,
         NAME
         FILENAME
                             ="E:".
         SIZE
                             =45000MB,
         FILEGROWTH
COLLATE Latin1_General_BIN
-- Store ending time
update
         end date = (select convert(char(30), getdate(),9))
set
select "Elapsed time (in seconds): ", datediff(second,(select start_date from
tpcc_timer),(select end_date from tpcc_timer))
         remove temporary table
```

```
if exists ( select name from sysobjects where name = 'tpcc timer' )
                                                                             EXEC sp_indexoption
                                                                                                             'district', 'DisAllowPageLocks', TRUE
          drop table tpcc timer
                                                                             EXEC sp_indexoption
                                                                                                             'warehouse',
                                                                                                                                  'DisAllowPageLocks',
                                                                             TRUE
go
                                                                             EXEC sp_indexoption
                                                                                                             'stock',
                                                                                                                       'DisAllowPageLocks', TRUE
                                                                             EXEC sp_indexoption
                                                                                                                                  'DisAllowRowLocks',
                                                                                                             'order_line',
dbopt1.sql
                                                                             EXEC sp_indexoption
                                                                                                             'orders',
                                                                                                                      'DisAllowRowLocks', TRUE
                                                                             EXEC sp_indexoption
                                                                                                             'new_order',
                                                                                                                                  'DisAllowRowLocks',
-- File: DBOPT1.SQL
                                                                             TRUE
        Microsoft TPC-C Benchmark Kit Ver. 4.22
                                                                             EXEC sp_indexoption
                                                                                                             'item',
                                                                                                                                  'DisAllowRowLocks',
        Copyright Microsoft, 2001
                                                                             TRUE
-- Purpose: Sets database options for data load
                                                                                                                                  'DisAllowPageLocks',
                                                                             EXEC sp_indexoption
                                                                                                             'item',
                                                                             TRUE
use master
go
                                                                             Print''
                                                                             Print '*************
exec sp dboption tpcc, 'select into/bulkcopy', true
                                                                             Print 'Pre-specified Locking Hierarchy:'
exec sp_dboption tpcc,'trunc. log on chkpt.',true
                                                                                         Lockflag = 0 ==> No pre-specified hierarchy'
                                                                             Print '
exec sp_dboption tpcc,'torn page detection',false
                                                                             Print ' Lockflag = 1 ==> Lock at Page-level then Table-level'
                                                                             Print ' Lockflag = 2 ==> Lock at Row-level then Table-level'
                                                                             .. Lockflag = 2 ==> Lock at Row-level the Print ' Lockflag = 3 ==> Lock at Table-level' Print ' '
use tpcc
go
                                                                             SELECT name, lockflags
checkpoint
                                                                             FROM
                                                                                       sysindexes
go
                                                                             WHERE
                                                                                       object id('warehouse')
                                                                                                                       = id OR
                                                                                       object_id('district') = id OR
dbopt2.sql
                                                                                        object id('customer') = id OR
                                                                                        object_id('stock')
                                                                                                            = id OR
                                                                                        object_id('orders')
                                                                                                            = id OR
-- File: DBOPT2.SQL
                                                                                        object_id('order_line') = id OR
        Microsoft TPC-C Benchmark Kit Ver. 4.22
                                                                                        object_id('history') = id OR
        Copyright Microsoft, 2001
                                                                                        object id('new order')
                                                                                                                       = id OR
-- Purpose: Resets database options after data load
                                                                                        object_id('item')
                                                                             ORDER BY lockflags asc
sp_dboption tpcc,'select into/bulkcopy',FALSE
                                                                             sp_configure 'allow updates',0
sp_dboption tpcc,'trunc. log on chkpt.',FALSE
GO
                                                                             RECONFIGURE WITH OVERRIDE
USE tpcc
GO
                                                                             EXEC sp_dboption tpcc,
                                                                                                             'auto update statistics', FALSE
CHECKPOINT
                                                                             EXEC sp_dboption tpcc,
                                                                                                             'auto create statistics', FALSE
sp_configure 'allow updates',1
                                                                             EXEC sp_tableoption 'district', 'pintable',true
                                                                             EXEC sp tableoption 'warehouse',
                                                                                                                       'pintable',true
                                                                             EXEC sp_tableoption 'new_order',
                                                                                                                        'pintable',true
RECONFIGURE WITH OVERRIDE
                                                                             EXEC sp tableoption 'item',
                                                                                                                        'pintable',true
DECLARE
                     @msg
                                          varchar(50)
                                                                             idxcuscl.sql
      OPTIONS FOR SQL SERVER 8.0
-- Set option values for user-defined indexes --
                                                                              -- File: IDXCUSCL.SQL
                                                                                      Microsoft TPC-C Benchmark Kit Ver. 4.22
                                                                                      Copyright Microsoft, 2001
          @msg
SET
                                                                             -- Purpose: Creates clustered index on customer table
PRINT
          @msg
SET
                     = 'Setting SQL Server indexoptions'
          @msg
PRINT
          @msg
                                                                             use tpcc
SET
          @msg
                                                                             go
PRINT
          @msg
                                                                             declare @startdate datetime
EXEC sp_indexoption
                               'customer', 'DisAllowPageLocks', TRUE
                                                                             declare @enddate datetime
```

```
select @startdate = getdate()
                                                                               select @enddate = getdate()
select "Start date:", convert(varchar(30),@startdate,9)
                                                                               select "End date: ", convert(varchar(30),@enddate,9)
                                                                               select "Elapsed time (in seconds): ", datediff(second, @startdate, @enddate)
if exists ( select name from sysindexes where name = 'customer_c1')
          drop index customer.customer_c1
create unique clustered index customer c1 on customer(c w id, c d id, c id)
          on MSSQL_cs_fg
                                                                               idxitmcl.sql
select @enddate = getdate()
select "End date: ", convert(varchar(30),@enddate,9)
                                                                                       IDXITMCL.SQL
                                                                                -- File:
select "Elapsed time (in seconds): ", datediff(second, @startdate, @enddate)
                                                                                        Microsoft TPC-C Benchmark Kit Ver. 4.22
                                                                                        Copyright Microsoft, 2001
                                                                               -- Purpose: Creates clustered index on item table
idxcusnc.sql
                                                                               use tpcc
                                                                               go
-- File: IDXCUSNC.SQL
        Microsoft TPC-C Benchmark Kit Ver. 4.22
                                                                               declare @startdate datetime
        Copyright Microsoft, 2001
                                                                               declare @enddate datetime
                                                                               select @startdate = getdate()
select "Start date:", convert(varchar(30),@startdate,9)
-- Purpose: Creates non-clustered index on customer table
use tpcc
                                                                               if exists (select name from sysindexes where name = 'item c1')
                                                                                          drop index item.item_c1
declare @startdate datetime
                                                                               create unique clustered index item c1 on item(i id)
declare @enddate datetime
                                                                                          on MSSQL_misc_fg
select @startdate = getdate()
select "Start date:", convert(varchar(30),@startdate,9)
                                                                               select @enddate = getdate()
                                                                               select "End date: ", convert(varchar(30),@enddate,9)
                                                                               select "Elapsed time (in seconds): ", datediff(second, @startdate, @enddate)
if exists (select name from sysindexes where name = 'customer_nc1')
          drop index customer.customer nc1
                                                                               go
create unique nonclustered index customer_nc1 on customer(c_w_id, c_d_id,
c_last, c_first, c_id)
          on MSSQL_cs_fg
                                                                               idxnodcl.sql
select @enddate = getdate()
                                                                                -- File: IDXNODCL.SQL
select "End date: ", convert(varchar(30),@enddate,9)
                                                                                        Microsoft TPC-C Benchmark Kit Ver. 4.22
select "Elapsed time (in seconds): ", datediff(second, @startdate, @enddate)
                                                                                        Copyright Microsoft, 2001
                                                                               -- Purpose: Creates clustered index on new_order table
go
                                                                               use tpcc
idxdiscl.sql
-- File: IDXDISCL.SQL
                                                                               declare @startdate datetime
        Microsoft TPC-C Benchmark Kit Ver. 4.22
                                                                               declare @enddate datetime
        Copyright Microsoft, 2001
                                                                               select @startdate = getdate()
                                                                               select "Start date:", convert(varchar(30),@startdate,9)
-- Purpose: Creates clustered index on district table
                                                                               if exists ( select name from sysindexes where name = 'new_order_c1')
                                                                                          drop index new_order.new_order_c1
use tpcc
                                                                               create unique clustered index new_order_c1 on new_order(no_w_id, no_d_id,
declare @startdate datetime
                                                                               no_o_id)
declare @enddate datetime
                                                                                          on MSSQL_misc_fg
select @startdate = getdate()
select "Start date:", convert(varchar(30),@startdate,9)
                                                                               select @enddate = getdate()
                                                                               select "End date: ", convert(varchar(30),@enddate,9)
if exists ( select name from sysindexes where name = 'district c1')
                                                                               select "Elapsed time (in seconds): ", datediff(second, @startdate, @enddate)
           drop index district_c1
                                                                               go
```

create unique clustered index district_c1 on district(d_w_id, d_id) with fillfactor=100 on MSSQL_misc_fg

idxodlcl.sql

use tpcc

-- File: IDXODLCL.SOL

Microsoft TPC-C Benchmark Kit Ver. 4.22

-- Purpose: Creates clustered index on order line table

Copyright Microsoft, 2001

```
declare @startdate datetime
declare @enddate datetime
select @startdate = getdate()
select "Start date:", convert(varchar(30),@startdate,9)
if exists (select name from sysindexes where name = 'order_line_c1')
          drop index order_line.order_line_c1
create unique clustered index order line c1 on order line(ol w id, ol d id,
ol o id, ol number)
          on MSSQL_misc_fg
select @enddate = getdate()
select "End date: ", convert(varchar(30),@enddate,9)
select "Elapsed time (in seconds): ", datediff(second, @startdate, @enddate)
idxordcl.sql
-- File: IDXORDCL.SQL
        Microsoft TPC-C Benchmark Kit Ver. 4.22
        Copyright Microsoft, 2001
-- Purpose: Creates clustered index on orders table
use tpcc
go
declare @startdate datetime
declare @enddate datetime
select @startdate = getdate()
select "Start date:", convert(varchar(30),@startdate,9)
if exists ( select name from sysindexes where name = 'orders_c1' )
          drop index orders.orders c1
create unique clustered index orders_c1 on orders(o_w_id, o_d_id, o_id)
          on MSSQL_misc_fg
select @enddate = getdate()
select "End date: ", convert(varchar(30),@enddate,9)
select "Elapsed time (in seconds): ", datediff(second, @startdate, @enddate)
go
idxordnc.sql
-- File: IDXORDNC.SQL
        Microsoft TPC-C Benchmark Kit Ver. 4.22
        Copyright Microsoft, 2001
-- Purpose: Creates non-clustered index on orders table
```

```
use tpcc
go
declare @startdate datetime
declare @enddate datetime
select @startdate = getdate()
select "Start date:", convert(varchar(30),@startdate,9)
if exists (select name from sysindexes where name = 'orders nc1')
           drop index orders_orders_nc1
create index orders nc1 on orders(o w id, o d id, o c id, o id)
           on MSSQL_misc_fg
select @enddate = getdate()
select "End date: ", convert(varchar(30),@enddate,9)
select "Elapsed time (in seconds): ", datediff(second, @startdate, @enddate)
go
idxstkcl.sql
-- File: IDXSTKCL.SQL
         Microsoft TPC-C Benchmark Kit Ver. 4.22
         Copyright Microsoft, 2001
-- Purpose: Creates clustered index on stock table
use tpcc
declare @startdate datetime
declare @enddate datetime
select @startdate = getdate()
select "Start date:", convert(varchar(30),@startdate,9)
if exists (select name from sysindexes where name = 'stock c1')
           drop index stock.stock_c1
```

```
idxwarcl.sql
-- File: IDXWARCL.SQL
        Microsoft TPC-C Benchmark Kit Ver. 4.22
        Copyright Microsoft, 2001
-- Purpose: Creates clustered index on warehouse table
use tpcc
go
declare @startdate datetime
declare @enddate datetime
select @startdate = getdate()
select "Start date:", convert(varchar(30),@startdate,9)
```

create unique clustered index stock_c1 on stock(s_i_id, s_w_id)

select "Elapsed time (in seconds): ", datediff(second, @startdate, @enddate)

select "End date: ", convert(varchar(30),@enddate,9)

on MSSQL_cs_fg

select @enddate = getdate()

```
if exists (select name from sysindexes where name = 'warehouse c1')
          drop index warehouse.warehouse c1
create unique clustered index warehouse_c1 on warehouse(w_id)
                                                                              exec sp_configure "show advanced option", 1
          with fillfactor=100 on MSSQL_misc_fg
select @enddate = getdate()
                                                                              reconfigure with override
select "End date: ", convert(varchar(30),@enddate,9)
select "Elapsed time (in seconds): ", datediff(second, @startdate, @enddate)
                                                                              /* change this value to approximately the number of connected users */
go
                                                                              exec sp_configure "max worker threads",240
                                                                              /* increase priority of user threads */
                                                                              exec sp_configure "priority boost",1
removedb.sql
                                                                              /* disable automatic checkpointing */
-- File: REMOVEDB.SQL
                                                                              exec sp_configure "recovery interval",56
        Microsoft TPC-C Benchmark Kit Ver. 4.22
        Copyright Microsoft, 2001
                                                                              /* change to a mask appropriate for the number of processors on the server */
-- Purpose: Removes tpcc database and backup files
                                                                              exec sp configure "affinity mask",0x7
                                                                              /* enable fibers */
use master
                                                                              exec sp_configure "lightweight pooling",1
                                                                              /* enable update */
-- remove any existing database and backup files
                                                                              exec sp_configure "allow updates",1
exec sp dbremove tpcc, dropdev
                                                                              /* set max degree of parallelism */
                                                                              exec sp configure "max degree of parallelism",1
exec sp dropdevice 'tpccback1'
exec sp_dropdevice 'tpccback2'
exec sp dropdevice 'tpccback3'
                                                                              reconfigure with override
exec sp_dropdevice 'tpccback4'
                                                                              sqlshutdown.sql
restore.sql
                                                                              use tpcc
-- File: RESTORE.SQL
        Microsoft TPC-C Benchmark Kit Ver. 4.22
                                                                              checkpoint
        Copyright Microsoft, 2001
-- Purpose: Loads database backup from backup files
                                                                              shutdown
                                                                              go
declare @startdate datetime
declare @enddate datetime
                                                                              tables.sql
select @startdate = getdate()
--select "Start date:", convert(varchar(30),@startdate,9)
                                                                              -- File: TABLES.SQL
load database tpcc from tpccback1, tpccback2, tpccback3, tpccback4 with stats
                                                                                      Microsoft TPC-C Benchmark Kit Ver. 4.22
                                                                                      Copyright Microsoft, 2001
                                                                              -- Purpose: Creates TPC-C tables
select @enddate = getdate()
--select "End date: ", convert(varchar(30),@enddate,9)
--select "Elapsed time (in seconds): ", datediff(second, @startdate, @enddate)
                                                                              use tpcc
                                                                              go
go
                                                                              -- Remove all existing TPC-C tables
RunSQLCfg.sql
                                                                              if exists ( select name from sysobjects where name = 'warehouse' )
/* TPC-C Benchmark Kit
                                                                                         drop table warehouse
  RUNSQLCFG.SQL
                                                                              if exists ( select name from sysobjects where name = 'district' )
                                                                                         drop table district
/* This script file is used to set runtime server configuration parameters */
```

```
char(9),
if exists ( select name from sysobjects where name = 'customer' )
                                                                                            c_zip
           drop table customer
                                                                                             c_phone
                                                                                                                                         char(16),
                                                                                            c_since
                                                                                                                                         datetime,
if exists ( select name from sysobjects where name = 'history' )
                                                                                             c_credit
                                                                                                                              char(2),
           drop table history
                                                                                            c\_credit\_lim
                                                                                                                              numeric(12,2),
                                                                                             c discount
                                                                                                                              numeric(4,4),
if exists (select name from sysobjects where name = 'new order')
                                                                                             c balance
                                                                                                                              numeric(12,2),
           drop table new_order
                                                                                            c_ytd_payment
                                                                                                                              numeric(12,2),
                                                                                             c_payment_cnt
                                                                                                                              smallint,
                                                                                            c_delivery_cnt
if exists ( select name from sysobjects where name = 'orders' )
                                                                                                                              smallint,
           drop table orders
                                                                                             c_data
                                                                                                                                         char(500)
                                                                                 ) on MSSQL_cs_fg
if exists (select name from sysobjects where name = 'order line')
           drop table order_line
                                                                                 create table history
if exists ( select name from sysobjects where name = 'item' )
           drop table item
                                                                                            h c id
                                                                                                                                         int.
                                                                                            h_c_d_id
                                                                                                                              tinyint,
go
if exists ( select name from sysobjects where name = 'stock' )
                                                                                                                              smallint,
                                                                                            h_c_w_id
           drop table stock
                                                                                            h d id
                                                                                                                                         tinyint,
                                                                                            h_w_id
                                                                                                                                         smallint,
go
                                                                                            h date
                                                                                                                                         datetime,
                                                                                            h_amount
                                                                                                                              numeric(6,2),
-- Create new tables
                                                                                            h data
                                                                                                                                         char(24)
                                                                                 ) on MSSQL_misc_fg
create table warehouse
                                                                                 create table new_order
           w id
                                                       smallint,
           w_name
                                                       char(10),
                                                                                            no_o_id
                                                                                                                                         int,
           w_street 1
                                            char(20),
                                                                                            no d id
                                                                                                                                         tinvint,
           w_street_2
                                            char(20),
                                                                                             no_w_id
                                                                                                                                         smallint
                                                                                 ) on MSSQL_misc_fg
           w_city
                                                       char(20),
           w state
                                                       char(2),
           w_zip
                                                       char(9),
                                                       numeric(4,4),
                                                                                 create table orders
           w_tax
           w_ytd
                                                       numeric(12,2)
) on MSSQL misc fg
                                                                                             o id
                                                                                                                                         int,
                                                                                             o_d_id
                                                                                                                                         tinyint,
                                                                                             o_w_id
                                                                                                                                         smallint,
create table district
                                                                                             o c id
                                                                                                                                         int,
                                                                                             o_entry_d
                                                                                                                              datetime.
           d id
                                                                                             o carrier id
                                                                                                                              tinyint,
                                                       tinyint,
                                                                                                                              tinyint,
           d_w_id
                                                       smallint,
                                                                                             o_ol_cnt
           d name
                                                       char(10),
                                                                                            o all local
                                                                                                                              tinyint
           d_street_1
                                            char(20),
                                                                                 ) on MSSQL_misc_fg
           d street 2
                                            char(20),
           d city
                                                       char(20),
           d_state
                                                                                 create table order_line
                                                       char(2),
           d zip
                                                       char(9),
           d_tax
                                                       numeric(4,4),
                                                                                             ol_o_id
                                                                                                                                         int,
                                                                                            ol d id
           d ytd
                                                       numeric(12,2),
                                                                                                                                         tinyint,
                                                                                                                                         smallint,
           d_next_o_id
                                                       int
                                                                                             ol_w_id
) on MSSQL_misc_fg
                                                                                             ol_number
                                                                                                                              tinyint,
                                                                                             ol i id
                                                                                                                                         int,
                                                                                            ol_supply_w_id
                                                                                                                              smallint,
create table customer
                                                                                             ol delivery d
                                                                                                                              datetime,
                                                                                             ol_quantity
                                                                                                                                         smallint,
           c id
                                                       int,
                                                                                             ol amount
                                                                                                                              numeric(6,2),
           c_d_id
                                                       tinyint,
                                                                                            ol_dist_info
                                                                                                                              char(24)
           c_w_id
                                                       smallint,
                                                                                 ) on MSSQL_misc_fg
           c first
                                                       char(16),
           c_middle
                                            char(2),
           c_last
                                                       char(16),
                                                                                 create table item
                                            char(20),
           c_street_1
           c street 2
                                            char(20),
                                                                                            i id
                                                                                                                                         int,
                                                       char(20),
           c_city
                                                                                            i_im_id
                                                                                                                                         int,
           c_state
                                                       char(2),
                                                                                             i name
                                                                                                                                         char(24),
```

```
i_price
                                                  numeric(5,2),
                                                                         go
          i_data
                                                  char(50)
                                                                         print
) on MSSQL_misc_fg
                                                                                   'CUSTOMER TABLE = (30,000 * No of warehouses)'
                                                                         select
                                                                                   rows
create table stock
                                                                         from
                                                                                   sysindexes
                                                                                             =object_id("customer")
                                                                         where
                                                                                   id
          s i id
                                                  int.
          s_w_id
                                                  smallint,
                                                                         print 'ORDERS TABLE = (30,000 * No of warehouses)'
                                        smallint,
          s_quantity
          s_dist_01
                                        char(24),
          s_dist_02
                                        char(24),
                                                                         select
                                                                                   rows
          s dist 03
                                        char(24),
                                                                         from
                                                                                   sysindexes
          s_dist_04
                                        char(24),
                                                                         where
                                                                                             =object_id("orders")
                                                                                   id
          s dist 05
                                        char(24),
                                                                         go
          s_dist_06
                                        char(24),
                                                                                   'HISTORY TABLE = (30,000 * No of warehouses)'
          s dist 07
                                        char(24),
                                                                         print
          s_dist_08
                                        char(24),
          s_dist 09
                                        char(24),
                                                                         select
          s dist 10
                                        char(24),
                                                                         from
                                                                                   sysindexes
          s_ytd
                                                                         where
                                                                                             =object_id("history")
                                                  int.
          s order cnt
                                                  smallint,
                                                                         go
          s_remote_cnt
                                        smallint,
                                                                         print
                                                                                   'STOCK TABLE = (100,000 * No of warehouses)'
                                                  char(50)
          s data
) on MSSQL_cs_fg
                                                                         select
                                                                         from
                                                                                   sysindexes
                                                                                             =object_id("stock")
                                                                                   id
                                                                         where
Verify_TpccLoad.sql
                                                                         go
                                                                         print
                                                                                   'ORDER LINE TABLE = (300,000 * No of warehouses + some
-- File: VERIFYTPCCLOAD.SQL
                                                                         change)'
        Microsoft TPC-C Benchmark Kit Ver. 4.22
        Copyright Microsoft, 2001
                                                                         select
                                                                                   rows
-- Purpose: Performs series of TPCC database checks to verify
                                                                         from
                                                                                   sysindexes
             that database load completed correctly
                                                                         where
                                                                                             =object id("order line")
                                                                         go
print
          convert(char(30), getdate(),9)
select
                                                                                   'NEW_ORDER TABLE = (9000 * No of warehouses)'
                                                                         print
print
                                                                         select
                                                                                   rows
use tpcc
                                                                         from
                                                                                   sysindexes
go
                                                                                             =object_id("new_order")
                                                                         where
                                                                         go
          ***********
                                                                                    ******
          Check rows per table from SYSINDEXES
                                                                                   Check indices
          ***********
                                                                                   ******
print
          'WAREHOUSE TABLE'
                                                                         print '***********Index Check************
          rows
select
from
          sysindexes
                                                                         use tpcc
                    = object_id("warehouse")
where
go
                                                                         sp_helpindex
                                                                                             customer
print
          'DISTRICT TABLE = (10 * No of warehouses)'
select
          rows
                                                                         sp_helpindex
                                                                                             stock
from
          sysindexes
                                                                         go
                    =object_id("district")
where
go
                                                                         sp_helpindex
                                                                                             district
                                                                         go
          'ITEM TABLE = 100,000'
print
                                                                         sp_helpindex
                                                                                             item
select
          rows
                                                                         go
from
          sysindexes
          id
                    =object_id("item")
where
```

```
sp_helpindex go

sp_helpindex go

sp_helpindex go

sp_helpindex go

sp_helpindex warehouse go
```

version.sql

```
-- File: VERSION.SQL
         Microsoft TPC-C Benchmark Kit Ver. 4.22
        Copyright Microsoft, 2001
-- Purpose: Returns version level of TPC-C stored procs
-- Note: Always update the return value of this proc for
        any interface changes or "must have" bug fixes.
-- The value returned by this SP defines the "interface level",
-- which must match between the stored procs and the client code.
-- The interface level may be down rev from the current kit. This
-- indicates that the interface hasn't changed since that version.
use tpcc
go
if exists ( select name from sysobjects where name = "tpcc version" )
           drop procedure tpcc_version
create proc tpcc_version
declare
           @version char(8)
begin
           select @version = "4.10.000"
           select @version as "Version"
end
go
```

Load Source Code

getargs.c

```
void GetArgsLoader(int argc, char **argv, TPCCLDR_ARGS *pargs)
  int
  char
          *ptr;
#ifdef DEBUG
  printf("[%ld]DBG: Entering GetArgsLoader()\n", (int)
GetCurrentThreadId());
#endif
  /* init args struct with some useful values */
  pargs->server
                              = SERVER;
  pargs->user
                              = USER;
                                        = PASSWORD;
  pargs->password
  pargs->database
                                        = DATABASE;
  pargs->batch
                              = BATCH;
  pargs->num warehouses
                                        = UNDEF;
         pargs->tables_all
                                                  = TRUE;
          pargs->table item
                                                  = FALSE;
          pargs->table warehouse
                                                  = FALSE;
         pargs->table customer
                                                  = FALSE;
         pargs->table_orders
                                                  = FALSE;
          pargs->loader_res_file
LOADER RES FILE;
          pargs->pack_size
                                        = DEFLDPACKSIZE;
          pargs->starting warehouse
DEF_STARTING_WAREHOUSE;
          pargs->build index
                                                  = BUILD INDEX;
                                                  = INDEX_ORDER;
          pargs->index_order
         pargs->index_script_path
                                        = INDEX_SCRIPT_PATH;
         pargs->scale down
                                                  = SCALE DOWN;
  /* check for zero command line args */
  if ( argc == 1 )
                    GetArgsLoaderUsage();
  for (i = 1; i < argc; ++i)
                    if (argv[i][0] != '-' && argv[i][0] != '/')
                    printf("\nUnrecognized command");
                    GetArgsLoaderUsage();
                    exit(1);
                    ptr = argv[i];
                    switch (ptr[1])
                    case 'h': /* Fall throught */
                    case 'H':
                                        GetArgsLoaderUsage();
                                        break;
                    case 'D'
                                        pargs->database = ptr+2;
                                        break;
                    case 'P':
                                        pargs->password = ptr+2;
                                        break;
                    case 'S':
                                        pargs->server = ptr+2;
                                        break;
```

```
break;
                     case 'U':
                                           pargs->user = ptr+2;
                                                                                                   case 'd':
                                           break;
                                                                                                                         pargs->index_script_path = ptr+2;
                                                                                                                         break:
                     case 'b':
                                                                                                   default:
                                           pargs->batch = atol(ptr+2);
                                           break;
                                                                                                              GetArgsLoaderUsage();
                                                                                                              exit(-1);
                     case 'W':
                                                                                                              break;
                                           pargs->num_warehouses =
atol(ptr+2);
                                           break;
                                                                                 /* check for required args */
                                                                                 if (pargs->num_warehouses == UNDEF)
                     case 's':
                                           pargs->starting_warehouse =
atol(ptr+2);
                                                                                                   printf("Number of Warehouses is required\n");
                                          break;
                                                                                                   exit(-2);
                     case 't':
                                                                                return;
                                                     pargs->tables all =
FALSE;
                                                     if (strcmp(ptr+2,"item")
== 0)
pargs->table item = TRUE;
                                                     else if
                                                                              // Function name: GetArgsLoaderUsage
(strcmp(ptr+2,"warehouse") == 0)
pargs->table warehouse = TRUE;
                                                     else if
(strcmp(ptr+2,"customer") == 0)
                                                                              void GetArgsLoaderUsage()
pargs->table_customer = TRUE;
                                                     else if
                                                                              #ifdef DEBUG
                                                                                printf("[%ld]DBG: Entering GetArgsLoaderUsage()\n", (int)
(strcmp(ptr+2,"orders") == 0)
                                                                              GetCurrentThreadId());
pargs->table_orders = TRUE;
                                                                              #endif
                                                     else
                                                                                 printf("TPCCLDR:\n\n");
                                                     printf("\nUnrecognized
command");
                                                                                         printf("Parameter
                                                                                                                                    Default\n");
                                                                                         printf("-----
                                                     GetArgsLoaderUsage();
                                                                                                                                      ----\n");
                                                                                 printf("-W Number of Warehouses to Load
                                                                                                                                       Required \n");
                                                     exit(1);
                                                                                 printf("-S Server
                                                                                                                           %s\n", SERVER);
                                                                                printf("-U Username
                                                                                                                             %s\n", USER);
                                                     break;
                                                                                printf("-P Password
                                                                                                                             %s\n", PASSWORD);
                                                                                                                             %s\n", DATABASE);
                                                                                printf("-D Database
                                          }
                                                                                         printf("-b Batch Size
                                                                                                                                     %ld\n", (long)
                                case 'f':
                                                                              BATCH);
                                                                                                                                       %ld\n", (long)
                                                                                         printf("-p TDS packet size
                                          pargs->loader res file = ptr+2;
                                                                              DEFLDPACKSIZE);
                                          break;
                                                                                         printf("-f Loader Results Output Filename
                                                                                                                                             %s\n",
                                                                              LOADER_RES_FILE);
                                case 'p':
                                                                                         printf("-s Starting Warehouse
                                                                                                                                        %ld\n", (long)
                                           pargs->pack_size = atol(ptr+2);
                                                                              DEF STARTING WAREHOUSE);
                                           break;
                                                                                         printf("-i Build Option (data = 0, data and index = 1)
                                                                                                                                               %ld\n"
                                case 'i':
                                                                              (long) BUILD INDEX);
                                           pargs->build_index = atol(ptr+2);
                                                                                         printf("-o Cluster Index Build Order (before = 1, after = 0) %ld\n",
                                                                              (long) INDEX ORDER);
                                           break:
                                                                                         printf("-c Build Scaled Database (normal = 0, tiny = 1)
                                                                                                                                                 %ld\n",
                                                                              (long) SCALE_DOWN);
                                case 'o':
                                           pargs->index_order = atol(ptr+2);
                                                                                         printf("-d Index Script Path
                                                                                                                                       %s\n",
                                                                              INDEX_SCRIPT_PATH);
                                          break:
                                                                                         printf("-t Table to Load
                                                                                                                                      all tables \n");
                                                                                 printf(" [item|warehouse|customer|orders]\n");
                                case 'c':
                                           pargs->scale down = atol(ptr+2);
                                                                                printf(" Notes: \n");
```

```
printf("
           - the '-t' parameter may be included multiple times to \n");
                                                                            printf("[%ld]DBG: Entering seed()...\n", (int) GetCurrentThreadId());
  printf("
            specify multiple tables to be loaded \n");
                                                                                    printf("Old Seed %ld New Seed %ld\n", Seed, val);
  printf("
           - 'item' loads ITEM table \n");
                                                                          #endif
           - 'warehouse' loads WAREHOUSE, DISTRICT, and STOCK
  printf("
                                                                            if (val < 0)
tables \n");
  printf("
            - 'customer' loads CUSTOMER and HISTORY tables \n");
                                                                                      val = abs(val);
           - 'orders' load NEW-ORDER, ORDERS, ORDER-LINE tables
  printf("
                                                                            Seed = val;
          printf("\nNote: Command line switches are case sensitive.\n");
  exit(0);
                                                                            irand - returns a 32 bit integer pseudo random number with a period of *
random.c
                                                                               1 to 2 ^ 32 - 1.
                              RANDOM.C
                                                                          * parameters:
                                        Microsoft TPC-C Kit Ver. 4.22
//
                                                                               none.
//
                                        Copyright Microsoft, 1996, 1997,
1998, 1999, 2000, 2001
          Purpose: Random number generation routines for database loader
                                                                               32 bit integer - defined as long ( see above ).
                                                                          * side effects:
// Includes
                                                                               seed get recomputed.
#include "tpcc.h"
#include "math.h"
// Defines
                                                                          long irand()
#define A
               16807
#define M
             2147483647
                                                                            register long s; /* copy of seed */
              127773 /* M div A */
#define Q
                                                                            register long test; /* test flag */
                2836 /* M mod A */
#define R
                                                                            register long hi; /* tmp value for speed */
                    __declspec(thread)
#define Thread
                                                                            register long lo; /* tmp value for speed */
// Globals
                                                                          #ifdef DEBUG
          Thread Seed = 0; /* thread local seed */
                                                                            printf("[%ld]DBG: Entering irand()...\n", (int) GetCurrentThreadId());
                                                                            s = Seed;
                                                                            hi = s / Q;
* random -
                                                                            lo = s \% Q;
     Implements a GOOD pseudo random number generator. This generator
                                                                            test = A * lo - R * hi;
     will/should? run the complete period before repeating.
                                                                            if (test > 0)
                                                                                              Seed = test;
* Copied from:
     Random Numbers Generators: Good Ones Are Hard to Find.
                                                                                              Seed = test + M;
     Communications of the ACM - October 1988 Volume 31 Number 10
                                                                            return( Seed );
* Machine Dependencies:
     long must be 2 ^ 31 - 1 or greater.
                                                                                             **************
                                                                          * drand - returns a double pseudo random number between 0.0 and 1.0.
/*********************
* seed - load the Seed value used in irand and drand. Should be used before *
                                                                          **********
     first call to irand or drand.
                                                                          double drand()
********************
                                                                          #ifdef DEBUG
void seed(long val)
                                                                            printf("[%ld]DBG: Entering drand()...\n", (int) GetCurrentThreadId());
#ifdef DEBUG
```

```
(int) GetCurrentThreadId(), lower,
  return( (double)irand() / 2147483647.0);
                                                                             upper, rand_num);
                                                                             #endif
                                                                               return rand num;
// Function : RandomNumber
                                                                             #endif
// Description:
long RandomNumber(long lower, long upper)
                                                                             // Function : NURand
  long rand_num;
                                                                             // Description:
#ifdef DEBUG
  printf("[%ld]DBG: Entering RandomNumber()...\n", (int)
                                                                             long NURand(int iConst,
GetCurrentThreadId());
                                                                                          long x,
#endif
                                                                                          long y
                                                                                          long C)
          if (upper == lower) /* pgd 08-13-96 perf enhancement */
                                                                               long rand num;
                    return lower;
          upper++;
                                                                             #ifdef DEBUG
                                                                               printf("[%ld]DBG: Entering NURand()...\n", (int) GetCurrentThreadId());
  if ( upper <= lower )
                    rand num = upper;
                                                                               rand_num = (((RandomNumber(0,iConst) | RandomNumber(x,y)) + C) %
  else
                    rand num = lower + irand() % (upper - lower); /* pgd
                                                                             (y-x+1)+x;
08-13-96 perf enhancement */
                                                                             #ifdef DEBUG
                                                                               printf("[%ld]DBG: NURand: num = %d\n", (int) GetCurrentThreadId(),
#ifdef DEBUG
                                                                             rand num);
  printf("[%ld]DBG: RandomNumber between %ld & %ld ==> %ld\n",
                                                                             #endif
                                         (int) GetCurrentThreadId(), lower,
upper, rand_num);
                                                                               return rand num;
#endif
  return rand num:
                                                                             strings.c
                                                                                                            STRINGS.C
                                                                                       File:
                                                                                                                       Microsoft TPC-C Kit Ver. 4.22
#if 0
                                                                                                                       Copyright Microsoft, 1996, 1997,
                                                                             1998, 1999, 2000, 2001
//Orginal code pgd 08/13/96
                                                                                       Purpose: Source file for database loader string functions
long RandomNumber(long lower,
                                           long upper)
                                                                             // Includes
                                                                             #include "tpcc.h"
  long rand_num;
                                                                             #include <string.h>
                                                                             #include <ctype.h>
#ifdef DEBUG
  printf("[%ld]DBG: Entering RandomNumber()...\n", (int)
GetCurrentThreadId());
#endif
                                                                             // Function name: MakeAddress
          upper++;
  if ((upper <= lower))
                    rand num = upper;
  else
                                                                             void MakeAddress(char *street 1,
                    rand_num = lower + irand() % ((upper > lower) ? upper -
                                                                                                                        char *street 2,
lower: upper);
                                                                                                                        char *city,
                                                                                                                        char *state,
                                                                                                                       char *zip)
#ifdef DEBUG
  printf("[%ld]DBG: RandomNumber between %ld & %ld ==> %ld\n",
```

```
#ifdef DEBUG
  printf("[%ld]DBG: Entering MakeAddress()\n", (int) GetCurrentThreadId());
  MakeAlphaString (10, 20, ADDRESS LEN, street 1);
  MakeAlphaString (10, 20, ADDRESS LEN, street 2);
  MakeAlphaString (10, 20, ADDRESS_LEN, city);
  MakeAlphaString (2, 2, STATE_LEN, state);
  MakeZipNumberString(9, 9, ZIP_LEN, zip);
#ifdef DEBUG
  printf("[%ld]DBG: MakeAddress: street_1: %s, street_2: %s, city: %s, state:
%s, zip: %s\n",
                                (int) GetCurrentThreadId(), street_1, street_2,
city, state, zip);
#endif
  return;
// Function name: LastName
void LastName(int num,
                         char *name)
  static char *n[] =
                     "BAR", "OUGHT", "ABLE", "PRI", "PRES", "ESE", "ANTI", "CALLY", "ATION", "EING"
  };
#ifdef DEBUG
  printf("[%ld]DBG: Entering LastName()\n", (int) GetCurrentThreadId());
#endif
  if ((num \ge 0) && (num < 1000))
                     strcpy(name, n[(num/100)%10]);
                     strcat(name, n[(num/10)%10]);
                     strcat(name, n[(num/1)\%10]);
                     if (strlen(name) < LAST NAME LEN)
                                PaddString(LAST NAME LEN, name);
  else
                     printf("\nError in LastName()... num <%ld> out of range
(0,999)\n", num);
                     exit(-1);
#ifdef DEBUG
  printf("[%ld]DBG: LastName: num = [%d] ==> [%d][%d][%d]\n",
                                (int) GetCurrentThreadId(), num, num/100,
(num/10)%10, num%10);
```

```
printf("[%ld]DBG: LastName: String = %s\n", (int)
GetCurrentThreadId(), name);
#endif
  return:
// Function name: MakeAlphaString
//philipdu 08/13/96 Changed MakeAlphaString to use A-Z, a-z, and 0-9 in
//accordance with spec see below:
//The spec says:
//4.3.2.2 The notation random a-string [x .. y]
//(respectively, n-string [x .. y]) represents a string of random alphanumeric
//(respectively, numeric) characters of a random length of minimum x,
maximum v.
//and mean (y+x)/2. Alphanumerics are A..Z, a..z, and 0..9. The only other
//requirement is that the character set used "must be able to represent a
minimum
//of 128 different characters". We are using 8-bit chars, so this is a non issue.
//It is completely unreasonable to stuff non-printing chars into the text fields.
//-CLevine 08/13/96
int MakeAlphaString( int x, int y, int z, char *str)
                                 len:
           int
                                 i;
  char cc = 'a';
           static
                      char chArray[] =
"0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopgrstu
vwxyz";
                                 chArrayMax = 61;
           static
                      int
#ifdef DEBUG
  printf("[%ld]DBG: Entering MakeAlphaString()\n", (int)
GetCurrentThreadId());
#endif
           len= RandomNumber(x, y);
           for (i=0; i<len; i++)
           {
                      cc = chArray[RandomNumber(0, chArrayMax)];
     str[i] = cc;
           if (len < z)
                      memset(str+len, '', z - len);
           str[len] = 0;
  return len:
// Function name: MakeOriginalAlphaString
```

```
itoa(RandomNumber(0, 9999999), tmp, 10);
int MakeOriginalAlphaString(int x,
                                                                                            memcpy(str, tmp, strlen(tmp));
                                                                    int y,
                                                                                            itoa(RandomNumber(0, 99999999), tmp, 10);
                                                                                           memcpy(str+8, tmp, strlen(tmp));
int z,
                                                                    char *str,
                                                                  int percent)
                                                                                           str[16] = 0;
  int
                      len;
                                                                                   return 16;
                      val;
  int
  int
                      start;
#ifdef DEBUG
  printf("[%ld]DBG: Entering MakeOriginalAlphaString()\n", (int)
GetCurrentThreadId());
#endif
                                                                                 // Function name: MakeZipNumberString
  // verify prercentage is valid
  if ((percent < 0) || (percent > 100))
                                                                                 int MakeZipNumberString(int x, int y, int z, char *str)
                      printf("MakeOrigianlAlphaString: Invalid percentage:
%d\n", percent);
                                                                                            char tmp[16];
                      exit(-1);
                                                                                           //MakeZipNumberString is always called MakeZipNumberString(9,
                                                                                 9, 9, string)
  // verify string is at least 8 chars in length
  if ((x + y) \le 8)
                                                                                            strcpy(str, "000011111");
                      printf("MakeOriginalAlphaString: string length must be
                                                                                            itoa(RandomNumber(0, 9999), tmp, 10);
>= 8\n");
                                                                                           memcpy(str, tmp, strlen(tmp));
                      exit(-1);
                                                                                   return 9;
  // Make Alpha String
  len = MakeAlphaString(x,y, z, str);
  val = RandomNumber(1,100);
  if (val <= percent)
                                                                                 // Function name: InitString
                      start = RandomNumber(0, len - 8);
                      strncpy(str + start, "ORIGINAL", 8);
                                                                                 void InitString(char *str, int len)
#ifdef DEBUG
                                                                                 #ifdef DEBUG
  printf("[%ld]DBG: MakeOriginalAlphaString: : %s\n",
                                                                                   printf("[%ld]DBG: Entering InitString()\n", (int) GetCurrentThreadId());
                                 (int) GetCurrentThreadId(), str);
                                                                                           memset(str, ' ', len);
  return strlen(str);
                                                                                   str[len] = 0;
                                                                                 // Function name: InitAddress
// Function name: MakeNumberString
//
                                                                                 // Description:
int MakeNumberString(int x, int y, int z, char *str)
           char tmp[16];
                                                                                 void InitAddress(char *street_1, char *street_2, char *city, char *state, char
                                                                                 *zip)
           //MakeNumberString is always called MakeZipNumberString(16,
16, 16, string)
                                                                                           memset(street 1, '', ADDRESS LEN+1);
                                                                                           memset(street_2, '', ADDRESS_LEN+1);
                                                                                           memset(city, '', ADDRESS_LEN+1);
           memset(str, '0', 16);
```

```
street 1[ADDRESS LEN+1] = 0;
                                                                             time now = ((el time.time - start sec) * 1000) + el time.millitm;
  street_2[ADDRESS_LEN+1] = 0;
  city[ADDRESS\_LEN+1] = 0;
                                                                             return time_now;
          memset(state, '', STATE LEN+1);
  state[STATE LEN+1] = 0;
                                                                           tpcc.h
          memset(zip, '', ZIP_LEN+1);
  zip[ZIP LEN+1] = 0;
                                                                                                          TPCC.H
                                                                                     File:
                                                                                                                    Microsoft TPC-C Kit Ver. 4.22
                                                                                                                    Copyright Microsoft, 1996, 1997,
                                                                           1998, 1999, 2000, 2001
                                                                                     Purpose: Header file for TPC-C database loader
//
// Function name: PaddString
                                                                           // Build number of TPC Benchmark Kit
                                                                           #define TPCKIT_VER
                                                                           // General headers
                                                                           #include <windows.h>
void PaddString(int max, char *name)
                                                                           #include <winbase.h>
                                                                           #include <stdlib.h>
          int
                              len:
                                                                           #include <stdio.h>
                                                                           #include cess.h>
          len = strlen(name);
                                                                           #include <stddef.h>
          if (len < max)
                                                                           #include <stdarg.h>
                    memset(name+len, '', max - len);
                                                                           #include <string.h>
          name[max] = 0;
                                                                           #include <time.h>
                                                                           #include <sys\timeb.h>
          return;
                                                                           #include <sys\types.h>
                                                                           // ODBC headers
                                                                           #include <sql.h>
time.c
                                                                           #include <sqlext.h>
                                                                           #include <odbcss.h>
//
          File:
                              TIME.C
                                         Microsoft TPC-C Kit Ver. 4.22
//
                                                                           // General constants
                                        Copyright Microsoft, 1996, 1997,
                                                                           #define MILLI
                                                                                                          1000
1998, 1999, 2000, 2001
                                                                           #define FALSE
                                                                                                                              0
                                                                           #define TRUE
          Purpose: Source file for time functions
                                                                           #define UNDEF
                                                                                                                                          -1
                                                                           #define MINPRINTASCII
                                                                                                                                32
// Includes
                                                                           #define MAXPRINTASCII
                                                                                                                               126
#include "tpcc.h"
                                                                           // Default environment constants
// Globals
                                                                           #define SERVER
                                                                                                                              "tpcc"
static long start sec;
                                                                           #define DATABASE
                                                                           #define USER
                                                                                                                    "sa"
                                                                           #define PASSWORD
                                                                           // Default loader arguments
                                                                           #define BATCH
                                                                                                                    10000
                                                                           #define DEFLDPACKSIZE
// Function name: TimeNow
                                                                                                                              32768
                                                                           #define LOADER RES FILE
                                                                                                                     "logs\\load.out"
                                                                           #define LOADER NURAND C
                                                                                                                               123
                                                                           #define DEF_STARTING_WAREHOUSE
                                                                                                                              1
                                                                           #define BUILD_INDEX
                                                                                                                                                   1
long TimeNow()
                                                                           // build both data and indexes
                                                                           #define INDEX_ORDER
                    time now;
                                                                           // build indexes before load
 long
                                                                           #define SCALE DOWN
 struct
         timeb el time;
                                                                                                                                                   0
                                                                            // build a normal scale database
#ifdef DEBUG
                                                                           #define INDEX_SCRIPT_PATH "scripts"
  printf("[%ld]DBG: Entering TimeNow()\n", (int) GetCurrentThreadId());
                                                                           typedef struct
#endif
 _ftime(&el_time);
                                                                             char
                                                                                                                    *server:
```

```
*database;
  char
                                                                        short
                                                                                  WURand();
  char
                                       *user;
                                                                        long
                                                                                  RandomNumber(long lower, long upper);
  char
                                        *password:
                                                                        // Functions in getargs.c;
          BOOL
                                                 tables_all;
// set if loading all tables
                                                                                  GetArgsLoader();
                                                                        void
          BOOL
                                                                                  GetArgsLoaderUsage();
                                                 table item;
                                                                        void
// set if loading ITEM table specifically
         BOOL
                                                 table warehouse;
                                                                        // Functions in time.c
set if loading WAREHOUSE, DISTRICT, and STOCK
                                                                        long
                                                                                  TimeNow();
         BOOL
                                                 table customer;
// set if loading CUSTOMER and HISTORY
                                                                        // Functions in strings.c
         BOOL
                                                                                  MakeAddress();
                                                 table orders;
                                                                        void
// set if loading NEW-ORDER, ORDERS, ORDER-LINE
                                                                        void
                                                                                  LastName();
                                                                                  MakeAlphaString();
                                       num_warehouses;
                                                                        int
                                       batch;
                                                                                  MakeOriginalAlphaString();
  long
                                                                        int
                                                                                  MakeNumberString();
  long
                                       verbose;
                                                                        int
         long
                                                 pack size:
                                                                        int
                                                                                  MakeZipNumberString();
          char
                                                 *loader res file;
                                                                                  InitString();
                                                                        void
                                                 *synch servername;
                                                                                  InitAddress();
         char
                                                                        void
                                                 case sensitivity;
                                                                                  PaddString();
          long
                                                                        void
                                                 starting_warehouse;
          long
                                                 build index;
          long
          long
                                                 index_order;
                                                                        tpccldr.c
                                                 scale down;
         long
          char
                                                 *index_script_path;
                                                                                  File:
                                                                                                      TPCCLDR.C
} TPCCLDR_ARGS;
                                                                                                               Microsoft TPC-C Kit Ver. 4.22
                                                                        //
                                                                                                               Copyright Microsoft, 2000, 2001
// String length constants
                                                                        //
                                                                                  Purpose: Source file for TPC-C database loader
         #define SERVER NAME LEN
                                            20
          #define DATABASE_NAME_LEN
                                              20
          #define USER NAME LEN
                                           20
                                                                        // Includes
          #define PASSWORD LEN
                                          20
                                                                        #include "tpcc.h"
          #define TABLE NAME LEN
                                           20
                                                                        #include "search.h"
          #define I DATA LEN
                                       50
          #define I_NAME_LEN
                                       24
                                                                        // Defines
         #define BRAND LEN
                                        1
                                                                        #define MAXITEMS
                                                                                                    100000
          #define LAST_NAME_LEN
                                          16
                                                                        #define MAXITEMS SCALE DOWN
                                                                                                                         100
          #define W NAME LEN
                                         10
                                                                        #define CUSTOMERS PER DISTRICT 3000
          #define ADDRESS_LEN
                                        20
                                                                        #define CUSTOMERS SCALE DOWN
          #define STATE LEN
                                       2
                                                                        #define DISTRICT_PER_WAREHOUSE 10
          #define ZIP LEN
                                                                        #define ORDERS_PER_DISTRICT #define ORDERS_SCALE_DOWN
          #define S DIST LEN
                                      24
          #define S DATA LEN
                                        50
                                                                        #define MAX CUSTOMER THREADS 2
          #define D_NAME_LEN
                                        10
                                                                        #define MAX_ORDER_THREADS
         #define FIRST NAME LEN
                                                                        #define MAX_MAIN_THREADS
          #define MIDDLE_NAME_LEN
          #define PHONE LEN
                                       16
                                                                        // Functions declarations
         #define CREDIT LEN
          #define C_DATA_LEN
                                       500
                                                                        void HandleErrorDBC (SQLHDBC hdbc1);
         #define H DATA LEN
                                        24
         #define DIST_INFO_LEN
                                        24
                                                                        void CheckSQL();
          #define MAX OL NEW ORDER ITEMS
                                                                        void CheckDataBase();
          #define MAX_OL_ORDER_STATUS_ITEMS 15
          #define STATUS_LEN
                                       25
                                                                        long NURand();
          #define OL DIST INFO LEN
                                                  24
                                                                        void LoadItem();
          #define C_SINCE_LEN
                                                                        void LoadWarehouse();
23
          #define H_DATE_LEN
                                                                        void Stock();
23
                                                                        void District();
          #define OL_DELIVERY_D_LEN
                                                            23
          #define O_ENTRY_D_LEN
                                          23
                                                                        void LoadCustomer();
                                                                        void CustomerBufInit();
                                                                        void CustomerBufLoad();
// Functions in random.c
                                                                        void LoadCustomerTable();
void
          seed();
                                                                        void LoadHistoryTable();
long
          irand();
double
          drand();
                                                                        void LoadOrders();
void
         WUCreate();
                                                                        void OrdersBufInit();
```

```
void OrdersBufLoad();
                                                                        typedef struct
void LoadOrdersTable();
void LoadNewOrderTable();
                                                                                  char
void LoadOrderLineTable();
                                                                        c_last[LAST_NAME_LEN+1];
void GetPermutation();
                                                                                  char
void CheckForCommit();
                                                                        c first[FIRST NAME LEN+1];
void OpenConnections();
                                                                                                                c_id;
void BuildIndex();
                                                                        } CUSTOMER_SORT_STRUCT;
void FormatDate ();
                                                                        typedef struct
// Shared memory structures
                                                                                       time start;
                                                                          long
typedef struct
                                                                        } LOADER TIME STRUCT;
  long
              ol;
                                                                        // Global variables
  long
              ol_i_id;
  short
              ol_supply_w_id;
  short
              ol_quantity;
                                                                        char
                                                                                  szLastError[300];
  double
               ol amount;
              ol dist info[DIST INFO LEN+1];
                                                                        HENV
  char
                                                                                  henv:
         char
ol delivery d[OL DELIVERY D LEN+1];
                                                                        HDBC
                                                                                  v hdbc;
                                                                                                                                             //
} ORDER_LINE_STRUCT;
                                                                        for SQL Server version verification
                                                                        HDBC
                                                                                                                                   // for ITEM
                                                                                  i hdbc1;
typedef struct
                                                                        table
                                                                        HDBC
                                                                                                                                   // for
                                                                                  w hdbc1;
                                                                        WAREHOUSE, DISTRICT, STOCK
              o id;
  long
              o_d_id;
                                                                        HDBC
                                                                                                                                   // for
  short
                                                                                  c_hdbc1;
  short
              o w id;
                                                                        CUSTOMER
                                                                        HDBC
  long
              o_c_id;
                                                                                  c_hdbc2;
                                                                                                                                   // for
  short
              o carrier id;
                                                                        HISTORY
  short
              o_ol_cnt;
                                                                        HDBC
                                                                                  o_hdbc1;
                                                                                                                                   // for
              o all local;
                                                                        ORDERS
  short
          ORDER LINE STRUCT o ol[15];
                                                                        HDBC
                                                                                  o hdbc2;
                                                                                                                                   // for
} ORDERS_STRUCT;
                                                                        NEW-ORDER
                                                                        HDBC
                                                                                  o hdbc3;
                                                                                                                                   // for
                                                                        ORDER-LINE
typedef struct
                                                                        HSTMT v_hstmt;
                                                                                                                                   // for SQL
  long
                                       c_id;
                                                                        Server version verification
  short
                                       c_d_id;
                                       c w id;
                                                                        HSTMT
                                                                                  i hstmt1;
                                                                        HSTMT
          char
                                                                                 w hstmt1:
c first[FIRST NAME LEN+1];
                                                                        HSTMT
                                                                                  c hstmt1, c hstmt2;
                                                                        HSTMT o_hstmt1, o_hstmt2, o_hstmt3;
  char
c middle[MIDDLE NAME LEN+1];
                                       c_last[LAST_NAME_LEN+1];
                                                                        ORDERS_STRUCT orders_buf[ORDERS_PER_DISTRICT];
  char
  char
                                       c street 1[ADDRESS LEN+1];
                                                                        CUSTOMER_STRUCT customer_buf[CUSTOMERS_PER_DISTRICT];
  char
                                       c street 2[ADDRESS LEN+1];
                                                                                  orders rows loaded;
                                       c_city[ADDRESS_LEN+1];
                                                                                  new_order_rows_loaded;
  char
                                                                        long
                                       c state[STATE LEN+1];
                                                                                  order line rows loaded;
  char
                                                                        long
                                       c_zip[ZIP_LEN+1];
                                                                                  history_rows_loaded;
  char
                                                                        long
                                       c phone[PHONE LEN+1];
                                                                                  customer rows loaded;
  char
                                                                        long
                                       c_credit[CREDIT_LEN+1];
                                                                                  stock_rows_loaded;
  char
                                                                        long
                                       c credit lim;
  double
                                                                                   district rows loaded;
                                                                        long
  double
                                       c discount;
                                                                        long
                                                                                  item rows loaded;
// fix to avoid ODBC float to numeric conversion problem.
                                                                                  warehouse_rows_loaded;
                                                                        long
// double
                                                                                  main time start;
                                       c balance;
                                                                        long
         char
                                                 c_balance[6];
                                                                        long
                                                                                  main_time_end;
                                                                        long
                                                                                                      max items;
                                                                                                      customers_per_district;
  double
                                       c_ytd_payment;
                                                                        long
          short
                                                 c payment cnt;
                                                                                                      orders per district;
                                                                        long
          short
                                                 c_delivery_cnt;
                                                                        long
                                                                                                      first_new_order;
                                                                                                      last_new_order;
         char
                                                                        long
c_data[C_DATA_LEN+1];
          double
                                                 h amount;
                                       h data[H DATA LEN+1];
                                                                        TPCCLDR ARGS *aptr, args;
} CUSTOMER_STRUCT;
```

```
//
// Function name: main
int main(int argc, char **argv)
          DWORD
                        dwThreadID[MAX MAIN THREADS];
          HANDLE
                        hThread[MAX MAIN THREADS];
                     *fLoader;
          FILE
                    buffer[255];
          char
          int
                                                 i;
          for (i=0; i<MAX_MAIN_THREADS; i++)
                   hThread[i] = \overline{NULL};
  printf("\n****************
          printf("\n*
          printf("\n* Microsoft SQL Server
          printf("\n*
          printf("\n* TPC-C BENCHMARK KIT: Database loader
          printf("\n* Version %s
                                                  *", TPCKIT_VER);
          printf("\n*
printf("\n***********\n\n");
  // process command line arguments
  aptr = &args;
  GetArgsLoader(argc, argv, aptr);
          // verify database and tables exist before attempting to load
          CheckSQL();
          CheckDataBase();
          printf("Build interface is ODBC.\n");
          if (aptr->build index == 0)
                   printf("Data load only - no index creation.\n");
          else
                   printf("Data load and index creation.\n");
          if (aptr->index order == 0)
                   printf("Clustered indexes will be created after bulk
load.\n");
          else
                   printf("Clustered indexes will be created before bulk
load.\n");
          // set database scale values
          if (aptr->scale_down == 1)
                   printf("*** Scaled Down Database ***\n");
                    max items = MAXITEMS SCALE DOWN;
                   customers_per_district =
CUSTOMERS SCALE DOWN;
                    orders_per_district = ORDERS_SCALE_DOWN;
                    first_new_order = 0;
                    last_new_order = 30;
          else
          {
                    max items = MAXITEMS;
```

```
customers_per_district =
CUSTOMERS PER DISTRICT;
                     orders_per_district = ORDERS_PER_DISTRICT;
                     first_new_order = 2100;
                     last_new_order = 3000;
          }
          // open connections to SQL Server
          OpenConnections();
          // open file for loader results
  fLoader = fopen(aptr->loader_res_file, "w");
  if(fLoader == NULL)
          {
                     printf("Error, loader result file open failed.");
                     exit(-1);
  // start loading data
  sprintf(buffer,"TPC-C load started for %ld
warehouses.\n",aptr->num_warehouses);
          printf("%s",buffer);
          fprintf(fLoader,"%s",buffer);
          main_time_start = (TimeNow() / MILLI);
          // start parallel load threads
          if (aptr->tables_all || aptr->table_item)
                     fprintf(fLoader, "\nStarting loader threads for: item\n");
                     hThread[0] = CreateThread(NULL,
0,
(LPTHREAD START ROUTINE) LoadItem,
NULL,
0.
&dwThreadID[0]);
                     if(hThread[0] == NULL)
                                printf("Error, failed in creating creating thread
= 0.\langle n'' \rangle;
                                exit(-1);
          if (aptr->tables all || aptr->table warehouse)
                     fprintf(fLoader, "Starting loader threads for:
warehouse\n");
                     hThread[1] = CreateThread(NULL,
0,
(LPTHREAD START ROUTINE) LoadWarehouse,
```

```
NULL,
                                                                                );
0,
&dwThreadID[1]);
                      if (hThread[1] == NULL)
                                printf("Error, failed in creating creating thread
= 1.\langle n'' \rangle;
                                exit(-1);
          }
          if (aptr->tables_all || aptr->table_customer)
                      fprintf(fLoader, "Starting loader threads for:
customer\n");
                     hThread[2] = CreateThread(NULL,
0,
(LPTHREAD_START_ROUTINE) LoadCustomer,
NULL,
0,
&dwThreadID[2]);
                     if (hThread[2] == NULL)
                                printf("Error, failed in creating creating main
thread = 2.\n");
                                exit(-1);
          if (aptr->tables all || aptr->table orders)
                      fprintf(fLoader, "Starting loader threads for: orders\n");
                     hThread[3] = CreateThread(NULL,
0,
(LPTHREAD_START_ROUTINE) LoadOrders,
NULL,
&dwThreadID[3]);
                      if (hThread[3] == NULL)
                                printf("Error, failed in creating creating main
thread = 3.\n");
                                exit(-1);
           // Wait for threads to finish...
          for (i=0; i<MAX MAIN THREADS; i++)
                     if (hThread[i] != NULL)
```

```
WaitForSingleObject( hThread[i], INFINITE
                               CloseHandle(hThread[i]);
                               hThread[i] = NULL;
          }
          main_time_end = (TimeNow() / MILLI);
  sprintf(buffer,"\nTPC-C load completed successfully in %ld minutes.\n",
                               (main_time_end - main_time_start)/60);
          printf("%s",buffer);
          fprintf(fLoader, "%s", buffer);
          fclose(fLoader);
          SQLFreeEnv(henv);
  exit(0);
          return 0;
// Function name: LoadItem
void LoadItem()
  long
                     i_id;
                               i im id;
          long
                     i_name[I_NAME_LEN+1];
  char
  double
                     i_price;
  char
                     i data[I DATA LEN+1];
          char
                               name[20];
          long
                               time start;
          RETCODE
          DBINT
                               rcint;
                               bcphint[128];
          char
          // Seed with unique number
          seed(1);
          printf("Loading item table...\n");
          // if build index before load
          if ((aptr->build index == 1) && (aptr->index order == 1))
                     BuildIndex("idxitmcl");
          InitString(i name, I NAME LEN+1);
          InitString(i_data, I_DATA_LEN+1);
          sprintf(name, "%s..%s", aptr->database, "item");
          rc = bcp_init(i_hdbc1, name, NULL, "logs\\item.err", DB_IN);
          if (rc != SUCCEED)
                     HandleErrorDBC(i_hdbc1);
          if ((aptr->build index == 1) && (aptr->index order == 1))
```

```
BuildIndex("idxitmcl");
                    sprintf(bcphint, "tablock, order (i id),
ROWS PER BATCH = 100000");
                    rc = bcp control(i hdbc1, BCPHINTS, (void*) bcphint);
                    if (rc != SUCCEED)
                              HandleErrorDBC(i_hdbc1);
          rc = bcp_bind(i_hdbc1, (BYTE *) &i_id, 0, SQL_VARLEN_DATA,
NULL, 0, SQLINT4, 1);
                                                                          // Function : LoadWarehouse
          if (rc != SUCCEED)
                    HandleErrorDBC(i_hdbc1);
                                                                          // Loads WAREHOUSE table and loads Stock and District as Warehouses are
                                                                          created
          rc = bcp bind(i hdbc1, (BYTE *) &i im id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 2);
          if (rc != SUCCEED)
                    HandleErrorDBC(i hdbc1);
                                                                          void LoadWarehouse()
          rc = bcp_bind(i_hdbc1, (BYTE *) i_name, 0, I_NAME_LEN,
NULL, 0, 0, 3);
                                                                             short
                                                                                    w id;
          if (rc != SUCCEED)
                                                                                    w name[W NAME LEN+1];
                                                                             char
                    HandleErrorDBC(i_hdbc1);
                                                                                    w_street_1[ADDRESS_LEN+1];
                                                                             char
                                                                                    w street 2[ADDRESS LEN+1];
                                                                             char
          rc = bcp_bind(i_hdbc1, (BYTE *) &i_price, 0,
                                                                             char
                                                                                    w city[ADDRESS LEN+1];
SQL VARLEN DATA, NULL, 0, SQLFLT8, 4);
                                                                                    w state[STATE LEN+1];
                                                                            char
          if (rc != SUCCEED)
                                                                             char
                                                                                    w_zip[ZIP_LEN+1];
                    HandleErrorDBC(i hdbc1);
                                                                             double w_tax;
                                                                             double
                                                                                    w ytd;
          rc = bcp_bind(i_hdbc1, (BYTE *) i_data, 0, I_DATA_LEN, NULL,
                                                                                              name[20];
                                                                                    char
0, 0, 5);
                                                                                    long
                                                                                              time start;
          if (rc != SUCCEED)
                                                                                    RETCODE
                    HandleErrorDBC(i hdbc1);
                                                                                    DBINT
                                                                                              reint;
                                                                                    char
                                                                                               bcphint[128];
          time_start = (TimeNow() / MILLI);
          item_rows_loaded = 0;
                                                                                    // Seed with unique number
                                                                                    seed(2);
          for (i_id = 1; i_id \le max_items; i_id++)
                                                                                    printf("Loading warehouse table...\n");
                    i_im_id = RandomNumber(1L, 10000L);
                                                                                    // if build index before load...
                    MakeAlphaString(14, 24, I NAME LEN, i name);
                                                                                    if ((aptr->build index == 1) && (aptr->index order == 1))
                                                                                              BuildIndex("idxwarel");
                    i price = ((float) RandomNumber(100L, 10000L))/100.0;
                                                                                    InitString(w_name, W_NAME_LEN+1);
                    MakeOriginalAlphaString(26, 50, I_DATA_LEN, i_data,
                                                                                    InitAddress(w street 1, w street 2, w city, w state, w zip);
10);
                                                                                    sprintf(name, "%s..%s", aptr->database, "warehouse");
                    rc = bcp_sendrow(i_hdbc1);
                    if (rc != SUCCEED)
                                                                                    rc = bcp_init(w_hdbc1, name, NULL, "logs\\whouse.err", DB_IN);
                              HandleErrorDBC(i hdbc1);
                                                                                    if (rc != SUCCEED)
                                                                                              HandleErrorDBC(w_hdbc1);
                    item rows loaded++;
                    CheckForCommit(i_hdbc1, i_hstmt1, item_rows_loaded,
"item", &time_start);
                                                                                    if ((aptr->build index == 1) && (aptr->index order == 1))
          }
                                                                                              sprintf(bcphint, "tablock, order (w_id),
          rcint = bcp_done(i_hdbc1);
                                                                          ROWS PER BATCH = %d", aptr->num warehouses);
          if (rcint < 0)
                                                                                              rc = bcp_control(w_hdbc1, BCPHINTS, (void*)
                    HandleErrorDBC(i_hdbc1);
                                                                          bcphint);
                                                                                              if (rc != SUCCEED)
          printf("Finished loading item table.\n");
                                                                                                         HandleErrorDBC(w hdbc1);
          SQLFreeStmt(i hstmt1, SQL DROP);
          SQLDisconnect(i_hdbc1);
                                                                                    rc = bcp_bind(w_hdbc1, (BYTE *) &w_id, 0,
          SQLFreeConnect(i_hdbc1);
                                                                          SQL_VARLEN_DATA, NULL, 0, SQLINT2, 1);
                                                                                    if (rc != SUCCEED)
          // if build index after load
                                                                                              HandleErrorDBC(w_hdbc1);
          if ((aptr->build_index == 1) && (aptr->index_order == 0))
```

```
rc = bcp_bind(w_hdbc1, (BYTE *) w_name, 0, W_NAME_LEN,
NULL, 0, 0, 2);
                                                                                     printf("Finished loading warehouse table.\n");
          if (rc != SUCCEED)
                    HandleErrorDBC(w_hdbc1);
                                                                                     // if build index after load...
                                                                                     if ((aptr->build_index == 1) && (aptr->index_order == 0))
                                                                                               BuildIndex("idxwarel");
          rc = bcp bind(w hdbc1, (BYTE *) w street 1, 0, ADDRESS LEN,
NULL, 0, 0, 3);
          if (rc != SUCCEED)
                                                                                     stock rows loaded = 0;
                    HandleErrorDBC(w hdbc1);
                                                                                     district rows loaded = 0;
          rc = bcp_bind(w_hdbc1, (BYTE *) w_street_2, 0, ADDRESS_LEN,
                                                                                     District();
NULL, 0, 0, 4);
                                                                                     Stock();
          if (rc != SUCCEED)
                    HandleErrorDBC(w_hdbc1);
          rc = bcp_bind(w_hdbc1, (BYTE *) w_city, 0, ADDRESS_LEN,
NULL, 0, 0, 5);
          if (rc != SUCCEED)
                    HandleErrorDBC(w_hdbc1);
                                                                           // Function : District
          rc = bcp_bind(w_hdbc1, (BYTE *) w_state, 0, STATE_LEN,
NULL, 0, 0, 6);
          if (rc != SUCCEED)
                    HandleErrorDBC(w hdbc1);
                                                                           void District()
          rc = bcp_bind(w_hdbc1, (BYTE *) w_zip, 0, ZIP_LEN, NULL, 0, 0,
                                                                             short
7);
                                                                                     d id;
          if (rc != SUCCEED)
                                                                             short
                                                                                     d_w_id;
                    HandleErrorDBC(w hdbc1);
                                                                             char
                                                                                     d_name[D_NAME_LEN+1];
                                                                                     d_street_1[ADDRESS_LEN+1];
                                                                             char
          rc = bcp bind(w hdbc1, (BYTE *) &w tax, 0,
                                                                             char
                                                                                     d street 2[ADDRESS LEN+1];
SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 8);
                                                                                     d_city[ADDRESS_LEN+1];
                                                                             char
          if (rc != SUCCEED)
                                                                             char
                                                                                     d state[STATE LEN+1];
                    HandleErrorDBC(w hdbc1);
                                                                                     d zip[ZIP LEN+1];
                                                                             char
                                                                             double d_tax;
          rc = bcp bind(w hdbc1, (BYTE *) &w ytd, 0,
                                                                             double d ytd;
SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 9);
                                                                                     char
                                                                                               name[20];
          if (rc != SUCCEED)
                                                                                     d_next_o_id;
                                                                             long
                    HandleErrorDBC(w_hdbc1);
                                                                                     long
                                                                                               time_start;
                                                                                     int
                                                                                                         w_id;
          time start = (TimeNow() / MILLI);
                                                                                     RETCODE
                                                                                                         rc;
                                                                                     DBINT
                                                                                               reint:
          warehouse rows loaded = 0;
                                                                                               bcphint[128];
                                                                                     char
          for (w id = (short)aptr->starting warehouse; w id <=
                                                                                     // Seed with unique number
aptr->num_warehouses; w_id++)
                                                                                     seed(4);
                    MakeAlphaString(6,10, W NAME LEN, w name);
                                                                                     printf("Loading district table...\n");
                    MakeAddress(w_street_1, w_street_2, w_city, w_state,
                                                                                     // build index before load
                                                                                     if ((aptr->build_index == 1) && (aptr->index_order == 1))
w_zip);
                                                                                               BuildIndex("idxdiscl");
                    w_tax = ((float) RandomNumber(0L,2000L))/10000.00;
                                                                                     InitString(d name, D NAME LEN+1);
                    w ytd = 300000.00;
                                                                                     InitAddress(d_street_1, d_street_2, d_city, d_state, d_zip);
                                                                                     sprintf(name, "%s..%s", aptr->database, "district");
                    rc = bcp sendrow(w hdbc1);
                    if (rc != SUCCEED)
                                                                                     rc = bcp_init(w_hdbc1, name, NULL, "logs\\district.err", DB_IN);
                              HandleErrorDBC(w_hdbc1);
                                                                                     if (rc != SUCCEED)
                    warehouse rows loaded++;
                                                                                               HandleErrorDBC(w hdbc1);
                    CheckForCommit(w hdbc1, i hstmt1,
warehouse_rows_loaded, "warehouse", &time_start);
                                                                                     if ((aptr->build_index == 1) && (aptr->index_order == 1))
                                                                                               sprintf(bcphint, "tablock, order (d_w_id, d_id),
          rcint = bcp done(w hdbc1);
                                                                           ROWS PER BATCH = \%u'', (aptr->num warehouses *10));
                                                                                               rc = bcp_control(w_hdbc1, BCPHINTS, (void*)
          if (rcint < 0)
                    HandleErrorDBC(w hdbc1);
                                                                           bcphint);
```

```
if (rc != SUCCEED)
                                                                                              d_w_id = w_id;
                              HandleErrorDBC(w hdbc1);
                                                                                              for (d id = 1; d id <= DISTRICT PER WAREHOUSE;
          }
                                                                          d_id++)
          rc = bcp bind(w hdbc1, (BYTE *) &d id, 0,
SQL VARLEN DATA, NULL, 0, SQLINT2, 1);
                                                                                                         MakeAlphaString(6,10,D NAME LEN,
          if (rc != SUCCEED)
                                                                          d name);
                    HandleErrorDBC(w hdbc1);
                                                                                                         MakeAddress(d_street_1, d_street_2, d_city,
          rc = bcp bind(w hdbc1, (BYTE *) &d w id, 0,
                                                                          d_state, d_zip);
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 2);
          if (rc != SUCCEED)
                                                                                                         d tax = ((float))
                    HandleErrorDBC(w hdbc1);
                                                                          RandomNumber(0L,2000L))/10000.00;
          rc = bcp_bind(w_hdbc1, (BYTE *) d_name, 0, D_NAME_LEN,
                                                                                                         rc = bcp_sendrow(w_hdbc1);
NULL, 0, 0, 3);
                                                                                                         if (rc != SUCCEED)
          if (rc != SUCCEED)
                                                                                                                   HandleErrorDBC(w hdbc1);
                    HandleErrorDBC(w_hdbc1);
                                                                                                         district rows loaded++;
          rc = bcp bind(w hdbc1, (BYTE *) d street 1, 0, ADDRESS LEN,
                                                                                                         CheckForCommit(w hdbc1, w hstmt1,
NULL, 0, 0, 4);
                                                                          district_rows_loaded, "district", &time_start);
          if (rc != SUCCEED)
                    HandleErrorDBC(w_hdbc1);
          rc = bcp_bind(w_hdbc1, (BYTE *) d_street_2, 0, ADDRESS_LEN,
                                                                                    rcint = bcp_done(w_hdbc1);
NULL, 0, 0, 5);
                                                                                    if (reint < 0)
          if (rc != SUCCEED)
                                                                                              HandleErrorDBC(w hdbc1);
                    HandleErrorDBC(w_hdbc1);
                                                                                    printf("Finished loading district table.\n");
          rc = bcp_bind(w_hdbc1, (BYTE *) d_city, 0, ADDRESS_LEN,
NULL, 0, 0, 6);
                                                                                    // if build index after load...
                                                                                    if ((aptr->build_index == 1) && (aptr->index_order == 0))
          if (rc != SUCCEED)
                                                                                              BuildIndex("idxdiscl");
                    HandleErrorDBC(w_hdbc1);
          rc = bcp_bind(w_hdbc1, (BYTE *) d_state, 0, STATE_LEN, NULL,
                                                                            return:
0, 0, 7);
          if (rc != SUCCEED)
                    HandleErrorDBC(w hdbc1);
          rc = bcp_bind(w_hdbc1, (BYTE *) d_zip, 0, ZIP_LEN, NULL, 0, 0,
8);
          if (rc != SUCCEED)
                                                                          // Function : Stock
                    HandleErrorDBC(w hdbc1);
          rc = bcp bind(w hdbc1, (BYTE *) &d tax, 0,
SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 9);
          if (rc != SUCCEED)
                                                                          void Stock()
                    HandleErrorDBC(w hdbc1);
                                                                                    s_i_id;
                                                                            long
          rc = bcp bind(w hdbc1, (BYTE *) &d ytd, 0,
                                                                                    s w id;
                                                                             short
SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 10);
                                                                             short
                                                                                    s_quantity;
          if (rc != SUCCEED)
                                                                                    s dist 01[S DIST LEN+1];
                                                                            char
                    HandleErrorDBC(w_hdbc1);
                                                                                    s_dist_02[S_DIST_LEN+1];
                                                                             char
                                                                                    s_dist_03[S_DIST_LEN+1];
s_dist_04[S_DIST_LEN+1];
                                                                             char
          rc = bcp_bind(w_hdbc1, (BYTE *) &d_next_o_id, 0,
                                                                             char
                                                                                    s_dist_05[S_DIST_LEN+1];
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 11);
                                                                             char
          if (rc != SUCCEED)
                                                                                    s dist 06[S DIST LEN+1];
                                                                             char
                                                                                    s_dist_07[S_DIST_LEN+1];
                    HandleErrorDBC(w_hdbc1);
                                                                             char
                                                                                    s dist 08[S_DIST_LEN+1];
                                                                             char
                                                                                    s_dist_09[S_DIST_LEN+1];
          d_ytd = 30000.0;
                                                                             char
                                                                             char
                                                                                    s_dist_10[S_DIST_LEN+1];
          d_next_o_id = orders_per_district+1;
                                                                             long
                                                                                    s_ytd;
                                                                             short
                                                                                    s_order_cnt;
          time_start = (TimeNow() / MILLI);
                                                                             short
                                                                                    s_remote_cnt;
                                                                                    s_data[S_DATA_LEN+1];
                                                                            char
          for (w id = aptr->starting warehouse; w id <=
                                                                             short
                                                                                    len;
aptr->num_warehouses; w_id++)
                                                                                              name[20];
                                                                                    char
                                                                                    long
                                                                                              time_start;
          {
```

```
RETCODE
                                                                                   rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_06, 0, S_DIST_LEN,
                              rc:
                                                                          NULL, 0, 0, 9);
          DBINT
                   rcint;
                   bcphint[128];
                                                                                   if (rc != SUCCEED)
          char
                                                                                              HandleErrorDBC(w_hdbc1);
          // Seed with unique number
          seed(3);
                                                                                   rc = bcp bind(w hdbc1, (BYTE *) s dist 07, 0, S DIST LEN,
                                                                          NULL, 0, 0, 10);
          // if build index before load...
                                                                                    if (rc != SUCCEED)
          if ((aptr->build index == 1) && (aptr->index order == 1))
                                                                                             HandleErrorDBC(w hdbc1);
                    BuildIndex("idxstkcl");
                                                                                   rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_08, 0, S_DIST_LEN,
          sprintf(name, "%s..%s", aptr->database, "stock");
                                                                          NULL, 0, 0, 11);
                                                                                    if (rc != SUCCEED)
                                                                                             HandleErrorDBC(w_hdbc1);
          rc = bcp_init(w_hdbc1, name, NULL, "logs\\stock.err", DB_IN);
          if (rc != SUCCEED)
                    HandleErrorDBC(w hdbc1);
                                                                                    rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_09, 0, S_DIST_LEN,
                                                                          NULL, 0, 0, 12);
                                                                                    if (rc != SUCCEED)
          if ((aptr->build_index == 1) && (aptr->index_order == 1))
                                                                                             HandleErrorDBC(w_hdbc1);
                    sprintf(bcphint, "tablock, order (s i id, s w id),
ROWS_PER_BATCH = %u", (aptr->num_warehouses * 100000));
                                                                                    rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_10, 0, S_DIST_LEN,
                   rc = bcp control(w hdbc1, BCPHINTS, (void*)
                                                                          NULL, 0, 0, 13);
bcphint);
                                                                                    if (rc != SUCCEED)
                   if (rc != SUCCEED)
                                                                                             HandleErrorDBC(w_hdbc1);
                              HandleErrorDBC(w_hdbc1);
                                                                                    rc = bcp_bind(w_hdbc1, (BYTE *) &s_ytd, 0,
          }
                                                                          SQL VARLEN DATA, NULL, 0, SQLINT4, 14);
          rc = bcp_bind(w_hdbc1, (BYTE *) &s_i_id, 0,
                                                                                    if (rc != SUCCEED)
SQL VARLEN DATA, NULL, 0, SQLINT4, 1);
                                                                                             HandleErrorDBC(w hdbc1);
          if (rc != SUCCEED)
                    HandleErrorDBC(w hdbc1);
                                                                                    rc = bcp bind(w hdbc1, (BYTE *) &s order cnt, 0,
                                                                          SQL_VARLEN_DATA, NULL, 0, SQLINT2, 15);
          bcp_bind(w_hdbc1, (BYTE *) &s_w_id, 0, SQL_VARLEN_DATA,
                                                                                   if (rc != SUCCEED)
                                                                                             HandleErrorDBC(w hdbc1);
NULL, 0, SQLINT2, 2);
          if (rc != SUCCEED)
                    HandleErrorDBC(w hdbc1);
                                                                                    rc = bcp bind(w hdbc1, (BYTE *) &s remote cnt, 0,
                                                                          SQL_VARLEN_DATA, NULL, 0, SQLINT2, 16);
          rc = bcp bind(w hdbc1, (BYTE *) &s quantity, 0,
                                                                                    if (rc != SUCCEED)
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 3);
                                                                                             HandleErrorDBC(w_hdbc1);
          if (rc != SUCCEED)
                    HandleErrorDBC(w hdbc1);
                                                                                   rc = bcp bind(w hdbc1, (BYTE *) s data, 0, S DATA LEN,
                                                                          NULL, 0, 0, 17);
          rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_01, 0, S_DIST_LEN,
                                                                                    if (rc != SUCCEED)
NULL, 0, 0, 4);
                                                                                             HandleErrorDBC(w_hdbc1);
          if (rc != SUCCEED)
                    HandleErrorDBC(w_hdbc1);
                                                                                    s_ytd = s_order_cnt = s_remote_cnt = 0;
          rc = bcp bind(w hdbc1, (BYTE *) s dist 02, 0, S DIST LEN,
                                                                                    time start = (TimeNow() / MILLI);
NULL, 0, 0, 5);
          if (rc != SUCCEED)
                                                                                    printf("...Loading stock table\n");
                    HandleErrorDBC(w_hdbc1);
                                                                                    for (s i id=1; s i id \leq max items; s i id++)
          rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_03, 0, S_DIST_LEN,
NULL, 0, 0, 6);
          if (rc != SUCCEED)
                                                                                             for (s_w_id = (short)aptr->starting_warehouse; s_w_id
                   HandleErrorDBC(w_hdbc1);
                                                                          <= aptr->num_warehouses; s_w_id++)
          rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_04, 0, S_DIST_LEN,
                                                                                                        s_quantity =
NULL, 0, 0, 7);
          if (rc != SUCCEED)
                                                                          (short)RandomNumber(10L,100L);
                   HandleErrorDBC(w hdbc1);
                                                                                                        len = MakeAlphaString(24,24,S DIST LEN,
                                                                          s_dist_01);
          rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_05, 0, S_DIST_LEN,
                                                                                                        len = MakeAlphaString(24,24,S_DIST_LEN,
NULL, 0, 0, 8);
                                                                          s_dist_02);
          if (rc != SUCCEED)
                                                                                                        len = MakeAlphaString(24,24,S_DIST_LEN,
                    HandleErrorDBC(w hdbc1);
                                                                          s dist 03);
                                                                                                        len = MakeAlphaString(24,24,S_DIST_LEN,
                                                                          s dist 04);
```

```
len = MakeAlphaString(24,24,S_DIST_LEN,
s_dist_05);
                             len = MakeAlphaString(24,24,S DIST LEN,
s_dist_06);
                             len = MakeAlphaString(24,24,S_DIST_LEN,
s dist 07);
                             len = MakeAlphaString(24,24,S DIST LEN,
s_dist_08);
                             len = MakeAlphaString(24,24,S DIST LEN,
s_dist_09);
                             len = MakeAlphaString(24,24,S_DIST_LEN,
s_dist_10);
                             len = MakeOriginalAlphaString(26,50,
S_DATA_LEN, s_data, 10);
                             rc = bcp_sendrow(w_hdbc1);
                             if (rc != SUCCEED)
                                       HandleErrorDBC(w_hdbc1);
                             stock_rows_loaded++;
                             CheckForCommit(w hdbc1, w hstmt1,
stock_rows_loaded, "stock", &time_start);
          rcint = bcp_done(w_hdbc1);
          if (rcint < 0)
                   HandleErrorDBC(w_hdbc1);
          printf("Finished loading stock table.\n");
          SQLFreeStmt(w hstmt1, SQL DROP);
         SQLDisconnect(w_hdbc1);
          SQLFreeConnect(w hdbc1);
          // if build index after load...
          if ((aptr->build_index == 1) && (aptr->index_order == 0))
                   BuildIndex("idxstkcl");
          return;
}
//
// Function : LoadCustomer
void LoadCustomer()
         LOADER TIME STRUCT
                                      customer_time_start;
         LOADER TIME STRUCT
                                      history time start;
          short
                                        w_id;
  short
                    d_id;
          DWORD
dwThreadID[MAX CUSTOMER THREADS];
         HANDLE
hThread[MAX_CUSTOMER_THREADS];
                                        name[20];
          RETCODE
rc;
          DBINT
                                                           rcint;
```

```
char
bcphint[128];
                                                              cmd[256];
          char
          // SQLRETURN
                                                              rc_1;
          // SQLSMALLINT
                                                              recnum,
MsgLen;
          // SQLCHAR
SqlState[6], Msg[SQL_MAX_MESSAGE_LENGTH];
          // SQLINTEGER
                                                              NativeError;
          // Seed with unique number
          seed(5);
          printf("Loading customer and history tables...\n");
          // if build index before load...
          if ((aptr->build index == 1) && (aptr->index order == 1))
                    BuildIndex("idxcuscl");
          // Initialize bulk copy
          sprintf(name, "%s..%s", aptr->database, "customer");
          rc = bcp_init(c_hdbc1, name, NULL, "logs\\customer.err", DB_IN);
          if (rc != SUCCEED)
                    HandleErrorDBC(c_hdbc1);
          if ((aptr->build_index == 1) && (aptr->index_order == 1))
                    sprintf(bcphint, "tablock, order (c_w_id, c_d_id, c_id),
ROWS PER BATCH = %u", (aptr->num warehouses * 30000));
                    rc = bcp_control(c_hdbc1, BCPHINTS, (void*) bcphint);
                    if (rc != SUCCEED)
                               HandleErrorDBC(c_hdbc1);
          }
          sprintf(name, "%s..%s", aptr->database, "history");
          rc = bcp_init(c_hdbc2, name, NULL, "logs\\history.err", DB_IN);
          if (rc != SUCCEED)
                    HandleErrorDBC(c hdbc2);
          sprintf(bcphint, "tablock");
          rc = bcp_control(c_hdbc2, BCPHINTS, (void*) bcphint);
          if (rc != SUCCEED)
                    HandleErrorDBC(c hdbc2);
          customer rows loaded = 0;
          history_rows_loaded = 0;
          CustomerBufInit();
          customer_time_start.time_start = (TimeNow() / MILLI);
          history_time_start.time_start = (TimeNow() / MILLI);
          for (w_id = (short)aptr->starting_warehouse; w_id <=
aptr->num warehouses; w id++)
                    for (d id = 1; d id <= DISTRICT PER WAREHOUSE;
d id++)
                               CustomerBufLoad(d_id, w_id);
                               // Start parallel loading threads here...
```

```
// Start customer table thread
                                                                                                                                                                           }
                                                             printf("...Loading customer table for: d id =
                                                                                                                                                                           // flush the bulk connection
d, w_id = d'n', d_id, w_id;
                                                                                                                                                                           rcint = bcp_done(c_hdbc1);
                                                                                                                                                                           if (rcint < 0)
                                                             hThread[0] = CreateThread(NULL,
                                                                                                                                                                                               HandleErrorDBC(c hdbc1);
                      0,
                                                                                                                                                                           rcint = bcp_done(c_hdbc2);
                                                                                                                                                                           if (rcint < 0)
                      (LPTHREAD START ROUTINE) LoadCustomerTable,
                                                                                                                                                                                               HandleErrorDBC(c hdbc2);
                                                                                                                                                                           printf("Finished loading customer table.\n");
                       &customer_time_start,
                      0,
                                                                                                                                                                           // if build index after load...
                                                                                                                                                                           if ((aptr->build index == 1) && (aptr->index order == 0))
                                                                                                                                                                                               BuildIndex("idxcuscl");
                       &dwThreadID[0]);
                                                             if(hThread[0] == NULL)
                                                                                                                                                                           // build non-clustered index
                                                                                                                                                                           if (aptr->build index == 1)
                                                                                                                                                                                               BuildIndex("idxcusnc");
                                                                                  printf("Error, failed in creating
creating thread = 0.\n");
                                                                                  exit(-1);
                                                                                                                                                                           // Output the NURAND used for the loader into C FIRST for C ID
                                                                                                                                                       = 1,
                                                                                                                                                                           // C W ID = 1, and C D ID = 1
                                                             // Start History table thread
                                                                                                                                                                           sprintf(cmd, "isql -S%s -U%s -P%s -d%s -e -Q\"update customer set
                                                                                                                                                       c_{inst} = C_{inst} 
                                                             printf("...Loading history table for: d id = %d,
                                                                                                                                                       logs\\nurand load.log",
w_id = %d\n'', d_id, w_id);
                                                                                                                                                                                                                                         aptr->server,
                                                                                                                                                                                                                                         aptr->user,
                                                             hThread[1] = CreateThread(NULL,
                                                                                                                                                                                                                                         aptr->password,
                                                                                                                                                                                                                                         aptr->database,
                      0,
                                                                                                                                                                                                                                         LOADER_NURAND_C);
                      (LPTHREAD START ROUTINE) LoadHistoryTable,
                                                                                                                                                                           system(cmd);
                                                                                                                                                                           SQLFreeStmt(c hstmt1, SQL DROP);
                      &history_time_start,
                                                                                                                                                                           SQLDisconnect(c_hdbc1);
                                                                                                                                                                           SQLFreeConnect(c hdbc1);
                      0,
                      &dwThreadID[1]);
                                                                                                                                                                           SQLFreeStmt(c_hstmt2, SQL_DROP);
                                                                                                                                                                           SQLDisconnect(c hdbc2);
                                                             if (hThread[1] == NULL)
                                                                                                                                                                           SQLFreeConnect(c_hdbc2);
                                                                                  printf("Error, failed in creating
creating thread = 1.\n");
                                                                                                                                                           return;
                                                                                  exit(-1);
                                                             WaitForSingleObject( hThread[0], INFINITE
);
                                                             WaitForSingleObject( hThread[1], INFINITE
);
                                                                                                                                                       // Function : CustomerBufInit
                                                             if(CloseHandle(hThread[0]) == FALSE)
                                                                                  printf("Error, failed in closing
customer thread handle with errno: %d\n", GetLastError());
                                                                                                                                                       void CustomerBufInit()
                                                             if(CloseHandle(hThread[1]) == FALSE)
                                                                                                                                                                           int i;
                                                                                  printf("Error, failed in closing
                                                                                                                                                                           for (i=0;i<customers_per_district;i++)
history thread handle with errno: %d\n", GetLastError());
                                                                                                                                                                                               customer_buf[i].c_id = 0;
                                                                                                                                                                                               customer_buf[i].c_d_id = 0;
                                                                                                                                                                                               customer buf[i].c w id = 0;
                                         }
                                                                                                                                                                                               strcpy(customer buf[i].c first,"");
```

```
strcpy(customer buf[i].c middle,"");
                                                                                        {
                     strcpy(customer_buf[i].c_last,"");
                     strcpy(customer_buf[i].c_street_1,"");
                                                                                                   customer buf[i].c d id = d id;
                     strcpy(customer_buf[i].c_street_2,"
                                                                                                   customer_buf[i].c_w_id = w_id;
                     strcpy(customer_buf[i].c_city,"");
                                                                                                   customer_buf[i].h_amount = 10.0;
                     strcpy(customer_buf[i].c_state,"");
                     strcpy(customer_buf[i].c_zip,"");
                                                                                                   customer buf[i].c ytd payment = 10.0;
                     strcpy(customer_buf[i].c_phone,"");
                     strcpy(customer_buf[i].c_credit,"");
                                                                                                   customer_buf[i].c_payment_cnt = 1;
                                                                                                   customer_buf[i].c_delivery_cnt = 0;
                     customer_buf[i].c_credit_lim = 0;
                     customer_buf[i].c_discount = (float) 0;
                                                                                                   // Generate CUSTOMER and HISTORY data
                     // fix to avoid ODBC float to numeric conversion
                                                                                                   customer_buf[i].c_id = c[i].c_id;
problem.
                     //
                               customer_buf[i].c_balance = 0;
                                                                                                   strcpy(customer_buf[i].c_first, c[i].c_first);
                     strcpy(customer_buf[i].c_balance,"");
                                                                                                   strcpy(customer_buf[i].c_last, c[i].c_last);
                     customer_buf[i].c_ytd_payment = 0;
                                                                                                   customer_buf[i].c_middle[0] = 'O';
                     customer buf[i].c payment cnt = 0;
                                                                                                   customer buf[i].c middle[1] = 'E';
                     customer_buf[i].c_delivery_cnt = 0;
                                                                                                   MakeAddress(customer buf[i].c street 1,
                     strcpy(customer_buf[i].c_data,"");
                                                                                                                           customer_buf[i].c_street_2,
                                                                                                                          customer buf[i].c city,
                     customer_buf[i].h_amount = 0;
                                                                                                                           customer_buf[i].c_state,
                                                                                                                          customer_buf[i].c_zip);
                     strcpy(customer buf[i].h data,"");
                                                                                                   MakeNumberString(16, 16, PHONE_LEN,
          }
                                                                              customer_buf[i].c_phone);
                                                                                                   if (RandomNumber(1L, 100L) > 10)
                                                                                                              customer_buf[i].c_credit[0] = 'G';
                                                                                                   else
                                                                                                              customer buf[i].c credit[0] = 'B';
                                                                                                   customer_buf[i].c_credit[1] = 'C';
                                                                                                   customer_buf[i].c_credit_lim = 50000.0;
                                                                                                   customer buf[i].c discount = ((float)
// Function : CustomerBufLoad
                                                                              RandomNumber(0L, 5000L)) / 10000.0;
// Fills shared buffer for HISTORY and CUSTOMER
                                                                                                   // fix to avoid ODBC float to numeric conversion
                                                                              problem.
                                                                                                   // customer buf[i].c balance = -10.0;
void CustomerBufLoad(int d_id, int w_id)
                                                                                                   strcpy(customer_buf[i].c_balance,"-10.0");
                                                                                                   MakeAlphaString(300, 500, C_DATA_LEN,
  CUSTOMER SORT STRUCT c[CUSTOMERS PER DISTRICT];
                                                                              customer buf[i].c data);
                                                                                                   // Generate HISTORY data
          for (i=0;i<customers_per_district;i++)
                                                                                                   MakeAlphaString(12, 24, H DATA LEN,
                                                                              customer_buf[i].h_data);
                     if (i < 1000)
                               LastName(i, c[i].c_last);
                     else
LastName(NURand(255,0,999,LOADER_NURAND_C), c[i].c_last);
                     MakeAlphaString(8,16,FIRST_NAME_LEN,
c[i].c first);
                                                                              // Function : LoadCustomerTable
                     c[i].c id = i+1;
          }
                                                                              void LoadCustomerTable(LOADER_TIME_STRUCT *customer_time_start)
          printf("...Loading customer buffer for: d_id = %d, w_id = %d\n",
                               d id, w id);
                                                                                        int
                                                                                long
                                                                                         c_id;
          for (i=0;i<customers per district;i++)
                                                                                short
                                                                                         c_d_id;
```

```
short
          c w id;
                                                                                            HandleErrorDBC(c_hdbc1);
  char
          c_first[FIRST_NAME_LEN+1];
          c middle[MIDDLE NAME LEN+1];
                                                                          rc = bcp_bind(c_hdbc1, (BYTE *) c_state, 0, STATE_LEN, NULL, 0, 0, 10);
  char
          c\_last[LAST\_NAME\_LEN+1];
                                                                                  if (rc != SUCCEED)
  char
          c_street_1[ADDRESS_LEN+1];
                                                                                            HandleErrorDBC(c hdbc1);
  char
          c street 2[ADDRESS LEN+1];
  char
          c city[ADDRESS LEN+1];
  char
                                                                          rc = bcp bind(c hdbc1, (BYTE *) c zip, 0, ZIP LEN, NULL, 0, 0, 11);
          c_state[STATE_LEN+1];
                                                                                  if (rc != SUCCEED)
  char
          c_zip[ZIP_LEN+1];
  char
                                                                                            HandleErrorDBC(c hdbc1);
          c_phone[PHONE LEN+1];
  char
  char
          c_credit[CREDIT_LEN+1];
                                                                          rc = bcp_bind(c_hdbc1, (BYTE *) c_phone, 0, PHONE_LEN, NULL, 0, 0,
          c_credit_lim;
  double
                                                                        12):
  double
           c discount;
                                                                                  if (rc != SUCCEED)
                                                                                            HandleErrorDBC(c_hdbc1);
         // fix to avoid ODBC float to numeric conversion problem.
         // double
                             c balance;
                                                                                  rc = bcp_bind(c_hdbc1, (BYTE *) &c_since, 0, C_SINCE_LEN,
         char
                             c balance[6];
                                                                        NULL, 0, SQLCHARACTER, 13);
                                                                                  if (rc != SUCCEED)
                                                                                            HandleErrorDBC(c_hdbc1);
  double
          c_ytd_payment;
          c payment cnt;
  short
          c_delivery_cnt;
  short
                                                                          rc = bcp_bind(c_hdbc1, (BYTE *) c_credit, 0, CREDIT_LEN, NULL, 0, 0,
          c data[C DATA LEN+1];
  char
                                                                        14);
          char
                             c_since[C_SINCE_LEN+1];
                                                                                  if (rc != SUCCEED)
          RETCODE
                                                                                            HandleErrorDBC(c hdbc1);
  rc = bcp_bind(c_hdbc1, (BYTE *) &c_id, 0, SQL_VARLEN_DATA, NULL,
                                                                          rc = bcp_bind(c_hdbc1, (BYTE *) &c_credit_lim, 0,
0, SQLINT4, 1);
                                                                        SQL VARLEN DATA, NULL, 0, SQLFLT8, 15);
          if (rc != SUCCEED)
                                                                                  if (rc != SUCCEED)
                   HandleErrorDBC(c hdbc1);
                                                                                            HandleErrorDBC(c hdbc1);
  rc = bcp bind(c hdbc1, (BYTE *) &c d id, 0, SQL VARLEN DATA,
                                                                          rc = bcp bind(c hdbc1, (BYTE *) &c discount, 0, SQL VARLEN DATA,
                                                                        NULL, 0, SQLFLT8, 16);
NULL, 0, SQLINT2, 2);
         if (rc != SUCCEED)
                                                                                  if (rc != SUCCEED)
                   HandleErrorDBC(c hdbc1);
                                                                                            HandleErrorDBC(c hdbc1);
          rc = bcp bind(c hdbc1, (BYTE *) &c w id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 3);
                                                                                  // fix to avoid ODBC float to numeric conversion problem.
         if (rc != SUCCEED)
                   HandleErrorDBC(c_hdbc1);
                                                                          // rc = bcp_bind(c_hdbc1, (BYTE *) &c_balance, 0,
                                                                        SQL VARLEN DATA, NULL, 0, SQLFLT8, 17);
  rc = bcp bind(c hdbc1, (BYTE *) c first, 0, FIRST NAME LEN, NULL, 0,
                                                                                  // if (rc != SUCCEED)
                                                                                            HandleErrorDBC(c_hdbc1);
0, 4);
          if (rc != SUCCEED)
                                                                          rc = bcp bind(c hdbc1, (BYTE *) c balance, 0, 5, NULL, 0,
                                                                        SQLCHARACTER, 17);
                   HandleErrorDBC(c_hdbc1);
                                                                                  if (rc != SUCCEED)
                                                                                            HandleErrorDBC(c_hdbc1);
  rc = bcp_bind(c_hdbc1, (BYTE *) c_middle, 0,
MIDDLE NAME LEN, NULL, 0, 0, 5);
         if (rc != SUCCEED)
                                                                          rc = bcp_bind(c_hdbc1, (BYTE *) &c_ytd_payment, 0,
                   HandleErrorDBC(c_hdbc1);
                                                                        SQL VARLEN DATA, NULL, 0, SQLFLT8, 18);
  rc = bcp_bind(c_hdbc1, (BYTE *) c_last, 0, LAST_NAME_LEN, NULL, 0,
                                                                                  if (rc != SUCCEED)
                                                                                            HandleErrorDBC(c hdbc1);
0, 6);
         if (rc != SUCCEED)
                   HandleErrorDBC(c hdbc1);
                                                                          rc = bcp bind(c hdbc1, (BYTE *) &c payment cnt, 0,
                                                                        SQL_VARLEN_DATA, NULL, 0, SQLINT2, 19);
  rc = bcp_bind(c_hdbc1, (BYTE *) c_street_1, 0, ADDRESS_LEN, NULL, 0,
                                                                                  if (rc != SUCCEED)
                                                                                            HandleErrorDBC(c hdbc1);
0, 7);
         if (rc != SUCCEED)
                   HandleErrorDBC(c hdbc1);
                                                                          rc = bcp bind(c hdbc1, (BYTE *) &c delivery cnt,0,
                                                                        SQL_VARLEN_DATA, NULL, 0, SQLINT2, 20);
  rc = bcp_bind(c_hdbc1, (BYTE *) c_street_2, 0, ADDRESS_LEN,
                                                                                  if (rc != SUCCEED)
NULL,0,0, 8);
                                                                                            HandleErrorDBC(c hdbc1);
         if (rc != SUCCEED)
                   HandleErrorDBC(c_hdbc1);
                                                                          rc = bcp_bind(c_hdbc1, (BYTE *) c_data, 0, 500, NULL, 0, 0, 21);
                                                                                  if (rc != SUCCEED)
  rc = bcp bind(c hdbc1, (BYTE *) c city, 0, ADDRESS LEN, NULL, 0, 0,
                                                                                            HandleErrorDBC(c hdbc1);
9);
          if (rc != SUCCEED)
                                                                          for (i = 0; i < customers_per_district; i++)
```

```
HandleErrorDBC(c hdbc2);
  {
                    c id = customer buf[i].c id;
                    c d id = customer buf[i].c d id;
                                                                             rc = bcp bind(c hdbc2, (BYTE *) &c d id, 0, SQL VARLEN DATA,
                    c_w_id = customer_buf[i].c_w_id;
                                                                           NULL, 0, SQLINT2, 2);
                                                                                     if (rc != SUCCEED)
                    strcpy(c first, customer buf[i].c first);
                                                                                               HandleErrorDBC(c hdbc2);
                    strcpy(c middle, customer buf[i].c middle);
                                                                             rc = bcp_bind(c_hdbc2, (BYTE *) &c_w_id, 0, SQL_VARLEN_DATA,
                    strcpy(c last, customer buf[i].c last);
                                                                           NULL, 0, SQLINT2, 3);
                    strcpy(c_street_1, customer_buf[i].c_street_1);
                                                                                     if (rc != SUCCEED)
                    strcpy(c_street_2, customer_buf[i].c_street_2);
                                                                                               HandleErrorDBC(c_hdbc2);
                    strcpy(c city, customer buf[i].c city);
                    strcpy(c_state, customer_buf[i].c_state);
                                                                             rc = bcp bind(c hdbc2, (BYTE *) &c d id, 0, SQL VARLEN DATA,
                    strcpy(c_zip, customer_buf[i].c_zip);
                                                                           NULL, 0, SQLINT2, 4);
                    strcpy(c_phone, customer_buf[i].c_phone);
                                                                                     if (rc != SUCCEED)
                    strcpy(c credit, customer buf[i].c credit);
                                                                                               HandleErrorDBC(c hdbc2);
                    FormatDate(&c_since);
                                                                             rc = bcp_bind(c_hdbc2, (BYTE *) &c_w_id, 0, SQL_VARLEN_DATA,
                                                                           NULL, 0, SQLINT2, 5);
                                                                                     if (rc != SUCCEED)
                    c_credit_lim = customer_buf[i].c_credit_lim;
                                                                                               HandleErrorDBC(c_hdbc2);
                    c_discount = customer_buf[i].c_discount;
                                                                                     rc = bcp_bind(c_hdbc2, (BYTE *) &h_date, 0, H_DATE_LEN,
                    // fix to avoid ODBC float to numeric conversion
                                                                           NULL, 0, SQLCHARACTER, 6);
problem.
                                                                                     if (rc != SUCCEED)
                    // c balance = customer buf[i].c balance;
                                                                                               HandleErrorDBC(c_hdbc2);
                    strcpy(c_balance, customer_buf[i].c_balance);
                                                                             rc = bcp_bind(c_hdbc2, (BYTE *) &h_amount, 0, SQL_VARLEN_DATA,
                    c_ytd_payment = customer_buf[i].c_ytd_payment;
                                                                           NULL, 0, SQLFLT8, 7);
                    c payment cnt = customer buf[i].c payment cnt;
                                                                                     if (rc != SUCCEED)
                    c delivery_cnt = customer_buf[i].c_delivery_cnt;
                                                                                               HandleErrorDBC(c hdbc2);
                    strcpy(c_data, customer_buf[i].c_data);
                                                                             rc = bcp_bind(c_hdbc2, (BYTE *) h_data, 0, H_DATA_LEN, NULL, 0, 0,
                    // Send data to server
                                                                                     if (rc != SUCCEED)
                    rc = bcp_sendrow(c_hdbc1);
                                                                                               HandleErrorDBC(c hdbc2);
                    if (rc != SUCCEED)
                              HandleErrorDBC(c_hdbc1);
                                                                             for (i = 0; i < customers_per_district; i++)
                    customer_rows_loaded++;
                                                                                               c_id = customer_buf[i].c_id;
                    CheckForCommit(c hdbc1, c hstmt1,
                                                                                               c d id = customer buf[i].c d id;
customer_rows_loaded, "customer", &customer_time_start->time_start);
                                                                                               c_w_id = customer_buf[i].c_w_id;
                                                                                               h amount = customer buf[i].h amount;
                                                                                               strcpy(h_data, customer_buf[i].h_data);
                                                                                               FormatDate(&h_date);
                                                                                               // send to server
                                                                                               rc = bcp_sendrow(c_hdbc2);
                                                                                                if (rc != SUCCEED)
// Function : LoadHistoryTable
                                                                                                          HandleErrorDBC(c_hdbc2);
//
                                                                                               history_rows_loaded++;
                                                                                               CheckForCommit(c hdbc2, c hstmt2,
                                                                           history_rows_loaded, "history", &history_time_start->time_start);
void LoadHistoryTable(LOADER_TIME_STRUCT *history_time_start)
          int
                 i;
  long
          c_id;
  short
          c d id;
  short
          c_w_id;
          double
                    h amount;
  char
          h_data[H_DATA_LEN+1];
                              h_date[H_DATE_LEN+1];
          char
                                                                           // Function : LoadOrders
          RETCODE
  rc = bcp bind(c hdbc2, (BYTE *) &c id, 0, SQL VARLEN DATA, NULL,
0, SQLINT4, 1);
          if (rc != SUCCEED)
```

```
void LoadOrders()
          LOADER TIME STRUCT
                                       orders time start;
          LOADER_TIME_STRUCT
                                       new_order_time_start;
          LOADER_TIME_STRUCT
                                       order_line_time_start;
                                         w id;
  short
                    d_id;
          DWORD
dwThreadID[MAX\_ORDER\_THREADS];
          HANDLE
hThread[MAX_ORDER_THREADS];
                                         name[20];
          char
          RETCODE
rc;
          char
bcphint[128];
          // seed with unique number
          seed(6);
          printf("Loading orders...\n");
          // if build index before load...
          if ((aptr->build index == 1) && (aptr->index order == 1))
                              BuildIndex("idxordcl");
                              BuildIndex("idxnodcl");
                              BuildIndex("idxodlcl");
          }
          // initialize bulk copy
          sprintf(name, "%s..%s", aptr->database, "orders");
          rc = bcp init(o hdbc1, name, NULL, "logs\\orders.err", DB IN);
          if (rc != SUCCEED)
                    HandleErrorDBC(o hdbc1);
          if ((aptr->build index == 1) && (aptr->index order == 1))
                    sprintf(bcphint, "tablock, order (o_w_id, o_d_id, o_id),
ROWS PER BATCH = %u", (aptr->num warehouses * 30000));
                    rc = bcp_control(o_hdbc1, BCPHINTS, (void*) bcphint);
                    if (rc != SUCCEED)
                              HandleErrorDBC(o hdbc1);
          sprintf(name, "%s..%s", aptr->database, "new order");
          rc = bcp_init(o_hdbc2, name, NULL, "logs\\neword.err", DB_IN);
          if (rc != SUCCEED)
                    HandleErrorDBC(o_hdbc2);
          if ((aptr->build index == 1) && (aptr->index order == 1))
                    sprintf(bcphint, "tablock, order (no w id, no d id,
no o id), ROWS_PER_BATCH = %u", (aptr->num_warehouses * 9000));
                    rc = bcp control(o hdbc2, BCPHINTS, (void*) bcphint);
                    if (rc != SUCCEED)
                              HandleErrorDBC(o_hdbc2);
          }
          sprintf(name, "%s..%s", aptr->database, "order_line");
          rc = bcp_init(o_hdbc3, name, NULL, "logs\\ordline.err", DB_IN);
```

```
if (rc != SUCCEED)
                     HandleErrorDBC(o hdbc3);
  if ((aptr->build_index == 1) && (aptr->index_order == 1))
                     sprintf(bcphint, "tablock, order (ol w id, ol d id,
ol o id, ol number), ROWS PER BATCH = %u", (aptr->num warehouses *
300000));
                     rc = bcp_control(o_hdbc3, BCPHINTS, (void*) bcphint);
                     if (rc != SUCCEED)
                               HandleErrorDBC(o hdbc3);
          orders rows loaded
          new_order_rows_loaded = 0;
          order line rows loaded = 0;
          OrdersBufInit();
          orders_time_start.time_start = (TimeNow() / MILLI);
          new order time start.time start = (TimeNow() / MILLI);
          order_line_time_start.time_start = (TimeNow() / MILLI);
          for (w_id = (short)aptr->starting_warehouse; w_id <=
aptr->num_warehouses; w id++)
          {
                     for (d_id = 1; d_id <= DISTRICT_PER_WAREHOUSE;
d id++)
                               OrdersBufLoad(d_id, w_id);
                               // start parallel loading threads here...
                               // start Orders table thread
                               printf("...Loading Order Table for: d id = %d,
w_id = %d\n'', d_id, w_id);
                               hThread[0] = CreateThread(NULL,
           0,
           (LPTHREAD START ROUTINE) LoadOrdersTable,
            &orders_time_start,
           0,
           &dwThreadID[0]);
                               if(hThread[0] == NULL)
                                          printf("Error, failed in creating
creating thread = 0.\n");
                                          exit(-1);
                               // start NewOrder table thread
                               printf("...Loading New-Order Table for: d_id
= \%d, w id = \%d\n", d_id, w_id);
                               hThread[1] = CreateThread(NULL,
           0,
```

```
(LPTHREAD START ROUTINE) LoadNewOrderTable,
           &new_order_time_start,
           &dwThreadID[1]);
                               if (hThread[1] == NULL)
                                          printf("Error, failed in creating
creating thread = 1.\n");
                                          exit(-1);
                               // start Order-Line table thread
                               printf("...Loading Order-Line Table for: d_id
= \%d, w id = \%d n'', d id, w id);
                               hThread[2] = CreateThread(NULL,
           (LPTHREAD_START_ROUTINE) LoadOrderLineTable,
           &order_line_time_start,
           0,
           &dwThreadID[2]);
                               if(hThread[2] == NULL)
                                          printf("Error, failed in creating
creating thread = 2.\n");
                                          exit(-1);
                               WaitForSingleObject( hThread[0], INFINITE
);
                               WaitForSingleObject( hThread[1], INFINITE
);
                               WaitForSingleObject( hThread[2], INFINITE
                               if(CloseHandle(hThread[0]) == FALSE)
                                          printf("Error, failed in closing
Orders thread handle with errno: %d\n", GetLastError());
                               if(CloseHandle(hThread[1]) == FALSE)
                                          printf("Error, failed in closing
NewOrder thread handle with errno: %d\n", GetLastError());
                               if(CloseHandle(hThread[2]) == FALSE)
                                          printf("Error, failed in closing
OrderLine thread handle with errno: %d\n", GetLastError());
                     }
```

```
printf("Finished loading orders.\n");
  return;
// Function : OrdersBufInit
// Clears shared buffer for ORDERS, NEWORDER, and ORDERLINE
void OrdersBufInit()
           int
          int
                                j;
           for (i=0;i<orders_per_district;i++)
                     orders_buf[i].o_id = 0;
                     orders_buf[i].o_d_id = 0;
                     orders buf[i].o w id = 0;
                     orders_buf[i].o_c_id = 0;
                     orders buf[i].o carrier id = 0;
                     orders_buf[i].o_ol_cnt = 0;
                     orders buf[i].o all local = 0;
                     for (j=0;j<=14;j++)
                                orders_buf[i].o_ol[j].ol = 0;
                                orders_buf[i].o_ol[j].ol_i_id = 0;
                                orders_buf[i].o_ol[j].ol_supply_w_id = 0;
                                orders_buf[i].o_ol[j].ol_quantity = 0;
                                orders_buf[i].o_ol[j].ol_amount = 0;
                                strcpy(orders buf[i].o ol[j].ol dist info,"");
// Function : OrdersBufLoad
// Fills shared buffer for ORDERS, NEWORDER, and ORDERLINE
void OrdersBufLoad(int d id, int w id)
           int cust[ORDERS_PER_DISTRICT+1];
          long o_id;
           short ol;
           printf("...Loading Order Buffer for: d_id = %d, w id = %d\n",
                                d_id, w_id);
```

```
GetPermutation(cust, orders_per_district);
          for (o id=0;o id<orders per district;o id++)
                     // Generate ORDER and NEW-ORDER data
                     orders_buf[o_id].o_d_id = d_id;
                     orders_buf[o_id].o_w_id = w_id;
                     orders_buf[o_id].o_id = o_id+1;
                     orders_buf[o_id].o_c_id = cust[o_id+1];
                     orders_buf[o_id].o_ol_cnt = (short)RandomNumber(5L,
15L);
                     if (o id < first new order)
                                orders_buf[o_id].o_carrier_id =
(short)RandomNumber(1L, 10L);
                                orders_buf[o_id].o_all_local = 1;
                     else
                     {
                                orders_buf[o_id].o_carrier_id = 0;
                                orders_buf[o_id].o_all_local = 1;
                     for (ol=0; ol<orders buf[o id].o ol cnt; ol++)
                                orders_buf[o_id].o_ol[ol].ol = ol+1;
                                orders buf[o id].o ol[ol].ol i id =
RandomNumber(1L, max_items);
                                orders_buf[o_id].o_ol[ol].ol_supply_w_id =
w id;
                                orders_buf[o_id].o_ol[ol].ol_quantity = 5;
                                MakeAlphaString(24, 24,
OL_DIST_INFO_LEN, &orders_buf[o_id].o_ol[ol].ol_dist_info);
                                // Generate ORDER-LINE data
                                if (o_id < first_new_order)</pre>
orders_buf[o_id].o_ol[ol].ol_amount = 0;
                                           // Added to insure ol delivery d
set properly during load
FormatDate(&orders_buf[o_id].o_ol[ol].ol_delivery_d);
                                else
orders buf[o id].o ol[ol].ol amount = RandomNumber(1,999999)/100.0;
                                           // Added to insure ol delivery d
set properly during load
                                           // odbc datetime format
strcpy(orders_buf[o_id].o_ol[ol].ol_delivery_d,"1899-12-31 00:00:00:00.000");
          }
}
```

```
// Function : LoadOrdersTable
void LoadOrdersTable(LOADER TIME STRUCT *orders time start)
          int
  long
          o id;
                   o_d_id;
          short
          short
                   o_w_id;
  long
          o_c_id;
  short
          o_carrier_id;
  short
          o_ol_cnt;
          o_all_local;
  short
                             o entry d[O ENTRY D LEN+1];
          char
          RETCODE
                                       rc;
          DBINT
                             rcint;
         // bind ORDER data
  rc = bcp_bind(o_hdbc1, (BYTE *) &o_id, 0, SQL_VARLEN_DATA,
NULL, 0, SQLINT4, 1);
          if (rc != SUCCEED)
                   HandleErrorDBC(o_hdbc1);
  rc = bcp_bind(o_hdbc1, (BYTE *) &o_d_id, 0, SQL_VARLEN_DATA,
NULL, 0, SQLINT2, 2);
         if (rc != SUCCEED)
                   HandleErrorDBC(o_hdbc1);
  rc = bcp_bind(o_hdbc1, (BYTE *) &o_w_id, 0, SQL_VARLEN_DATA,
NULL, 0, SQLINT2, 3);
         if (rc != SUCCEED)
                   HandleErrorDBC(o hdbc1);
  rc = bcp_bind(o_hdbc1, (BYTE *) &o_c_id, 0, SQL_VARLEN_DATA,
NULL, 0, SQLINT4, 4);
         if (rc != SUCCEED)
                   HandleErrorDBC(o hdbc1);
         rc = bcp bind(o hdbc1, (BYTE *) &o entry d, 0,
O_ENTRY_D_LEN, NULL, 0, SQLCHARACTER, 5);
         if (rc != SUCCEED)
                   HandleErrorDBC(o hdbc1);
  rc = bcp bind(o hdbc1, (BYTE *) &o carrier id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 6);
          if (rc != SUCCEED)
                   HandleErrorDBC(o_hdbc1);
  rc = bcp_bind(o_hdbc1, (BYTE *) &o_ol_cnt, 0, SQL_VARLEN_DATA,
NULL, 0, SQLINT2, 7);
          if (rc != SUCCEED)
                   HandleErrorDBC(o_hdbc1);
  rc = bcp_bind(o_hdbc1, (BYTE *) &o_all_local, 0, SQL_VARLEN_DATA,
NULL, 0, SQLINT2, 8);
          if (rc != SUCCEED)
                   HandleErrorDBC(o_hdbc1);
  for (i = 0; i < orders_per_district; i++)
                            = orders_buf[i].o_id;
                   o_id
                   o_d_id
                             = orders_buf[i].o_d_id;
```

```
o w id
                              = orders_buf[i].o_w_id;
                    o c id
                              = orders_buf[i].o_c_id;
                                                                              rc = bcp_bind(o_hdbc2, (BYTE *) &o_d_id, 0, SQL_VARLEN_DATA,
                                                                            NULL, 0, SQLINT2, 2);
                    o carrier id = orders buf[i].o carrier id;
                    o_ol_cnt = orders_buf[i].o_ol_cnt;
                                                                                     if (rc != SUCCEED)
                    o_all_local = orders_buf[i].o_all_local;
                                                                                                HandleErrorDBC(o_hdbc2);
                    FormatDate(&o entry d);
                                                                              rc = bcp bind(o hdbc2, (BYTE *) &o w id, 0, SQL VARLEN DATA,
                                                                            NULL, 0, SQLINT2, 3);
                                                                                      if (rc != SUCCEED)
                    // send data to server
                    rc = bcp sendrow(o hdbc1);
                                                                                                HandleErrorDBC(o hdbc2);
                    if (rc != SUCCEED)
                               HandleErrorDBC(o_hdbc1);
                                                                                      for (i = first_new_order; i < last_new_order; i++)
                    orders_rows_loaded++;
                                                                                                o_id = orders_buf[i].o_id;
                    CheckForCommit(o hdbc1, o hstmt1,
                                                                                                o d id = orders_buf[i].o_d_id;
orders_rows_loaded, "orders", &orders_time_start->time_start);
                                                                                                o_w_id = orders_buf[i].o_w_id;
                                                                                                rc = bcp_sendrow(o_hdbc2);
          // rcint = bcp_batch(o_hdbc1);
                                                                                                if (rc != SUCCEED)
          // if (rcint < 0)
                                                                                                          HandleErrorDBC(o hdbc2);
                    HandleErrorDBC(o_hdbc1);
                                                                                                new order rows loaded++;
          if ((o_w_id == aptr->num_warehouses) && (o_d_id == 10))
                                                                                                CheckForCommit(o_hdbc2, o_hstmt2,
                                                                           new_order_rows_loaded, "new_order", &new_order_time_start->time_start);
                     rcint = bcp_done(o_hdbc1);
                    if (rcint < 0)
                               HandleErrorDBC(o hdbc1);
                                                                                      // rcint = bcp_batch(o_hdbc2);
                                                                                      // if (rcint < 0)
                     SQLFreeStmt(o hstmt1, SQL DROP);
                                                                                                HandleErrorDBC(o hdbc2);
                     SQLDisconnect(o_hdbc1);
                    SQLFreeConnect(o hdbc1);
                                                                                      if ((o w id == aptr->num warehouses) && (o d id == 10))
                    // if build index after load...
                                                                                                rcint = bcp_done(o_hdbc2);
                    if ((aptr->build index == 1) && (aptr->index order ==
                                                                                                if (rcint < 0)
0))
                                                                                                          HandleErrorDBC(o_hdbc2);
                               BuildIndex("idxordcl");
                                                                                                SQLFreeStmt(o_hstmt2, SQL_DROP);
                    // build non-clustered index
                                                                                                SQLDisconnect(o hdbc2);
                    if (aptr->build_index == 1)
                                                                                                SQLFreeConnect(o_hdbc2);
                               BuildIndex("idxordnc");
                                                                                                // if build index after load...
          }
                                                                                                if ((aptr->build_index == 1) && (aptr->index_order ==
                                                                           0))
}
                                                                                                          BuildIndex("idxnodcl");
// Function : LoadNewOrderTable
void LoadNewOrderTable(LOADER TIME STRUCT *new order time start)
                                                                            // Function : LoadOrderLineTable
  int
          i;
                                                                            //=
  long
          o id;
  short
          o_d_id;
                                                                            void LoadOrderLineTable(LOADER TIME STRUCT *order line time start)
  short
          o w id;
          RETCODE
                                         rc;
          DBINT
                               rcint;
                                                                                      int
                                                                                             i,j;
                                                                              long
                                                                                      o_id;
          // Bind NEW-ORDER data
                                                                                      short
                                                                                                o_d_id;
                                                                                      short
                                                                                                o_w_id;
  rc = bcp_bind(o_hdbc2, (BYTE *) &o_id, 0, SQL_VARLEN_DATA,
                                                                              long
                                                                                      ol:
NULL, 0, SQLINT4, 1);
                                                                                      long
                                                                                                ol i id;
          if (rc != SUCCEED)
                                                                              short
                                                                                      ol_supply_w_id;
                    HandleErrorDBC(o hdbc2);
                                                                              short
                                                                                      ol_quantity;
```

```
double
           ol amount;
                                                                                                         ol_supply_w_id =
  char
          ol dist info[DIST INFO LEN+1];
                                                                           orders_buf[i].o_ol[j].ol_supply_w_id;
                              ol_delivery_d[OL_DELIVERY_D_LEN+1];
          char
                                                                                                         ol_quantity
          RETCODE
                                                                           orders_buf[i].o_ol[j].ol_quantity;
          DBINT
                                                                                                         ol amount
                              reint:
                                                                           orders buf[i].o ol[j].ol amount;
          // bind ORDER-LINE data
  rc = bcp_bind(o_hdbc3, (BYTE *) &o_id, 0, SQL_VARLEN_DATA,
                                                                           strcpy(ol_delivery_d,orders_buf[i].o_ol[j].ol_delivery_d);
NULL, 0, SQLINT4, 1);
          if (rc != SUCCEED)
                    HandleErrorDBC(o_hdbc3);
                                                                           strcpy(ol_dist_info,orders_buf[i].o_ol[j].ol_dist_info);
  rc = bcp bind(o hdbc3, (BYTE *) &o d id, 0, SQL VARLEN DATA,
                                                                                                         rc = bcp sendrow(o hdbc3);
NULL, 0, SQLINT2, 2);
                                                                                                         if (rc != SUCCEED)
          if (rc != SUCCEED)
                                                                                                                   HandleErrorDBC(o hdbc3);
                    HandleErrorDBC(o hdbc3);
                                                                                                         order line rows loaded++;
  rc = bcp_bind(o_hdbc3, (BYTE *) &o_w_id, 0, SQL_VARLEN_DATA,
                                                                                                         CheckForCommit(o_hdbc3, o_hstmt3,
NULL, 0, SQLINT2, 3);
                                                                           order_line_rows_loaded, "order_line", &order_line_time_start->time_start);
          if (rc != SUCCEED)
                    HandleErrorDBC(o_hdbc3);
  rc = bcp_bind(o_hdbc3, (BYTE *) &ol, 0, SQL_VARLEN_DATA, NULL,
0, SQLINT4, 4);
                                                                                     // rcint = bcp_batch(o_hdbc3);
          if (rc != SUCCEED)
                                                                                     // if (rcint \leq 0)
                    HandleErrorDBC(o_hdbc3);
                                                                                               HandleErrorDBC(o hdbc3);
  rc = bcp_bind(o_hdbc3, (BYTE *) &ol_i_id, 0, SQL_VARLEN_DATA,
                                                                                     if ((o_w_id == aptr->num_warehouses) && (o_d_id == 10))
NULL, 0, SQLINT4, 5);
          if (rc != SUCCEED)
                                                                                               rcint = bcp_done(o_hdbc3);
                    HandleErrorDBC(o hdbc3);
                                                                                               if (rcint < 0)
                                                                                                         HandleErrorDBC(o_hdbc3);
  rc = bcp_bind(o_hdbc3, (BYTE *) &ol_supply_w_id, 0,
SQL VARLEN DATA, NULL, 0, SQLINT2, 6);
                                                                                               SQLFreeStmt(o hstmt3, SQL DROP);
          if (rc != SUCCEED)
                                                                                               SQLDisconnect(o_hdbc3);
                    HandleErrorDBC(o hdbc3);
                                                                                               SQLFreeConnect(o hdbc3);
          rc = bcp bind(o hdbc3, (BYTE *) &ol delivery d, 0,
                                                                                               // if build index after load...
OL_DELIVERY_D_LEN, NULL, 0, SQLCHARACTER, 7);
                                                                                               if ((aptr->build_index == 1) && (aptr->index_order ==
          if (rc != SUCCEED)
                                                                          0))
                    HandleErrorDBC(o hdbc3);
                                                                                                         BuildIndex("idxodlcl");
  rc = bcp bind(o hdbc3, (BYTE *) &ol quantity, 0, SQL VARLEN DATA,
                                                                                     }
NULL, 0, SQLINT2, 8);
          if (rc != SUCCEED)
                    HandleErrorDBC(o_hdbc3);
  rc = bcp bind(o hdbc3, (BYTE *) &ol amount, 0, SQL VARLEN DATA,
NULL, 0, SQLFLT8, 9);
          if (rc != SUCCEED)
                    HandleErrorDBC(o_hdbc3);
                                                                           // Function : GetPermutation
  rc = bcp_bind(o_hdbc3, (BYTE *) ol_dist_info, 0, DIST_INFO_LEN,
NULL, 0, 0, 10);
          if (rc != SUCCEED)
                    HandleErrorDBC(o_hdbc3);
                                                                           void GetPermutation(int perm[], int n)
          for (i = 0; i < orders_per_district; i++)
                                                                                     int i, r, t;
                    o_id = orders_buf[i].o_id;
                                                                                     for (i=1;i \le n;i++)
                    o d id = orders_buf[i].o_d_id;
                                                                                               perm[i] = i;
                    o_w_id = orders_buf[i].o_w_id;
                                                                                     for (i=1;i<=n;i++)
                    for (j=0; j < orders\_buf[i].o\_ol\_cnt; j++)
                                                                                               r = RandomNumber(i,n);
                                       = orders buf[i].o ol[j].ol;
                                                                                               t = perm[i];
                                                                                               perm[i] = perm[r];
                              ol_i_id
                                         = orders_buf[i].o_ol[j].ol_i_id;
                                                                                               perm[r] = t;
```

```
}
                                                                                 SQLAllocHandle(SQL_HANDLE_ENV, SQL_NULL_HANDLE,
                                                                        &henv);
}
                                                                                 SQLSetEnvAttr(henv, SQL_ATTR_ODBC_VERSION,
                                                                        (void*)SQL_OV_ODBC3, 0 );
                                                                                 SQLAllocHandle(SQL_HANDLE_DBC, henv, &i_hdbc1);
                                                                                 SQLAllocHandle(SQL_HANDLE_DBC, henv, &w_hdbc1); SQLAllocHandle(SQL_HANDLE_DBC, henv, &c_hdbc1);
// Function : CheckForCommit
                                                                                 SQLAllocHandle(SQL HANDLE DBC, henv, &c hdbc2);
                                                                                 SQLAllocHandle(SQL_HANDLE_DBC, henv, &o_hdbc1);
                                                                                 SQLAllocHandle(SQL_HANDLE_DBC, henv, &o_hdbc2);
                                                                                 SQLAllocHandle(SQL HANDLE DBC, henv, &o hdbc3);
void CheckForCommit(HDBC hdbc,
                                                                                 SQLSetConnectAttr(i_hdbc1, SQL_COPT_SS_BCP, (void
                                                 HSTMT hstmt,
                                       int rows loaded,
                                                                        *)SQL_BCP_ON, SQL_IS_INTEGER );
                                                char *table_name,
                                                                                 SQLSetConnectAttr(w hdbc1, SQL COPT SS BCP, (void
                                                                        *)SQL_BCP_ON, SQL_IS_INTEGER );
                                       long *time_start)
                                                                                 SQLSetConnectAttr(c_hdbc1, SQL_COPT_SS_BCP, (void
                                                                        *)SQL BCP ON, SQL IS INTEGER);
                                                                                 SQLSetConnectAttr(c_hdbc2, SQL_COPT_SS_BCP, (void
         time_end, time_diff;
  long
          // DBINT rcint;
                                                                        *)SQL BCP ON, SQL IS INTEGER);
                                                                                 SQLSetConnectAttr(o_hdbc1, SQL_COPT_SS_BCP, (void
                                                                        *)SOL BCP ON, SQL IS INTEGER);
  if (!(rows loaded % aptr->batch))
                                                                                 SQLSetConnectAttr(o_hdbc2, SQL_COPT_SS_BCP, (void
                                                                        *)SQL_BCP_ON, SQL_IS_INTEGER );
                   // reint = bcp batch(hdbc);
                                                                                 SQLSetConnectAttr(o hdbc3, SQL COPT SS BCP, (void
                                                                        *)SQL_BCP_ON, SQL_IS_INTEGER );
                   // if (rcint < 0)
                             HandleErrorDBC(hdbc);
                                                                                 // Open connections to SQL Server
                   time end = (TimeNow() / MILLI);
                   time_diff = time_end - *time_start;
                                                                                 // Connection 1
                   printf("-> Loaded %ld rows into %s in %ld sec - Total =
                                                                                 sprintf( szDriverString , "DRIVER={SQL
%d (%.2f rps)\n",
                                                                        Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s",
                             aptr->batch,
                             table_name,
                                                                        aptr->server,
                                       time diff,
                                                                                                                                    aptr->user,
                             rows_loaded,
                                       (float) aptr->batch / (time diff?
                                                                        aptr->password,
time diff: 1L));
                                                                        aptr->database );
                   *time start = time end;
                                                                                 rc = SQLSetConnectOption (i_hdbc1, SQL_PACKET_SIZE,
  }
                                                                        aptr->pack size);
                                                                                 if (rc != SUCCEED)
  return;
                                                                                           HandleErrorDBC(i hdbc1);
                                                                                 rc = SQLDriverConnect ( i_hdbc1,
                                                                                                                           NULL,
                                                                        (SQLCHAR*)&szDriverString[0],
                                                                                                                                   SQL_NTS,
// Function : OpenConnections
                                                                        (SQLCHAR*)&szDriverStringOut[0],
                                                                        sizeof(szDriverStringOut),
void OpenConnections()
                                                                        &cbDriverStringOut,
          RETCODE
                                                                        SOL DRIVER NOPROMPT);
                        rc:
                                                                                 if (rc != SUCCEED)
                                       szDriverString[300];
                                                                                           HandleErrorDBC(i_hdbc1);
          char
          char
                                       szDriverStringOut[1024];
          SQLSMALLINT
                                       cbDriverStringOut;
                                                                                 // Connection 2
                                                                                 sprintf( szDriverString , "DRIVER={SQL
                                                                        Server\;SERVER=\%s;UID=\%s;PWD=\%s;DATABASE=\%s",
```

```
sprintf( szDriverString , "DRIVER={SQL
aptr->server,
                                                                       Server\;SERVER=\%s;UID=\%s;PWD=\%s;DATABASE=\%s",
                                                           aptr->user.
                                                                       aptr->server,
aptr->password,
                                                                                                                                   aptr->user,
aptr->database);
                                                                       aptr->password,
         rc = SQLSetConnectOption (w_hdbc1, SQL_PACKET_SIZE,
                                                                       aptr->database );
aptr->pack_size);
         if (rc != SUCCEED)
                                                                                 rc = SQLSetConnectOption (c_hdbc2, SQL_PACKET_SIZE,
                   HandleErrorDBC(w_hdbc1);
                                                                       aptr->pack_size);
                                                                                 if (rc != SUCCEED)
                                                                                           HandleErrorDBC(c_hdbc2);
         rc = SQLDriverConnect (w_hdbc1,
NULL,
                                                                                 rc = SQLDriverConnect ( c_hdbc2,
(SQLCHAR*)&szDriverString[0],
                                                                       NULL,
SQL NTS,
                                                                       (SQLCHAR*)&szDriverString[0],
(SQLCHAR*)&szDriverStringOut[0],
                                                                       SQL_NTS,
sizeof(szDriverStringOut),
                                                                       (SQLCHAR*)&szDriverStringOut[0],
&cbDriverStringOut,
                                                                       sizeof(szDriverStringOut),
SQL_DRIVER_NOPROMPT);
                                                                       &cbDriverStringOut,
         if (rc != SUCCEED)
                                                                       SQL DRIVER NOPROMPT);
                   HandleErrorDBC(w_hdbc1);
                                                                                 if (rc != SUCCEED)
         // Connection 3
                                                                                           HandleErrorDBC(c_hdbc2);
         sprintf( szDriverString , "DRIVER={SQL
                                                                                 // Connection 5
Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s",
                                                                                 sprintf( szDriverString , "DRIVER={SQL
                                                                       Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s",
aptr->server,
                                                           aptr->user,
                                                                       aptr->server,
aptr->password,
                                                                                                                                   aptr->user,
aptr->database );
                                                                       aptr->password,
         rc = SQLSetConnectOption (c_hdbc1, SQL_PACKET_SIZE,
                                                                       aptr->database );
aptr->pack size);
         if (rc != SUCCEED)
                                                                                 rc = SQLSetConnectOption (o_hdbc1, SQL_PACKET_SIZE,
                   HandleErrorDBC(c hdbc1);
                                                                       aptr->pack size);
                                                                                 if (rc != SUCCEED)
                                                                                           HandleErrorDBC(o_hdbc1);
         rc = SQLDriverConnect ( c_hdbc1,
NULL,
                                                                                 rc = SQLDriverConnect (o_hdbc1,
(SQLCHAR*)&szDriverString[0],
                                                                       NULL,
                                                                       (SQLCHAR*)&szDriverString[0],
SQL_NTS,
(SQLCHAR*)&szDriverStringOut[0],
                                                                       SQL NTS,
sizeof(szDriverStringOut),
                                                                       (SQLCHAR*)&szDriverStringOut[0],
&cbDriverStringOut,
                                                                       sizeof(szDriverStringOut),
SQL DRIVER NOPROMPT);
                                                                       &cbDriverStringOut,
         if (rc != SUCCEED)
                                                                       SQL_DRIVER_NOPROMPT);
                   HandleErrorDBC(c_hdbc1);
                                                                                 if (rc != SUCCEED)
         // Connection 4
                                                                                           HandleErrorDBC(o_hdbc1);
```

```
// Connection 6
                                                                                              HandleErrorDBC(o_hdbc3);
          sprintf( szDriverString , "DRIVER={SQL
Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s",
aptr->server,
                                                             aptr->user,
                                                                          // Function name: BuildIndex
aptr->password,
aptr->database );
          rc = SQLSetConnectOption (o hdbc2, SQL PACKET SIZE,
                                                                          void BuildIndex(char *index script)
aptr->pack_size);
          if (rc != SUCCEED)
                                                                                    char
                                                                                              cmd[256];
                    HandleErrorDBC(o_hdbc2);
                                                                                    printf("Starting index creation: %s\n",index script);
          rc = SQLDriverConnect (o_hdbc2,
                                                                                    sprintf(cmd, "isql -S%s -U%s -P%s -e -i%s\\%s.sql > logs\\%s.log",
NULL,
                                                                                                                  aptr->server,
                                                                                                                  aptr->user.
(SQLCHAR*)&szDriverString[0],
                                                                                                                  aptr->password,
                                                                                                                  aptr->index_script_path,
                                                                                                                  index script,
SQL_NTS,
                                                                                                                  index_script);
(SQLCHAR*)&szDriverStringOut[0],
                                                                                    system(cmd);
sizeof(szDriverStringOut),
                                                                                    printf("Finished index creation: %s\n",index script);
&cbDriverStringOut,
SQL_DRIVER_NOPROMPT);
          if (rc != SUCCEED)
                                                                          void HandleErrorDBC (SQLHDBC hdbc1)
                    HandleErrorDBC(o hdbc2);
          // Connection 7
                                                                                    SQLCHAR
                                                                                                                  SqlState[6],
                                                                          Msg[SQL_MAX_MESSAGE_LENGTH];
          sprintf( szDriverString , "DRIVER={SQL
                                                                                    SQLINTEGER NativeError;
Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s",
                                                                                    SQLSMALLINT i, MsgLen;
                                                                                    SQLRETURN rc2;
aptr->server,
                                                                                    char
                                                                                                        timebuf[128];
                                                                                                        datebuf[128];
                                                             aptr->user,
                                                                                    char
                                                                                    FILE
                                                                                                        *fp1;
aptr->password,
                                                                                    i = 1;
                                                                                    while (( rc2 = SQLGetDiagRec(SQL_HANDLE_DBC, hdbc1, i,
aptr->database );
                                                                          SqlState, &NativeError,
          rc = SQLSetConnectOption (o_hdbc3, SQL_PACKET_SIZE,
                                                                                                                    Msg, sizeof(Msg), &MsgLen))
                                                                          != SQL_NO_DATA )
aptr->pack_size);
          if (rc != SUCCEED)
                    HandleErrorDBC(o_hdbc3);
                                                                                              sprintf( szLastError , "%s" , Msg );
          rc = SQLDriverConnect (o_hdbc3,
                                                                                              _strtime(timebuf);
NULL,
                                                                                              strdate(datebuf);
(SQLCHAR*)&szDriverString[0],
                                                                                              printf( "[%s : %s] %s\n" , datebuf, timebuf, szLastError);
                                                                                              fp1 = fopen("logs\\tpccldr.err","w");
SQL NTS,
                                                                                              if(fp1 == NULL)
(SQLCHAR*)&szDriverStringOut[0],
                                                                                                        printf("ERROR: Unable to open errorlog
                                                                          file.\n");
sizeof(szDriverStringOut),
                                                                                              else
&cbDriverStringOut,
                                                                                                        fprintf(fp1, "[%s: %s] %s\n", datebuf,
                                                                          timebuf, szLastError);
SQL_DRIVER_NOPROMPT);
                                                                                                        fclose(fp1);
          if (rc != SUCCEED)
```

```
i++;
}
void HandleErrorSTMT (HSTMT hstmt1)
          SQLCHAR
                                         SqlState[6],
Msg[SQL\_MAX\_MESSAGE\_LENGTH];
          SQLINTEGER NativeError;
          SQLSMALLINT i, MsgLen;
          SQLRETURN rc2;
          char
                              timebuf[128];
                              datebuf[128];
          char
          FILE
                              *fp1;
          i = 1
          while (( rc2 = SQLGetDiagRec(SQL HANDLE STMT, hstmt1, i,
SqlState, &NativeError,
                                          Msg, sizeof(Msg), &MsgLen))
!= SQL_NO_DATA)
                    sprintf( szLastError , "%s" , Msg );
                     _strtime(timebuf);
                    strdate(datebuf);
                    printf( "[%s : %s] %s\n" , datebuf, timebuf, szLastError);
                    fp1 = fopen("logs\\tpccldr.err","w");
                    if (fp1 == NULL)
                              printf("ERROR: Unable to open errorlog
file.\n");
                    else
                              fprintf(fp1, "[%s: %s] %s\n", datebuf,
timebuf, szLastError);
                              fclose(fp1);
                    i++:
}
void FormatDate ( char* szTimeCOutput )
          struct tm when;
          time_t now;
          time( &now );
          when = *localtime( &now );
          mktime( &when );
          // odbc datetime format
          strftime( szTimeCOutput , 30 , "%Y-%m-%d %H:%M:%S.000",
&when);
          return;
```

```
// Function : CheckSQL
void CheckSQL()
         RETCODE
                       rc:
                                     szDriverString[300];
         char
                                     szDriverStringOut[1024];
         char
                                              SQLBuildFlag;
         int
         char
                                     resp;
         SQLSMALLINT
                                     cbDriverStringOut;
         SQLCHAR
                                              SQLVersion[19];
         SQLINTEGER
                                     SQLVersionInd;
         SQLAllocHandle(SQL_HANDLE_ENV, SQL_NULL_HANDLE,
&henv);
         SQLSetEnvAttr(henv, SQL_ATTR_ODBC_VERSION,
(void*)SQL OV ODBC3, 0);
         SQLAllocHandle(SQL HANDLE DBC, henv, &v hdbc);
         SQLSetConnectAttr(v_hdbc, SQL_COPT_SS_BCP, (void
*)SQL_BCP_ON, SQL_IS_INTEGER );
         // Open connection to SQL Server
         sprintf( szDriverString , "DRIVER={SQL
Server};SERVER=%s;UID=%s;PWD=%s",
aptr->server,
                                                         aptr->user,
aptr->password );
         if ( SQLSetConnectAttr( v_hdbc, SQL_ATTR_PACKET_SIZE,
(SQLPOINTER)aptr->pack_size, SQL_IS_UINTEGER) != SQL_SUCCESS)
                  HandleErrorDBC(v hdbc);
         rc = SQLDriverConnect (v hdbc,
                                                NULL,
(SQLCHAR*)&szDriverString[0],
                                                        SQL_NTS,
(SQLCHAR*)&szDriverStringOut[0],
sizeof(szDriverStringOut),
&cbDriverStringOut,
SQL_DRIVER_NOPROMPT);
         if ((rc != SQL_SUCCESS) && (rc !=
SQL_SUCCESS_WITH_INFO))
                  HandleErrorDBC(v_hdbc);
```

```
if ( SQLAllocHandle(SQL_HANDLE_STMT, v_hdbc , &v_hstmt)
!= SQL SUCCESS)
                    HandleErrorSTMT(v hstmt);
          rc = SQLBindCol(v_hstmt, 4, SQL_C_CHAR, &SQLVersion,
sizeof(SQLVersion), &SQLVersionInd);
          // issue SQL Server extended stored procedure (xp msver) to
determine installed version
          rc = SQLExecDirect(v hstmt, "EXECUTE xp msver
ProductVersion", SQL NTS);
          if ((rc != SQL_SUCCESS) && (rc !=
SQL_SUCCESS_WITH_INFO))
                    HandleErrorSTMT(v hstmt);
          rc = SQLFetch(v_hstmt);
          if (rc != SQL_SUCCESS)
                    HandleErrorDBC(v_hdbc);
          // Check build number to ensure 8.00.194 or higher
          SQLBuildFlag = 1;
          // first check the Major version
          if (SQLVersion[0] == '8')
                    if (( SQLVersion[2] == '0') & ( SQLVersion[3] == '0' ) )
                              if (SQLVersion[5] == '1')
                                         if ( (SQLVersion[6] == '9') &
(SQLVersion[7] == '4')
                                                   SQLBuildFlag = 0;
                                                   printf("You are using
SQL Server version = \%9s\n', SQLVersion;
                                         else
                                                   SQLBuildFlag = 1;
                              else
                                         if (SQLVersion[5] == '3')
                                                   if ((SQLVersion[6]>=
53) & (SQLVersion[7] >= 48))
SQLBuildFlag = 0;
                                                             printf("You
are using SQL Server version = \%9s\n\n", SQLVersion);
                                                   else
SQLBuildFlag = 1;
                    }
          else
```

```
SQLBuildFlag = 1;
         if (SQLBuildFlag == 1)
                   printf("NOTE: The SQL Server version you are using is
not supported\n");
                   printf("for TPC-C benchmarking. You currently have
SQL Server version %9s\n",SQLVersion);
                   printf("installed. Please upgrade to Microsoft SQL
Server 2000 (8.00.0194) or better.\n");
                   printf("and re-run the SETUP program.\n\n");
                   printf("Do you wish to continue with setup? (Y/N): ");
                   resp = getchar();
                   if ((resp == 'N') | (resp == 'n'))
                             printf("\nSetup Aborted!\n");
                             exit(1);
         SQLFreeHandle(SQL HANDLE STMT, v hstmt);
         SQLDisconnect(v hdbc);
         SQLFreeHandle(SQL HANDLE DBC, v hdbc);
         return;
// Function : CheckDataBase
void CheckDataBase()
         RETCODE
                        rc;
                                       szDriverString[300];
         char
                                       szDriverStringOut[1024];
         char
                                       TablesBitMap[9] =
         char
{"000000000"};
                                                 i, ExitFlag;
         SQLSMALLINT
                                       cbDriverStringOut;
         SQLCHAR
                                                 TabName[10];
         SQLINTEGER
                                       TabNameInd, TabCount,
TabCountInd;
         ExitFlag = 0;
         SQLAllocHandle(SQL_HANDLE_ENV, SQL_NULL_HANDLE,
&henv);
         SQLSetEnvAttr(henv, SQL ATTR ODBC VERSION,
(void*)SQL_OV_ODBC3, 0);
         SQLAllocHandle(SQL_HANDLE_DBC, henv, &v_hdbc);
         SQLSetConnectAttr(v hdbc, SQL COPT SS BCP, (void
*)SQL_BCP_ON, SQL_IS_INTEGER );
```

```
// Open connection to SQL Server
          sprintf( szDriverString , "DRIVER={SQL
Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s",
aptr->server,
                                                            aptr->user,
aptr->password,
aptr->database );
         rc = SQLSetConnectAttr( v hdbc, SQL ATTR PACKET SIZE,
(SQLPOINTER)aptr->pack_size, SQL_IS_UINTEGER);
          if (rc != SQL SUCCESS)
                   HandleErrorDBC(v hdbc);
          rc = SQLDriverConnect (v_hdbc,
                                                   NULL,
(SQLCHAR*)&szDriverString[0],
                                                           SQL NTS,
(SQLCHAR*)&szDriverStringOut[0],
sizeof(szDriverStringOut),
&cbDriverStringOut,
SQL DRIVER NOPROMPT);
         // if the rc is SQL_ERROR, the the TPCC database probably does
not exist
          if (rc == SQL ERROR)
                    printf("The database TPCC does not appear to exist!\n");
                   printf("\nCheck LOGS\\ directory for database creation
errors.\n");
                   // cleanup database connections and handles
                    SQLFreeHandle(SQL HANDLE STMT, v hstmt);
                    SQLDisconnect(v hdbc);
                    SQLFreeHandle(SQL HANDLE DBC, v hdbc);
                   // since there is not a database, exit back to SETUP.CMD
                   exit(1);
          }
          if ( SQLAllocHandle(SQL_HANDLE_STMT, v_hdbc , &v_hstmt)
!= SQL SUCCESS)
                   HandleErrorDBC(v_hdbc);
         if (SQLBindCol(v_hstmt, 1, SQL_C_ULONG, &TabCount, 0,
&TabCountInd) != SQL_SUCCESS )
                   HandleErrorSTMT(v hstmt);
          // count the number of user tables from sysobjects
         rc = SQLExecDirect(v_hstmt, "select count(*) from sysobjects
where xtype = \U\, SQL NTS);
          if ((rc != SQL_SUCCESS) && (rc !=
SQL SUCCESS WITH INFO))
                    HandleErrorSTMT(v_hstmt);
          if ( SQLFetch(v_hstmt) != SQL_SUCCESS )
                   HandleErrorSTMT(v_hstmt);
         // if the number of tables is less than 9, select all the user tables in
TPCC
```

```
if (TabCount != 9)
                     SQLFreeHandle(SQL HANDLE STMT, v hstmt);
                     SQLAllocHandle(SQL\_HANDLE\_STMT, v\_hdbc\;,
&v hstmt);
                     if (SQLBindCol(v hstmt, 1, SQL C CHAR,
&TabName, sizeof(TabName), &TabNameInd) != SQL SUCCESS)
                                HandleErrorSTMT(v hstmt);
                     // select the list of user tables into a result set
                     rc = SQLExecDirect(v hstmt, "select * from sysobjects
where xtype = \U\, SQL_NTS);
                     if ((rc!= SQL SUCCESS) && (rc!=
SQL_SUCCESS_WITH_INFO))
                                HandleErrorSTMT(v hstmt);
                     // go through the result set and set the bitmap for each
found table
                     // set the bitmap to '1' if the table name is found
                     while ((rc = SQLFetch(v hstmt)) != SQL NO DATA)
                                switch( TabName[0] )
                                case 'w':
                                           TablesBitMap[0] = '1';
                                           break;
                                case 'd'-
                                           TablesBitMap[1] = '1';
                                           break;
                                case 'c':
                                           TablesBitMap[2] = '1';
                                           break;
                                case 'h':
                                           TablesBitMap[3] = '1';
                                           break;
                                case 'n':
                                           TablesBitMap[4] = '1';
                                           break;
                                case 'o':
                                           if (TabName[5] = 's')
                                                      TablesBitMap[5] = '1';
                                           if (TabName[5] = ' ')
                                                      TablesBitMap[6] = '1';
                                           break:
                                case 'i':
                                           TablesBitMap[7] = '1';
                                           break;
                                case 's':
                                           TablesBitMap[8] = '1';
                                           break;
                     // a '0' ExitFlag means do NOT exit the loader early, a '1'
means exit the loader early
                     ExitFlag = 0;
                     // interate through the bitmap to display which table(s) is
actually missing
                     for (i = 0; i \le 8; i++)
                                switch(i)
                                case 0:
                                           if(TablesBitMap[i] == '0')
```

```
printf("The Warehouse
                                                                                                                       break;
table is missing or damaged.\n");
                                                    ExitFlag = 1;
                                          break;
                                                                                                  // if one or more tables are missing, display message and
                               case 1:
                                                                             exit the loader
                                          if (TablesBitMap[i] == '0')
                                                                                                  if(ExitFlag = 1)
                                                                                                             printf("\nExiting TPC-C Loader!\n");
                                                    printf("The District
table is missing or damaged.\n");
                                                                                                             printf("\nCheck LOGS\\ directory for
                                                    ExitFlag = 1;
                                                                             database\n");
                                                                                                             printf("or table creation errors.\n");
                                          break;
                               case 2:
                                                                                                             // cleanup database connections and handles
                                          if(TablesBitMap[i] == '0')
                                                                                                             SQLFreeHandle(SQL HANDLE STMT,
                                                                             v hstmt);
                                                    printf("The Customer
                                                                                                             SQLDisconnect(v_hdbc);
table is missing or damaged.\n");
                                                                                                             SQLFreeHandle(SQL_HANDLE_DBC,
                                                    ExitFlag = 1;
                                                                             v hdbc);
                                          break;
                                                                                                             exit(1);
                               case 3:
                                          if (TablesBitMap[i] == '0')
                                                                                        // cleanup database connections and handles
                                                    printf("The History
table is missing or damaged.\n");
                                                                                        SQLFreeHandle(SQL HANDLE STMT, v hstmt);
                                                                                        SQLDisconnect(v_hdbc);
                                                    ExitFlag = 1;
                                                                                        SQLFreeHandle(SQL HANDLE DBC, v hdbc);
                                          break;
                               case 4:
                                                                                        return;
                                          if (TablesBitMap[i] == '0')
                                                    printf("The New Order
table is missing or damaged.\n");
                                                                             tpccldr.mak
                                                    ExitFlag = 1;
                                                                             # Microsoft Developer Studio Generated NMAKE File, Format Version 4.10
                                          break;
                                                                             # ** DO NOT EDIT **
                               case 5:
                                          if (TablesBitMap[i] == '0')
                                                                             # TARGTYPE "Win32 (x86) Console Application" 0x0103
                                                    printf("The Orders table
                                                                             !IF "$(CFG)" == ""
is missing or damaged.\n");
                                                                             CFG=tpccldr - Win32 Debug
                                                    ExitFlag = 1;
                                                                             !MESSAGE No configuration specified. Defaulting to tpccldr - Win32 Debug.
                                                                             !ENDIF
                                          break;
                               case 6:
                                                                             !IF "$(CFG)" != "tpccldr - Win32 Release" && "$(CFG)" !=\
                                          if (TablesBitMap[i] == '0')
                                                                              "tpccldr - Win32 Debug"
                                                                             !MESSAGE Invalid configuration "$(CFG)" specified.
                                                    printf("The Order Line
                                                                             !MESSAGE You can specify a configuration when running NMAKE on this
table is missing or damaged.\n");
                                                                             makefile
                                                    ExitFlag = 1;
                                                                             !MESSAGE by defining the macro CFG on the command line. For example:
                                                                             !MESSAGE
                                          break;
                                                                             !MESSAGE NMAKE /f "tpccldr.mak" CFG="tpccldr - Win32 Debug"
                               case 7:
                                                                             !MESSAGE
                                          if (TablesBitMap[i] == '0')
                                                                             !MESSAGE Possible choices for configuration are:
                                                                             !MESSAGE
                                                    printf("The Item table is
                                                                             !MESSAGE "tpccldr - Win32 Release" (based on "Win32 (x86) Console
missing or damaged.\n");
                                                                             Application")
                                                    ExitFlag = 1;
                                                                             !MESSAGE "tpccldr - Win32 Debug" (based on "Win32 (x86) Console
                                                                             Application")
                                          break;
                                                                             !MESSAGE
                               case 8:
                                                                             !ERROR An invalid configuration is specified.
                                          if (TablesBitMap[i] == '0')
                                                                             !ENDIF
                                                    printf("The Stock table
                                                                             !IF "$(OS)" == "Windows NT"
is missing or damaged.\n");
                                                                             NULL=
                                                    ExitFlag = 1;
                                                                             !ELSE
```

```
NULL=nul
                                                                        winspool.lib comdlg32.lib advapi32.lib shell32.lib ole32.lib oleaut32.lib\
!ENDIE
                                                                        uuid.lib odbc32.lib odbccp32.lib /nologo /subsystem:console /pdb:none\
                                                                        /machine:I386 /out:"$(OUTDIR)/tpccldr.exe"
LINK32 OBJS=
                                                                                  "$(INTDIR)\getargs.obj" \
# Begin Project
# PROP Target Last Scanned "tpccldr - Win32 Debug"
                                                                                 "$(INTDIR)\random.obj" \
RSC=rc.exe
                                                                                 "$(INTDIR)\strings.obj" \
                                                                                 "$(INTDIR)\time.obj"
CPP=cl.exe
                                                                                 "$(INTDIR)\tpccldr.obj"
!IF "$(CFG)" == "tpccldr - Win32 Release"
                                                                        "$(OUTDIR)\tpccldr.exe" : "$(OUTDIR)" $(DEF_FILE) $(LINK32_OBJS)
# PROP BASE Use MFC 0
                                                                         $(LINK32) @<<
                                                                        $(LINK32 FLAGS) $(LINK32 OBJS)
# PROP BASE Use Debug Libraries 0
# PROP BASE Output_Dir "Release"
# PROP BASE Intermediate Dir "Release"
# PROP BASE Target Dir "
                                                                       !ELSEIF "$(CFG)" == "tpccldr - Win32 Debug"
# PROP Use MFC 0
# PROP Use_Debug_Libraries 0
                                                                       # PROP BASE Use_MFC 0
# PROP Output Dir "bin"
                                                                       # PROP BASE Use Debug Libraries 1
# PROP Intermediate Dir "objects"
                                                                       # PROP BASE Output Dir "Debug"
# PROP Target_Dir "
                                                                       # PROP BASE Intermediate_Dir "Debug"
OUTDIR=.\bin
                                                                       # PROP BASE Target Dir "
                                                                       # PROP Use_MFC 0
INTDIR=.\objects
                                                                       # PROP Use Debug_Libraries 1
ALL: "$(OUTDIR)\tpccldr.exe"
                                                                       # PROP Output_Dir "bin"
                                                                       # PROP Intermediate_Dir "objects"
# PROP Target_Dir ""
CLEAN:
                                                                       OUTDIR=.\bin
          -@erase "$(INTDIR)\getargs.obj"
         -@erase "$(INTDIR)\random.obj"
                                                                       INTDIR=.\objects
          -@erase "$(INTDIR)\strings.obj"
          -@erase "$(INTDIR)\time.obj"
                                                                       ALL: "$(OUTDIR)\tpccldr.exe"
          -@erase "$(INTDIR)\tpccldr.obj"
          -@erase "$(OUTDIR)\tpccldr.exe"
                                                                       CLEAN:
                                                                                 -@erase "$(INTDIR)\getargs.obj"
"$(OUTDIR)":
                                                                                 -@erase "$(INTDIR)\random.obj"
                                                                                 -@erase "$(INTDIR)\strings.obj'
  if not exist "$(OUTDIR)/$(NULL)" mkdir "$(OUTDIR)"
                                                                                 -@erase "$(INTDIR)\time.obj"
                                                                                 -@erase "$(INTDIR)\tpccldr.obj"
  if not exist "$(INTDIR)/$(NULL)" mkdir "$(INTDIR)"
                                                                                 -@erase "$(INTDIR)\vc40.idb"
                                                                                 -@erase "$(INTDIR)\vc40.pdb"
# ADD BASE CPP /nologo /W3 /GX /O2 /D "WIN32" /D "NDEBUG" /D
                                                                                 -@erase "$(OUTDIR)\tpccldr.exe"
 CONSOLE" /YX /c
#ADD CPP /nologo /MT /W3 /GX /O2 /I "c:\mssql\dblib\include" /D
                                                                        "$(OUTDIR)"
"NDEBUG" /D "WIN32" /D "_CONSOLE" /D "DBNTWIN32" /c
                                                                         if not exist "$(OUTDIR)/$(NULL)" mkdir "$(OUTDIR)"
# SUBTRACT CPP /YX
CPP_PROJ=/nologo /MT /W3 /GX /O2 /I "c:\mssql\dblib\include" /D
                                                                        "$(INTDIR)" :
"NDEBUG" /D\
                                                                          if not exist "$(INTDIR)/$(NULL)" mkdir "$(INTDIR)"
"WIN32" /D " CONSOLE" /D "DBNTWIN32" /Fo"$(INTDIR)/" /c
CPP_OBJS=.\objects/
                                                                       # ADD BASE CPP /nologo /W3 /Gm /GX /Zi /Od /D "WIN32" /D "_DEBUG"
CPP SBRS=.\.
                                                                       /D " CONSOLE" /YX /c
# ADD BASE RSC /l 0x409 /d "NDEBUG"
                                                                       # ADD CPP /nologo /MTd /W3 /Gm /GX /Zi /Od /I "c:\mssql\dblib\include" /D
# ADD RSC /I 0x409 /d "NDEBUG"
                                                                       " DEBUG" /D "WIN32" /D " CONSOLE" /D "DBNTWIN32" /c
BSC32=bscmake.exe
                                                                       # SUBTRACT CPP /YX
# ADD BASE BSC32 /nologo
                                                                       CPP PROJ=/nologo /MTd /W3 /Gm /GX /Zi /Od /I "c:\mssql\dblib\include" /D\
                                                                        " DEBUG" /D "WIN32" /D " CONSOLE" /D "DBNTWIN32"
# ADD BSC32 /nologo
BSC32 FLAGS=/nologo /o"$(OUTDIR)/tpccldr.bsc"
                                                                       /Fo"$(INTDIR)/"\
BSC32 SBRS=\
                                                                        /Fd"$(INTDIR)/"/c
                                                                       CPP_OBJS=.\objects/
LINK32=link.exe
                                                                       CPP SBRS=.\.
                                                                       # ADD BASE RSC /l 0x409 /d "_DEBUG"
# ADD BASE LINK32 kernel32.lib user32.lib gdi32.lib winspool.lib
comdlg32.lib advapi32.lib shell32.lib ole32.lib oleaut32.lib uuid.lib odbc32.lib
                                                                       # ADD RSC /l 0x409 /d " DEBUG"
odbccp32.lib /nologo /subsystem:console /machine:I386
                                                                       BSC32=bscmake.exe
                                                                       # ADD BASE BSC32 /nologo
# ADD LINK32 c:\mssql\dblib\lib\ntwdblib.lib kernel32.lib user32.lib gdi32.lib
winspool.lib comdlg32.lib advapi32.lib shell32.lib ole32.lib oleaut32.lib
                                                                       # ADD BSC32 /nologo
uuid.lib odbc32.lib odbccp32.lib /nologo /subsystem:console /pdb:none
                                                                       BSC32_FLAGS=/nologo /o"$(OUTDIR)/tpccldr.bsc"
/machine:I386
                                                                       BSC32 SBRS=\
LINK32_FLAGS=c:\mssql\dblib\lib\ntwdblib.lib kernel32.lib user32.lib
gdi32.lib\
                                                                       LINK32=link.exe
```

```
"$(INTDIR)\random.obj": $(SOURCE) $(DEP CPP RANDO) "$(INTDIR)"
# ADD BASE LINK32 kernel32.lib user32.lib gdi32.lib winspool.lib
comdlg32.lib advapi32.lib shell32.lib ole32.lib oleaut32.lib uuid.lib odbc32.lib
                                                                $(CPP) $(CPP PROJ) $(SOURCE)
odbccp32.lib /nologo /subsystem:console /debug /machine:1386
# ADD LINK32 c:\mssql\dblib\lib\ntwdblib.lib kernel32.lib user32.lib gdi32.lib
winspool.lib comdlg32.lib advapi32.lib shell32.lib ole32.lib oleaut32.lib
                                                              # End Source File
                                                              uuid.lib odbc32.lib odbccp32.lib /nologo /subsystem:console /pdb:none /debug
                                                              /machine:I386
LINK32 FLAGS=c:\mssql\dblib\lib\ntwdblib.lib kernel32.lib user32.lib
                                                              # Begin Source File
gdi32.lib\
winspool.lib comdlg32.lib advapi32.lib shell32.lib ole32.lib oleaut32.lib
                                                              SOURCE=.\src\strings.c
uuid.lib odbc32.lib odbccp32.lib /nologo /subsystem:console /pdb:none /debug\
                                                              DEP_CPP_STRIN=\
/machine:I386 /out:"$(OUTDIR)/tpccldr.exe"
                                                                       ".\src\tpcc.h"\
LINK32 OBJS=
                                                                       "\mssql\dblib\include\sqldb.h"\
        "$(INTDIR)\getargs.obj" \
                                                                       "\mssql\dblib\include\sqlfront.h"\
        "$(INTDIR)\random.obj" \
        "$(INTDIR)\strings.obj" \
        "$(INTDIR)\time.obj" \
                                                               "$(INTDIR)\strings.obj" : $(SOURCE) $(DEP_CPP_STRIN) "$(INTDIR)"
        "$(INTDIR)\tpccldr.obj'
                                                                $(CPP) $(CPP_PROJ) $(SOURCE)
"$(OUTDIR)\tpccldr.exe": "$(OUTDIR)" $(DEF_FILE) $(LINK32_OBJS)
                                                              # End Source File
 $(LINK32) @<<
 $(LINK32 FLAGS) $(LINK32 OBJS)
                                                              # Begin Source File
!ENDIF
                                                              SOURCE=.\src\time.c
                                                              DEP_CPP_TIME_=\
".\src\tpcc.h"\
.c{$(CPP OBJS)}.obj:
 $(CPP) $(CPP_PROJ) $<
                                                                       "\mssql\dblib\include\sqldb.h"\
.cpp{$(CPP_OBJS)}.obj:
                                                                       "\mssql\dblib\include\sqlfront.h"\
 $(CPP) $(CPP PROJ) $<
.cxx{$(CPP OBJS)}.obi:
                                                               "$(INTDIR)\time.obj": $(SOURCE) $(DEP_CPP_TIME_) "$(INTDIR)"
                                                                $(CPP) $(CPP PROJ) $(SOURCE)
 $(CPP) $(CPP PROJ) $<
.c{$(CPP SBRS)}.sbr:
 $(CPP) $(CPP_PROJ) $<
                                                              # End Source File
                                                              .cpp{$(CPP_SBRS)}.sbr:
                                                              $(CPP) $(CPP_PROJ) $<
                                                              # Begin Source File
                                                              SOURCE=.\src\tpccldr.c
.cxx{$(CPP SBRS)}.sbr:
 $(CPP) $(CPP PROJ) $<
                                                              DEP CPP TPCCL=\
                                                                       ".\src\tpcc.h"\
"\mssql\dblib\include\sqldb.h"\
"\mssql\dblib\include\sqlfront.h"\
# Begin Target
                                                               "$(INTDIR)\tpccldr.obj" : $(SOURCE) $(DEP_CPP_TPCCL) "$(INTDIR)"
# Name "tpccldr - Win32 Release"
# Name "tpccldr - Win32 Debug"
                                                                $(CPP) $(CPP PROJ) $(SOURCE)
!IF "$(CFG)" == "tpccldr - Win32 Release"
                                                              # End Source File
!ELSEIF "$(CFG)" == "tpccldr - Win32 Debug"
                                                              !ENDIF
                                                              # Begin Source File
SOURCE=.\src\getargs.c
DEP CPP GETAR=\
                                                                       ".\src\tpcc.h"\
# Begin Source File
                                                                       "\mssql\dblib\include\sqldb.h"\
SOURCE=.\src\random.c
                                                                       "\mssql\dblib\include\sqlfront.h"\
DEP_CPP_RANDO=\
        ".\src\tpcc.h"\
                                                               "$(INTDIR)\getargs.obj" : $(SOURCE) $(DEP_CPP_GETAR) "$(INTDIR)"
        "\mssql\dblib\include\sqldb.h"\
        "\mssql\dblib\include\sqlfront.h"\
                                                                $(CPP) $(CPP PROJ) $(SOURCE)
```

End Source File
End Source File # End Target # End Project
End Project
######################################

Appendix C: Tunable Parameters

Microsoft SQL Server 2000

Microsoft SQL Server 2000 Startup Parameters

c:\Program Files\Microsoft SQL Server\MSSQL\Binn\sqlservr.exe -c -x -t3502 -g88

Where:

- -c Start SQL Server independent of the Service Control Manager
- -x Disable the keeping of CPU time and cache hit ratio statistics
- -t3502 Writes a message to the SQL Server Errorlog showing the beginning and ending time of each checkpoint
- -g88 Specifies the amount of memory that is set aside for allocations not from the buffer pool

Server Configuration Parameters

Microsoft Windows Server 2003 Standard Edition Configuration

The following services were set as manual on the server:

Alerter

Automatic Updates

Computer Browser

Cryptographic Services

DHCP Client

Distributed File System

Distributed Link Tracking Client

DNS Client

Help and Support

IPSEC Policy Agent

MSSQLSERVER

Print Spooler

Remote Registry Service

System Event Notification

Task Scheduler

Wireless Configuration

Configuration Parameters

1> 2> 3> 4> 5> 6> 7> 8> 9> 10> 11>

-- File: VERSION.SQL

-- Microsoft TPC-C Benchmark Kit Ver. 4.22

- Copyright Microsoft, 2001

-- Purpose: Returns SQL Server version string

print " "
select convert(char(30), getdate(),9)
print " "

Jul 7 2003 7:55:58:140AM

(1 row affected)

1> 2> 3> select @@version

Microsoft SQL Server 2000 - 8.00.760 (Intel X86) Dec 17 2002 14:22:05 Cop yright (c) 1988-2003 Microsoft Corporation Standard Edition on Windows NT 5.2 (Build 3790:) (1 row affected) 1>2> 1> 2> 3> 4> 5> 6> 7> 8> 9> 10> -- File: CONFIG.SQL Microsoft TPC-C Benchmark Kit Ver. 4.22 Copyright Microsoft, 2001 -- Purpose: Collects SQL Server configuration parameters print " " select convert(char(30), getdate(),9) print " " Jul 7 2003 7:55:59:090AM (1 row affected) 1>2>3>DBCC execution completed. If DBCC printed error messages, contact your system administrator. Configuration option 'show advanced options' changed from 1 to 1. Run the RECONFIGURE statement to install. sp_configure "show advanced",1 1>2> reconfigure with override 1>2> sp_configure name minimum maximum config value run value affinity mask -2147483648 2147483647 3 0 allow updates 1 awe enabled 0 0 c2 audit mode cost threshold for parallelism 32767 Cross DB Ownership Chaining -1 2147483647 cursor threshold default full-text language 0 2147483647 1033 default language 9999 0 fill factor (%) 100 0 index create memory (KB) 704 2147483647 lightweight pooling

locks 0	5000 2147483647 0	System Manufacturer IBM System Model eserver xSeries 235 -[867173Z]-
max degree of parallelism	0 32 0	System Type X86-based PC
0	4 2147402647	Processor x86 Family 15 Model 2 Stepping 7 GenuineIntel ~3059 Mhz
max server memory (MB) 2147483647 2147483647	4 2147483647	Processor x86 Family 15 Model 2 Stepping 7 GenuineIntel ~3059 Mhz BIOS Version/Date IBM -[GRE138AUS-1.06]-, 3/10/2003
max text repl size (B)	0 2147483647	SMBIOS Version 2.3
65536 65536		Windows Directory C:\WINDOWS
max worker threads	32 32767	System Directory C:\WINDOWS\system32
300 300 media retention	0 365 0	Boot Device \Device\HarddiskVolume4 Locale United States
0	0 303 0	Hardware Abstraction Laver Version = "5.2.3790.0
min memory per query (KB)	512 2147483647	(srv03_rtm.030324-2048)"
512 512	0. 2147492647	User NameIBMSERVER3\Administrator Time Zone Eastern Daylight Time
min server memory (MB) 0 0	0 2147483647	Total Physical Memory 2,560.00 MB
nested triggers	0 1 1	Available Physical Memory 2.20 GB
1		Total Virtual Memory 6.83 GB
network packet size (B) 1024 1024	512 65536	Available Virtual Memory 6.38 GB
1024 1024 open objects	0 2147483647 0	Page File Space 4.34 GB Page File C:\pagefile.sys
0		
priority boost	0 1 1	[Hardware Resources]
query governor cost limit	0 2147483647	
0 0	1 2147492647 1	[Canfliata/Charina]
query wait (s)	-1 2147483647 -1	(
recovery interval (min) 84 84	0 32767	Resource Device 1/O Port 0x00000000-0x000003AF PCI bus
remote access	0 1 1	I/O Port 0x00000000-0x000003AF Direct memory access controller
1		,
remote login timeout (s)	0 2147483647	I/O Port 0x000003C0-0x000003DF PCI bus
20 20 remote proc trans	0 1 0	I/O Port 0x000003C0-0x000003DF RAGE XL PCI Family (Microsoft Corporation)
0	0 1 0	Corporation)
remote query timeout (s)	0 2147483647	IRQ 5 Microsoft ACPI-Compliant System
600 600	0 1 0	IRQ 5 ServerWorks (RCC) PCI to USB Open Host Controller
scan for startup procs 0	0 1 0	I/O Port 0x00002300-0x000024FF PCI bus
set working set size	0 1 0	I/O Port 0x00002300-0x000024FF
0		Host Adapter
show advanced options	0 1 1	M A 11 O A 0000 O DEFEE DOLL
1 two digit year cutoff	1753 9999	Memory Address 0xA0000-0xBFFFF PCI bus Memory Address 0xA0000-0xBFFFF RAGE XL PCI Family (Microsoft
2049 2049	1733 7777	Corporation)
user connections	0 32767 0	
0	0 22767 0	Memory Address 0xF8000000-0xF9FFFFFPCI bus
user options 0	0 32767 0	Memory Address 0xF8000000-0xF9FFFFFMylex AcceleRAID 352 Controller (Accelerated)
		(· · · · · ··························
1>		I/O Port 0x000003B0-0x000003BB PCI bus
Microsoft Windows Server 2	0003	I/O Port 0x000003B0-0x000003BB RAGE XL PCI Family (Microsoft Corporation)
WIICIOSOIL WIIIGOWS Server 2	.003	····· /
System Information Report	for the x235	ГРМАТ

System Information Report for the x235

System Information report written at: 07/07/03 19:42:14 System Name: IBMSERVER3 [System Summary]

Item Value

OS Name Microsoft(R) Windows(R) Server 2003, Standard Edition

Version 5.2.3790 Build 3790

OS Manufacturer Microsoft Corporation

Activation Status Activation Pending (2 days remaining)

System Name IBMSERVER3

[DMA]

Resource Device Status

Channel 2 Standard floppy disk controller OK Channel 4 Direct memory access controller OK

[Forced Hardware]

Device PNP Device ID

[I/O]

Resource Device Status		IRQ 26 RAGE XL PCI Family (Microsoft Corporation) OK
0x00000000-0x000003AF	PCI bus OK	IRQ 1 Standard 101/102-Key or Microsoft Natural PS/2 Keyboard
0x00000000-0x000003AF	Direct memory access controller OK	OK
0x000003B0-0x000003BB	PCI bus OK	IRQ 12 PS/2 Compatible Mouse OK
0x000003B0-0x000003BB	RAGE XL PCI Family (Microsoft	IRQ 6 Standard floppy disk controller OK
Corporation) OK	DCI OV	IRQ 4 Communications Port (COM1) OK
0x000003BC-0x000003BF	PCI bus OK	IRQ 0 System timer OK
0x000003C0-0x000003DF	PCI bus OK	IRQ 8 System CMOS/real time clock OK
0x000003C0-0x000003DF	RAGE XL PCI Family (Microsoft	IRQ 13 Numeric data processor OK IRQ 14 Primary IDE Channel OK
Corporation) OK 0x000003E0-0x0000022FF	PCI bus OK	IRQ 14 Primary IDE Channel OK IRQ 18 Mylex AcceleRAID 352 Controller (Accelerated) OK
0x000003E0-0x000022FF 0x00002200-0x000022FF	RAGE XL PCI Family (Microsoft	IRQ 29 Broadcom NetXtreme Gigabit Ethernet OK
Corporation) OK	RAGE AL FCI Failing (wheresoft	IRQ 27 LSI Logic PCI-X Ultra320 SCSI Host Adapter OK
0x00000900-0x0000093F	Motherboard resources OK	IRQ 28 LSI Logic PCI-X Ultra320 SCSI Host Adapter OK
0x00000510-0x00000517	Motherboard resources OK Motherboard resources	IRQ 24 Mylex AcceleRAID 352 Controller (Accelerated) OK
0x00000510-0x00000517 0x00000504-0x00000507	Motherboard resources OK	IRQ 31 Mylex AcceleRAID 352 Controller (Accelerated) OK
0x00000504 0x00000507 0x00000500-0x00000503	Motherboard resources OK	in (31 Mylex received 115 332 Controller (receivated)
0x00000520-0x0000053F	Motherboard resources OK	[Memory]
0x00000420-0x00000427	Motherboard resources OK	[
0x00000460-0x00000461	Motherboard resources OK	Resource Device Status
0x0000002E-0x0000002F	Motherboard resources OK	0xA0000-0xBFFFF PCI bus OK
0x00000064-0x00000064	Standard 101/102-Key or Microsoft Natural	0xA0000-0xBFFFF RAGE XL PCI Family (Microsoft Corporation)
PS/2 Keyboard OK	.,	OK
0x00000060-0x00000060	Standard 101/102-Key or Microsoft Natural	0xFC000000-0xFFFFFFF PCI bus OK
PS/2 Keyboard OK	~·····································	0xFD000000-0xFDFFFFFF RAGE XL PCI Family (Microsoft
0x000003F0-0x000003F5	Standard floppy disk controller OK	Corporation) OK
0x000003F7-0x000003F7	Standard floppy disk controller OK	0xFEBFF000-0xFEBFFFFF RAGE XL PCI Family (Microsoft
0x00000378-0x0000037B	Printer Port (LPT1) OK	Corporation) OK
0x000003F8-0x000003FF	Communications Port (COM1) OK	0xFEC00000-0xFFFFFFFF Motherboard resources OK
0x00000020-0x00000021	Advanced programmable interrupt controller	0xFEBFE000-0xFEBFEFFF ServerWorks (RCC) PCI to USB Open Host
OK		Controller OK
0x000000A0-0x000000A1	Advanced programmable interrupt controller	0xF8000000-0xF9FFFFFF PCI bus OK
OK		0xF8000000-0xF9FFFFFF Mylex AcceleRAID 352 Controller
0x00000080-0x0000008F	Direct memory access controller OK	(Accelerated) OK
0x000000C0-0x000000DF	Direct memory access controller OK	0xFA000000-0xFBFFFFFF PCI bus OK
0x00000040-0x000000043	System timer OK	0xFBFF0000-0xFBFFFFFF Broadcom NetXtreme Gigabit Ethernet
0x00000070-0x000000073	System CMOS/real time clock OK	OK
0x00000061-0x00000061	System speaker OK	0xF6000000-0xF7FFFFFF PCI bus OK
0x000000F0-0x000000FF	Numeric data processor OK	0xF7FF0000-0xF7FFFFFF LSI Logic PCI-X Ultra320 SCSI Host
0x00000092-0x00000092	Motherboard resources OK	Adapter OK
0x000000EB-0x000000EC	Motherboard resources OK	0xF7FE0000-0xF7FEFFFF LSI Logic PCI-X Ultra320 SCSI Host
0x000001EC-0x000001EF	Motherboard resources OK	Adapter OK
0x00000400-0x0000004FE	Motherboard resources OK	0xF7FD0000-0xF7FDFFFF LSI Logic PCI-X Ultra320 SCSI Host
0x00000600-0x00000600	Motherboard resources OK	Adapter OK
0x00000800-0x0000080F	Motherboard resources OK	0xF7FC0000-0xF7FCFFFF LSI Logic PCI-X Ultra320 SCSI Host
0x00000C00-0x00000CFE	Motherboard resources OK	Adapter OK
0x00000F50-0x00000F58	Motherboard resources OK	0xF1C00000-0xF4BFFFFF PCI bus OK
0x00000700-0x0000070F	CSB5 IDE Controller OK	0xF4C00000-0xF5FFFFFF PCI bus OK
0x000001F0-0x000001F7	Primary IDE Channel OK	0xF2000000-0xF3FFFFFF Mylex AcceleRAID 352 Controller
0x000003F6-0x000003F6	Primary IDE Channel OK	(Accelerated) OK
0x00000170-0x00000177	Secondary IDE Channel OK	0xEBC00000-0xEDBFFFFF PCI bus OK
0x00000376-0x00000376	Secondary IDE Channel OK	0xEDC00000-0xF1BFFFFF PCI bus OK
0x00000A79-0x00000A79	ISAPNP Read Data Port OK	0xEE000000-0xEFFFFFFF Mylex AcceleRAID 352 Controller
0x00000279-0x00000279	ISAPNP Read Data Port OK	(Accelerated) OK
0x00000274-0x00000277	ISAPNP Read Data Port OK	
0x00002300-0x000024FF	PCI bus OK	[Components]
0x00002300-0x000024FF	LSI Logic PCI-X Ultra320 SCSI Host	
Adapter OK	TOTAL : DOLVIII 220 GOGIII 4	
0x00002400-0x000024FF	LSI Logic PCI-X Ultra320 SCSI Host	DV W P 1
Adapter OK	DCI b OV	[Multimedia]
0x00002500-0x00009FFF	PCI bus OK	
0x0000A000-0x0000FFFF	PCI bus OK	
[IRQs]		[Audio Codecs]
[11/(2)]		[Audio Codecs]
Resource Device Status		CODEC Manufacturer Description Status File
IRQ 5 Microsoft ACPI-Co	ompliant System OK	Version Size Creation Date
) PCI to USB Open Host Controller OK	
(,	

Microsoft Corporation c:\windows\system32\msaud32.acm Drive D: Windows Media Audio Codec OK Description CD-ROM Drive C:\WINDOWS\system32\MSAUD32.ACM 8.00.00.4487 288.00 KB Media Loaded No CD-ROM (294,912 bytes) 3/25/2003 7:00 AM Media Type c:\windows\system32\sl anet.acm Sipro Lab Telecom Inc. LG CD-ROM CRD-8484B Name Sipro Lab Telecom Audio Codec OK Manufacturer (Standard CD-ROM drives) C:\WINDOWS\system32\SL ANET.ACM 3.02 84.00 KB (86,016 Status OK 3/25/2003 7:00 AM bytes) Transfer Rate Not Available c:\windows\system32\l3codeca.acm Fraunhofer Institut Integrierte SCSI Target ID Fraunhofer IIS MPEG Layer-3 Codec PNP Device ID Schaltungen IIS OK C:\WINDOWS\system32\L3CODECA.ACM 1, 9, 0, 0305 IDE\CDROMLG CD-ROM CRD-8484B 2.01 \5 284.00 KB (290,816 bytes) 3/25/2003 7:00 AM &FB0C83D&0&0.0.0 c:\windows\system32\msadp32.acm Microsoft Corporation c:\windows\system32\drivers\cdrom.sys (5.2.3790.0 C:\WINDOWS\system32\MSADP32.ACM 5.2.3790.0 (srv03_rtm.030324-2048), 49.50 KB (50,688 bytes), 3/25/2003 7:00 AM) OK (srv03 rtm.030324-2048) 14.50 KB (14,848 bytes) 3/25/2003 7:00 AM [Sound Device] DSP GROUP, INC. c:\windows\system32\tssoft32.acm C:\WINDOWS\system32\TSSOFT32.ACM 1.01 OK 9.50 KB Item Value 3/25/2003 7:00 AM (9,728 bytes) c:\windows\system32\msg711.acm Microsoft Corporation [Display] C:\WINDOWS\system32\MSG711.ACM 5.2.3790.0 OK (srv03 rtm.030324-2048) 10.00 KB (10,240 bytes) 3/25/2003 Value Item 7:00 AM Name RAGE XL PCI Family (Microsoft Corporation) c:\windows\system32\imaadp32.acm Microsoft Corporation PNP Device ID OK C:\WINDOWS\system32\IMAADP32.ACM5.2.3790.0 PCI\VEN 1002&DEV 4752&SUBSYS 02401014&REV 27\3&267A616A& (srv03 rtm.030324-2048) 15.50 KB (15,872 bytes) 3/25/2003 0&48 7:00 AM Adapter Type ATI RAGE XL PCI (B41), ATI Technologies Inc. $c:\windows\system 32\msg723.acm$ compatible Microsoft Corporation Adapter Description RAGE XL PCI Family (Microsoft Corporation) Adapter RAM 6/9/2003 12:45 PM 8.00 MB (8,388,608 bytes) (118,784 bytes) c:\windows\system32\msgsm32.acm Microsoft Corporation Installed Drivers ati2drad.dll C:\WINDOWS\system32\MSGSM32.ACM 5.2.3790.0 Driver Version 5.10.3663.6013 (srv03 rtm.030324-2048) 20.50 KB (20,992 bytes) 3/25/2003 INF File atiixpad.inf (ati2mpad section) 7:00 AM Color Planes Color Table Entries 4294967296 [Video Codecs] Resolution 800 x 600 x 60 hertz Bits/Pixel 32 CODEC Manufacturer Memory Address 0xFD000000-0xFDFFFFFF Description Status File I/O Port 0x00002200-0x000022FF Version Size Creation Date c:\windows\system32\msh261.dry Microsoft Corporation Memory Address 0xFEBFF000-0xFEBFFFFF C:\WINDOWS\system32\MSH261.DRV 4.4.4000 180.00 KB IRO Channel **IRO 26** 6/9/2003 12:45 PM 0x000003B0-0x000003BB (184,320 bytes) I/O Port c:\windows\system32\msrle32.dllMicrosoft Corporation I/O Port 0x000003C0-0x000003DF C:\WINDOWS\system32\MSRLE32.DLL 5.2.3790.0 OK 0xA0000-0xBFFFF Memory Address (srv03 rtm.030324-2048) 10.50 KB (10,752 bytes) 3/25/2003 c:\windows\system32\drivers\ati2mpad.sys (5.10.3663.6013, 335.38 KB (343,424 bytes), 6/9/2003 8:29 AM) 7:00 AM c:\windows\system32\ivuv 32.dll Microsoft Corporation C:\WINDOWS\system32\IYUV_32.DLL 5.2.3790.0 OK [Infrared] (srv03 rtm.030324-2048) 45.00 KB (46,080 bytes) 3/24/2003 8:49 PM Value Item c:\windows\system32\msyuv.dll Microsoft Corporation C:\WINDOWS\system32\MSYUV.DLL [Input] (srv03_rtm.030324-2048) 3/24/2003 16.50 KB (16,896 bytes) 8:49 PM c:\windows\system32\msvidc32.dll Microsoft Corporation C:\WINDOWS\system32\MSVIDC32.DLL 5.2.3790.0 [Keyboard] (srv03 rtm.030324-2048) 26.50 KB (27,136 bytes) 3/25/2003 Value 7:00 AM Item c:\windows\system32\tsbyuv.dll Microsoft Corporation Description Standard 101/102-Key or Microsoft Natural PS/2 C:\WINDOWS\system32\TSBYUV.DLL 5.2.3790.0 Keyboard (srv03 rtm.030324-2048) 8.00 KB (8,192 bytes) 3/24/2003 8:50 PM Name Enhanced (101- or 102-key) c:\windows\system32\msh263.drv Microsoft Corporation Layout 00000409 C:\WINDOWS\system32\MSH263.DRV 4.4.4000 284.00 KB PNP Device ID ACPI\PNP0303\4&35118DFF&0 3/24/2003 8:46 PM (290,816 bytes) Number of Function Keys IRO Channel IRO 1 I/O Port [CD-ROM] 0x00000064-0x00000064 0x00000060-0x00000060 I/O Port Item Value

c:\windows\system32\drivers\i8042prt.sys (5.2.3790.0 [00000003] WAN Miniport (L2TP) Name (srv03 rtm.030324-2048), 68.50 KB (70,144 bytes), 3/25/2003 7:00 AM) Adapter Type Not Available Product Type WAN Miniport (L2TP) [Pointing Device] Installed Yes PNP Device ID ROOT\MS L2TPMINIPORT\0000 Value Last Reset 7/7/2003 2:51 PM Item Hardware Type PS/2 Compatible Mouse Index 3 Number of Buttons Service Name Rasl2tp Status OKIP Address Not Available ACPI\PNP0F13\4&35118DFF&0 PNP Device ID IP Subnet Not Available Power Management Supported No Default IP Gateway Not Available Double Click Threshold DHCP Enabled Nο Handedness Right Handed Operation DHCP Server Not Available IRQ Channel DHCP Lease Expires Not Available IRO 12 c:\windows\system32\drivers\i8042prt.sys (5.2.3790.0 DHCP Lease Obtained Driver Not Available (srv03 rtm.030324-2048), 68.50 KB (70,144 bytes), 3/25/2003 7:00 AM) MAC Address Not Available Driver c:\windows\system32\drivers\rasl2tp.sys (5.2.3790.0 (srv03_rtm.030324-2048), 77.00 KB (78,848 bytes), 3/25/2003 7:00 AM) [Modem] Item Value [00000004] WAN Miniport (PPTP) Adapter Type Wide Area Network (WAN) Product Type WAN Miniport (PPTP) [Network] Installed Yes ROOT\MS PPTPMINIPORT\0000 PNP Device ID Last Reset 7/7/2003 2:51 PM Index [Adapter] Service Name **PptpMiniport** Value IP Address Not Available Item Name [00000001] Broadcom NetXtreme Gigabit Ethernet IP Subnet Not Available Ethernet 802.3 Default IP Gateway Adapter Type Not Available Product Type Broadcom NetXtreme Gigabit Ethernet **DHCP** Enabled Installed Yes DHCP Server Not Available PNP Device ID DHCP Lease Expires Not Available PCI\VEN 14E4&DEV 16A7&SUBSYS 026F1014&REV 02\3&13C0B0C5 DHCP Lease Obtained Not Available MAC Address 50:50:54:50:30:30 &0&40 Last Reset 7/7/2003 2:51 PM c:\windows\system32\drivers\raspptp.sys (5.2.3790.0 Driver (srv03_rtm.030324-2048), 70.50 KB (72,192 bytes), 3/25/2003 7:00 AM) Index Service Name b57w2k IP Address 192.168.132.200 [00000005] WAN Miniport (PPPOE) Name IP Subnet 255.255.255.0 Adapter Type Wide Area Network (WAN) Default IP Gateway Not Available Product Type WAN Miniport (PPPOE) DHCP Enabled Installed Yes No **DHCP Server** PNP Device ID ROOT\MS PPPOEMINIPORT\0000 Not Available DHCP Lease Expires Not Available Last Reset 7/7/2003 2:51 PM DHCP Lease Obtained Not Available Index 5 MAC Address 00:02:55:07:8A:8A Service Name RasPppoe Memory Address 0xFBFF0000-0xFBFFFFFF IP Address Not Available IRO Channel **IRO 29** IP Subnet Not Available Default IP Gateway Driver c:\windows\system32\drivers\b57xp32.sys (2.91.0.0 built by: Not Available WinDDK, 137.00 KB (140,288 bytes), 6/9/2003 8:29 AM) DHCP Enabled No DHCP Server Not Available [00000002] RAS Async Adapter DHCP Lease Expires Not Available Name Adapter Type DHCP Lease Obtained Not Available Not Available Product Type RAS Async Adapter MAC Address 33:50:6F:45:30:30 Installed Yes Driver c:\windows\system32\drivers\raspppoe.sys (5.2.3790.0 PNP Device ID Not Available (srv03_rtm.030324-2048), 38.00 KB (38,912 bytes), 3/25/2003 7:00 AM) Last Reset 7/7/2003 2:51 PM [00000006] Direct Parallel Index Name Adapter Type Not Available Service Name AsyncMac IP Address Not Available Product Type Direct Parallel IP Subnet Not Available Installed Yes Not Available PNP Device ID ROOT\MS PTIMINIPORT\0000 Default IP Gateway **DHCP** Enabled Last Reset 7/7/2003 2:51 PM No **DHCP Server** Not Available Index DHCP Lease Expires Not Available Service Name Raspti DHCP Lease Obtained Not Available IP Address Not Available MAC Address Not Available IP Subnet Not Available Default IP Gateway Not Available

DHCP Enabled Guarantees Sequencing No DHCP Server Not Available Maximum Address Size 16 bytes DHCP Lease Expires Not Available Maximum Message Size 63.93 KB (65,467 bytes) DHCP Lease Obtained Not Available Message Oriented MAC Address Not Available Minimum Address Size 16 bytes c:\windows\system32\drivers\raspti.sys (5.2.3790.0 Pseudo Stream Oriented Driver No (srv03 rtm.030324-2048), 18.50 KB (18,944 bytes), 3/25/2003 7:00 AM) Supports Broadcasting Yes Supports Connect Data No Name [00000007] WAN Miniport (IP) Supports Disconnect Data No Adapter Type Supports Encryption No Not Available WAN Miniport (IP) Product Type Supports Expedited Data No Installed Yes Supports Graceful Closing No PNP Device ID ROOT\MS NDISWANIP\0000 Supports Guaranteed Bandwidth No Last Reset 7/7/2003 2:51 PM Supports Multicasting Yes Index RSVP UDP Service Provider Service Name NdisWan Name IP Address Not Available Connectionless Service IP Subnet Not Available Guarantees Delivery No Default IP Gateway Not Available Guarantees Sequencing No **DHCP** Enabled No Maximum Address Size 16 bytes DHCP Server 63.93 KB (65,467 bytes) Not Available Maximum Message Size DHCP Lease Expires Not Available Message Oriented DHCP Lease Obtained Not Available Minimum Address Size 16 bytes MAC Address Not Available Pseudo Stream Oriented No Driver c:\windows\system32\drivers\ndiswan.sys (5.2.3790.0 Supports Broadcasting Yes (srv03 rtm.030324-2048), 96.50 KB (98,816 bytes), 3/25/2003 7:00 AM) Supports Connect Data Nο Supports Disconnect Data No [00000008] IBM Netfinity 10/100 Ethernet Adapter Supports Encryption Yes Name Adapter Type Not Available Supports Expedited Data No Product Type IBM Netfinity 10/100 Ethernet Adapter Supports Graceful Closing No Installed Yes Supports Guaranteed Bandwidth No PNP Device ID Not Available Supports Multicasting Yes Last Reset 7/7/2003 2:51 PM Index 8 Name RSVP TCP Service Provider Service Name E100B Connectionless Service No IP Address Not Available Guarantees Delivery Yes IP Subnet Not Available Guarantees Sequencing Yes Default IP Gateway Not Available Maximum Address Size 16 bytes **DHCP** Enabled Maximum Message Size Nο 0 bytes DHCP Server Not Available Message Oriented DHCP Lease Expires Not Available Minimum Address Size 16 bytes Not Available DHCP Lease Obtained Pseudo Stream Oriented No MAC Address Supports Broadcasting No Not Available Supports Connect Data No Supports Disconnect Data [Protocol] No Supports Encryption Yes Item Value Supports Expedited Data Yes Name MSAFD Tepip [TCP/IP] Supports Graceful Closing Yes Supports Guaranteed Bandwidth No Connectionless Service Guarantees Delivery Yes Supports Multicasting No **Guarantees Sequencing** Yes Maximum Address Size MSAFD NetBIOS 16 bytes Name Maximum Message Size [\Device\NetBT_Tcpip_{7A67E8EE-C587-4A6C-8DAC-869CD5CB7314}] 0 bytes Message Oriented SEOPACKET 3 Minimum Address Size 16 bytes Connectionless Service No Pseudo Stream Oriented Guarantees Delivery Yes No Supports Broadcasting Guarantees Sequencing No Yes Supports Connect Data No Maximum Address Size 20 bytes Supports Disconnect Data 62.50 KB (64,000 bytes) No Maximum Message Size Supports Encryption No Message Oriented Yes Supports Expedited Data Yes Minimum Address Size 20 bytes Supports Graceful Closing Yes Pseudo Stream Oriented No Supports Guaranteed Bandwidth No Supports Broadcasting No Supports Multicasting No Supports Connect Data No Supports Disconnect Data No MSAFD Topip [UDP/IP] Supports Encryption No Supports Expedited Data Connectionless Service Yes No Guarantees Delivery No Supports Graceful Closing No

Supports Guaranteed Bandwidth No Guarantees Sequencing Yes Supports Multicasting No Maximum Address Size 20 bytes Maximum Message Size 62.50 KB (64,000 bytes) MSAFD NetBIOS Name Message Oriented $[\Device\NetBT_Tcpip_\{7A67E8EE-C587-4A6C-8DAC-869CD5CB7314\}]$ Minimum Address Size 20 bytes DATAGRAM 3 Pseudo Stream Oriented No Connectionless Service Yes Supports Broadcasting No Guarantees Delivery No Supports Connect Data No Guarantees Sequencing No Supports Disconnect Data No Supports Encryption No Maximum Address Size 20 bytes 62.50 KB (64,000 bytes) Maximum Message Size Supports Expedited Data No Message Oriented Supports Graceful Closing No Minimum Address Size 20 bytes Supports Guaranteed Bandwidth No Pseudo Stream Oriented Supports Multicasting No No Supports Broadcasting Yes MSAFD NetBIOS Supports Connect Data No Name Supports Disconnect Data No [\Device\NetBT_Tcpip_{22FFBB96-7A17-4B97-B00C-04AF0657D964}] Supports Encryption No DATAGRAM 1 Supports Expedited Data Connectionless Service Nο Yes Supports Graceful Closing Guarantees Delivery No No Supports Guaranteed Bandwidth No Guarantees Sequencing No Supports Multicasting No Maximum Address Size 20 bytes Maximum Message Size 62.50 KB (64,000 bytes) MSAFD NetBIOS Message Oriented Name [\Device\NetBT_Tcpip_{88B09A25-8F51-4EAC-A3E3-425FA99266F7}] Minimum Address Size 20 bytes SEOPACKET 0 Pseudo Stream Oriented No Connectionless Service Supports Broadcasting Yes No Guarantees Delivery Yes Supports Connect Data No Guarantees Sequencing Supports Disconnect Data No Maximum Address Size Supports Encryption No 20 bytes Supports Expedited Data Maximum Message Size 62.50 KB (64,000 bytes) No Supports Graceful Closing Message Oriented No Minimum Address Size 20 bytes Supports Guaranteed Bandwidth No Pseudo Stream Oriented Supports Multicasting No No Supports Broadcasting No Supports Connect Data MSAFD NetBIOS No Name Supports Disconnect Data $[\Device\NetBT_Tcpip_\{A0109638\text{-}F722\text{-}4EEF\text{-}96BC\text{-}58A66E11CAFF}\}]$ No Supports Encryption No SEQPACKET 2 Supports Expedited Data No Connectionless Service No Supports Graceful Closing Guarantees Delivery Yes No Supports Guaranteed Bandwidth No Guarantees Sequencing Yes Supports Multicasting No Maximum Address Size 20 bytes Maximum Message Size 62.50 KB (64,000 bytes) Message Oriented MSAFD NetBIOS Name [\Device\NetBT Tcpip {88B09A25-8F51-4EAC-A3E3-425FA99266F7}] Minimum Address Size 20 bytes DATAGRAM 0 Pseudo Stream Oriented No Connectionless Service Yes Supports Broadcasting No Guarantees Delivery No Supports Connect Data No Supports Disconnect Data Guarantees Sequencing No No Maximum Address Size 20 bytes Supports Encryption No 62.50 KB (64,000 bytes) Supports Expedited Data Maximum Message Size No Message Oriented Supports Graceful Closing No Supports Guaranteed Bandwidth No Minimum Address Size 20 bytes Pseudo Stream Oriented Supports Multicasting No No Supports Broadcasting Yes Supports Connect Data No Name MSAFD NetBIOS Supports Disconnect Data [\Device\NetBT Tcpip {A0109638-F722-4EEF-96BC-58A66E11CAFF}] No DATAGRAM 2 Supports Encryption No Supports Expedited Data No Connectionless Service Yes Supports Graceful Closing No Guarantees Delivery No Supports Guaranteed Bandwidth No Guarantees Sequencing No Supports Multicasting No Maximum Address Size 20 bytes Maximum Message Size 62.50 KB (64,000 bytes) MSAFD NetBIOS Message Oriented Minimum Address Size [\Device\NetBT_Tcpip_{22FFBB96-7A17-4B97-B00C-04AF0657D964}] 20 bytes SEQPACKET 1 Pseudo Stream Oriented No Connectionless Service No Supports Broadcasting Yes Guarantees Delivery Yes Supports Connect Data No

Supports Disconnect Data No c:\windows\system32\drivers\serial.sys (5.2.3790.0 Supports Encryption No (srv03 rtm.030324-2048), 76.00 KB (77,824 bytes), 3/25/2003 7:00 AM) Supports Expedited Data No Supports Graceful Closing No [Parallel] Supports Guaranteed Bandwidth No Supports Multicasting No Item Value LPT1 Name [WinSock] PNP Device ID ACPI\PNP0400\1 I/O Port 0x00000378-0x0000037B Item Value Driver c:\windows\system32\drivers\parport.sys (5.2.3790.0 File c:\windows\system32\winsock.dll (srv03_rtm.030324-2048), 76.50 KB (78,336 bytes), 3/24/2003 6:04 PM) 2.80 KB (2,864 bytes) Size Version 3.10 [Storage] c:\windows\system32\wsock32.dll File 22.00 KB (22,528 bytes) Size 5.2.3790.0 (srv03 rtm.030324-2048) Version [Drives] Value [Ports] Item Drive Description 3 1/2 Inch Floppy Drive [Serial] Drive Description Local Fixed Disk Item Value Compressed No Communications Port (COM1) NTFS File System Name Status OK Size 16.94 GB (18,186,059,776 bytes) PNP Device ID ACPI\PNP0501\1 Free Space 13.31 GB (14,296,236,032 bytes) Maximum Input Buffer Size Volume Name Maximum Output Buffer Size Volume Serial Number No 885F2549 Settable Baud Rate Yes Settable Data Bits Yes Drive D: Settable Flow Control Yes Description CD-ROM Disc Settable Parity Settable Parity Check Yes Drive E٠ Settable Stop Bits Description Local Fixed Disk Settable RLSD Yes Not Available Compressed Supports RLSD Not Available Yes File System Supports 16 Bit Mode No Not Available Size Supports Special Characters Free Space Not Available Nο Baud Rate 9600 Volume Name Not Available Bits/Byte 8 Volume Serial Number Not Available Stop Bits 1 Parity F٠ None Drive Description Local Fixed Disk Busy No Abort Read/Write on Error Compressed Not Available No Binary Mode Enabled Yes File System Not Available Continue XMit on XOff No Size Not Available CTS Outflow Control No Free Space Not Available Discard NULL Bytes No Volume Name Not Available DSR Outflow Control 0 Volume Serial Number Not Available DSR Sensitivity DTR Flow Control Type Enable Drive GEOF Character Description Local Fixed Disk Error Replace Character Compressed Not Available Error Replacement Enabled Not Available No File System **Event Character** Not Available Parity Check Enabled No Free Space Not Available RTS Flow Control Type Volume Name Enable Not Available XOff Character Volume Serial Number Not Available XOffXMit Threshold 512 XOn Character Drive H: XOnXMit Threshold 2048 Description Local Fixed Disk XOnXOff InFlow Control 0 Compressed Not Available XOnXOff OutFlow Control Not Available 0 File System IRQ Channel IRQ4 Size Not Available I/O Port 0x000003F8-0x000003FF Free Space Not Available Volume Name Not Available

Total Sectors 35,535,780 Volume Serial Number Not Available Total Tracks 564,060 Tracks/Cylinder Drive I: 255 Local Fixed Disk Partition Disk #0, Partition #0 Description Compressed Not Available Partition Size 16.94 GB (18,186,061,824 bytes) File System Not Available Partition Starting Offset 32,256 bytes Size Not Available Description Free Space Not Available Disk drive Volume Name Not Available Manufacturer (Standard disk drives) LSILOGIC 1030 IM SCSI Disk Device Volume Serial Number Not Available Model Bytes/Sector 512 Drive Media Loaded Yes J. Local Fixed Disk Description Media Type Fixed hard disk Not Available Partitions 1 Compressed Not Available SCSI Bus 0 File System Size Not Available SCSI Logical Unit 0 Free Space Not Available SCSI Port 2 SCSI Target ID Volume Name Not Available Volume Serial Number Not Available Sectors/Track 63 68.24 GB (73,270,794,240 bytes) Size **Total Cylinders** Drive $K \cdot$ 8,908 Description Local Fixed Disk **Total Sectors** 143,107,020 Not Available Compressed Total Tracks 2,271,540 File System Not Available Tracks/Cylinder 255 Partition Disk #1, Partition #0 Size Not Available Free Space Not Available Partition Size 44.00 GB (47,245,976,064 bytes) Volume Name Not Available Partition Starting Offset 8,257,536 bytes Volume Serial Number Not Available Description Mylex RAID Disk Device Manufacturer Drive X. Mylex Description Network Connection Model MYLEX AcceleRAID 352 SCSI Disk Device \\192.168.132.253\e\$ Provider Name Bytes/Sector 512 Media Loaded Yes Fixed hard disk Drive Y: Media Type Description Local Fixed Disk Partitions 3 Compressed SCSI Bus 2 No File System NTFS SCSI Logical Unit 0 78.13 GB (83,889,594,368 bytes) SCSI Port 5 Size Free Space 21.72 GB (23,324,512,256 bytes) SCSI Target ID 0 Volume Name Backup_cool1 Sectors/Track 63 Volume Serial Number A8FC56D7 Size 237.01 GB (254,490,163,200 bytes) Total Cylinders 30,940 Drive **Total Sectors** 497,051,100 7,889,700 Description Local Fixed Disk Total Tracks Compressed Tracks/Cylinder No 255 NTFS File System Partition Disk #3, Partition #0 Size 78.13 GB (83,889,594,368 bytes) Partition Size 28.54 GB (30,647,361,024 bytes) Free Space 21.73 GB (23,332,900,864 bytes) Partition Starting Offset 8,257,536 bytes Volume Name Backup_cool2 Partition Disk #3, Partition #1 Volume Serial Number 240D2182 Partition Size 13.73 GB (14,739,669,504 bytes) Partition Starting Offset 30,655,650,816 bytes Partition Disk #3, Partition #2 [Disks] 78.13 GB (83,889,598,464 bytes) Partition Size Item Value Partition Starting Offset 45,395,352,576 bytes Disk drive Description Manufacturer (Standard disk drives) Description Mylex RAID Disk Device Model Manufacturer Mylex Model MYLEX AcceleRAID 352 SCSI Disk Device Bytes/Sector 512 Media Loaded Bytes/Sector Yes 512 Media Type Fixed hard disk Media Loaded Yes Partitions 1 Media Type Fixed hard disk SCSI Bus 0 Partitions 3 SCSI Logical Unit 0 SCSI Bus 2 SCSI Logical Unit SCSI Port 2 0 SCSI Target ID SCSI Port 4 0 Sectors/Track 63 SCSI Target ID 16.94 GB (18,194,319,360 bytes) Sectors/Track Size **Total Cylinders** Size 237.01 GB (254,490,163,200 bytes)

Total Cylinders 30,940 PNP Device ID PCI\VEN 1000&DEV 0030&SUBSYS 026C1014&REV 07\3&1070020&0 Total Sectors 497,051,100 Total Tracks 7,889,700 I/O Port 0x00002400-0x000024FF Tracks/Cylinder 255 Partition Disk #2, Partition #0 Memory Address 0xF7FD00000-0xF7FDFFFF Partition Size 28.54 GB (30,647,361,024 bytes) Memory Address 0xF7FC0000-0xF7FCFFFF IRQ Channel Partition Starting Offset 8,257,536 bytes IRQ 28 Partition Disk #2, Partition #1 c:\windows\system32\drivers\symmpi.sys (1.08.18.00 Driver Partition Size 13.73 GB (14,739,669,504 bytes) (NT.021001-2000), 25.88 KB (26,496 bytes), 3/25/2003 7:00 AM) Partition Starting Offset 30,655,650,816 bytes Partition Disk #2, Partition #2 Name Mylex AcceleRAID 352 Controller (Accelerated) Partition Size 78.13 GB (83,889,598,464 bytes) Manufacturer Mylex OK Partition Starting Offset 45,395,352,576 bytes Status PNP Device ID PCI\VEN 1069&DEV 0050&SUBSYS 00501069&REV 02\3&29E81982&0 Description Mylex RAID Disk Device Manufacturer Mylex &28 MYLEX AcceleRAID 352 SCSI Disk Device Model Memory Address 0xF2000000-0xF3FFFFFF IRQ Channel Bytes/Sector 512 IRQ 24 Media Loaded Driver c:\windows\system32\drivers\dac2w2k.sys (7.05-01 built by: Yes Media Type Fixed hard disk WinDDK, 173.00 KB (177,152 bytes), 5/1/2003 5:13 PM) Partitions 2 SCSI Bus 2 Mylex AcceleRAID 352 Controller (Accelerated) Name SCSI Logical Unit 0 Manufacturer Mylex SCSI Port 6 Status SCSI Target ID 0 PNP Device ID Sectors/Track PCI\VEN_1069&DEV_0050&SUBSYS_00501069&REV_02\3&172E68DD& 63 237.01 GB (254,490,163,200 bytes) 0&30 Size 0xEE000000-0xEFFFFFF Total Cylinders 30,940 Memory Address **Total Sectors** 497,051,100 IRQ Channel IRQ 31 Total Tracks 7,889,700 c:\windows\system32\drivers\dac2w2k.sys (7.05-01 built by: Driver Tracks/Cylinder 255 WinDDK, 173.00 KB (177,152 bytes), 5/1/2003 5:13 PM) Partition Disk #4, Partition #0 Partition Size 28.54 GB (30,647,361,024 bytes) [IDE] Partition Starting Offset 8,257,536 bytes Partition Disk #4, Partition #1 Item Value 13.73 GB (14,739,669,504 bytes) CSB5 IDE Controller Partition Size Name Partition Starting Offset 30,655,650,816 bytes Manufacturer ServerWorks Status [SCSI] PNP Device ID PCI\VEN_1166&DEV_0212&SUBSYS_02121166&REV_93\3&267A616A& Value 0&79 Item 0x00000700-0x0000070F Name Mylex AcceleRAID 352 Controller (Accelerated) I/O Port c:\windows\system32\drivers\pciide.sys (5.2.3790.0 Manufacturer Mylex Driver (srv03_rtm.030324-2048), 5.50 KB (5,632 bytes), 3/25/2003 7:00 AM) Status OK PNP Device ID PCI\VEN_1069&DEV_0050&SUBSYS_00501069&REV_02\3&13C0B0C5& Primary IDE Channel Name Manufacturer (Standard IDE ATA/ATAPI controllers) Memory Address 0xF8000000-0xF9FFFFF Status OK IRQ Channel IRQ 18 PNP Device ID PCIIDE\IDECHANNEL\4&1024D5C6&0&0 0x000001F0-0x000001F7 c:\windows\system32\drivers\dac2w2k.sys (7.05-01 built by: I/O Port Driver 0x000003F6-0x000003F6 WinDDK, 173.00 KB (177,152 bytes), 5/1/2003 5:13 PM) I/O Port IRQ 14 IRQ Channel LSI Logic PCI-X Ultra320 SCSI Host Adapter c:\windows\system32\drivers\atapi.sys (5.2.3790.0 Name Driver Manufacturer LSI Logic Inc. (srv03 rtm.030324-2048), 89.00 KB (91,136 bytes), 3/25/2003 7:00 AM) Status OK PNP Device ID Secondary IDE Channel Name PCI\VEN 1000&DEV 0030&SUBSYS 026C1014&REV 07\3&1070020&0 Manufacturer (Standard IDE ATA/ATAPI controllers) OK &38 Status I/O Port 0x00002300-0x000024FF PNP Device ID PCIIDE\IDECHANNEL\4&1024D5C6&0&1 0xF7FF0000-0xF7FFFFFFI/O Port 0x00000170-0x00000177 Memory Address Memory Address 0xF7FE0000-0xF7FEFFFF I/O Port 0x00000376-0x00000376 IRQ Channel IRQ 27 c:\windows\system32\drivers\atapi.sys (5.2.3790.0 Driver c:\windows\system32\drivers\symmpi.sys (1.08.18.00 (srv03_rtm.030324-2048), 89.00 KB (91,136 bytes), 3/25/2003 7:00 AM)

© IBM Corporation TPC Benchmark C Full Disclosure Report - July 2003

[Printing]

Driver

Name

Port Name Server Name

(NT.021001-2000), 25.88 KB (26,496 bytes), 3/25/2003 7:00 AM)

LSI Logic PCI-X Ultra320 SCSI Host Adapter

LSI Logic Inc.

Name

Status

Manufacturer

OK

[Problem I	Devices						beep	Веер	c:\window	s\system32	\drivers\bee	en.svs	Kernel
-	PNP Devi	aa ID	Error Cod				Driver Yes	Yes		Running		Normal	No
Device Not Availa for this dev		ACPI\IBN		DABA3FF&	0	The drivers	cbidf2k Driver No	cbidf2k No	c:\window Disabled	s\system32 Stopped	\drivers\cbi OK	df2k.sys Normal	Kernel No
[USB]							cd20xrnt	cd20xrnt	Not Availa		Kernel Dri		No
Device ServerWoo	PNP Device		3 Open Host	Controller			Disabled cdfs Driver	Stopped Cdfs Yes		Normal s\system32 Running		No s.sys Normal	File System No
	. ,				EV_05\3&	267A616A&	Yes	CD-ROM		Ü		\drivers\cdr	
USB Root	Hub	USB\RO0	OT_HUB\48	&AF5358C&	2 0		Kernel Dri		Yes	System	Running		Normal
[Software	Environme	nt]					changer System		Not Availa OK	ible Ignore	Kernel Dri No	iver No	No
							clusdisk Kernel Dri	Cluster Di	sk Driver No	_	s\system32		sdisk.sys Normal
[System D	rivers]						No	No			**		
Name	Descriptio		File	J 1	Started	Start Mode	cmdide Disabled	CmdIde Stopped	Not Availa OK	Normal	Kernel Dri No	No	No
State abiosdsk	Status Abiosdsk	Error Cor Not Avail		Accept Pau Kernel Driv		Accept Stop No	cpqarray Disabled	Cpqarray Stopped	Not Availa OK	ıble Normal	Kernel Dri No	iver No	No
	Stopped		Ignore		No	110	cpqarry2		Not Availa		Kernel Dri		No
acpi	Microsoft			v 15:		**	Disabled	Stopped	OK	Normal	No	No	
c:\window Boot	s\system32 Running	OK	pı.sys Normal	Kernel Driv No	ver Yes	Yes	Disabled	Stopped	Not Availa OK	Normal	Kernel Dri No	No No	No
acpiec	ACPIEC			\drivers\acpi		Kernel			Not Availa		Kernel Dri		No
Driver	No	Disabled	Stopped	OK	Normal	No			OK D	Normal	No	No	
No admu160m	adpu160m	Not Avail	able	Kernel Driv	ver	No	credisk c·\window		Filter Driv drivers\crc		Kernel Dri	iver	Yes
-	Stopped	OK	Normal		No	110	Boot	Running	OK	Normal	No	Yes	100
adpu320	adpu320	Not Avail		Kernel Driv		No	dac2w2k		c:\window	-		-	Kernel
Disabled afent	Stopped afent	OK Not Avail	Normal able	No Kernel Driv	No	No	Driver Yes	Yes	Boot	Running	OK	Normal	No
	Stopped	OK	Normal		No	110		dac960nt	Not Availa	ıble	Kernel Dri	iver	No
afd			pport Enviro			**	Disabled	Stopped	OK	Normal	No	No	
c:\window Auto	s\system32 Running	\drivers\afe OK	d.sys Normal	Kernel Driv No	ver Yes	Yes	dellcerc Disabled	dellcerc Stopped	Not Availa OK	ible Normal	Kernel Dri No	No	No
	Aha154x			Kernel Driv		No			c:\window				File System
Disabled	Stopped	OK	Normal		No		Driver	Yes	Boot	Running	OK	Normal	No
aic78u2 Disabled	aic78u2 Stopped	Not Avail OK	able Normal	Kernel Driv No	ver No	No	Yes disk	Disk Drive	or	c:\window	c\cvctem32	\drivers\dis	k eve
aic78xx	aic78xx	Not Avail		Kernel Driv		No	Kernel Dri		Yes	Boot	Running		Normal
Disabled	Stopped	OK	Normal		No		No	Yes					
aliide Disabled	AliIde Stopped	Not Avail OK	able Normal	Kernel Driv No	ver No	No	dmboot Driver	dmboot No		s\system32 Stopped			Kernel No
			Media Drive		110		No	110	Disabled	Stopped	OK	rvormar	110
	-		,	Kernel Driv		No	dmio	U	sk Manage				
Manual atapi	Stopped Standard I		Normal Hard Disk C		No		c:\window Boot	s\system32 Running	\drivers\dm OK	io.sys Normal	Kernel Dri No	ver Yes	Yes
	s\system32			Kernel Driv	ver	Yes	dmload	dmload		s\system32			Kernel
Boot	Running	OK	Normal		Yes		Driver	Yes	Boot	Running		Normal	No
atdisk Disabled	Atdisk Stopped	Not Avail OK	able Ignore	Kernel Driv No	ver No	No	Yes dpti2o	dpti2o	Not Availa	hla	Kernel Dri	ivor	No
				\drivers\ati2		Kernel	Disabled	Stopped	OK	Normal	No	No	NO
Driver	Yes	Manual	Running	OK	Ignore	No	e100b	Intel(R) Pl	RO Adaptei				
Yes atmarpc	ATM ARE	Client Dr	otogol				c:\window Manual	s\system32 Stopped	\drivers\e10 OK	00b325.sys Normal	Kernel Dri	iver No	No
	s\system32			Kernel Driv	ver	No	fastfat	Fastfat		s\system32			File System
Manual	Stopped	OK	Normal	No	No		Driver	No	Disabled	-	OK	Normal	No
audstub Kernel Dri	Audio Stu	b Driver Yes	c:\window Manual	s\system32\ Running		dstub.sys Normal	No fdc	Flonny Di-	sk Controll	or Driver			
No	Yes	103	ivianuai	Running	OK.	roinai			\drivers\fdc		Kernel Dri	iver	Yes
b57w2k			e Gigabit Et				Manual	Running	OK	Normal	No	Yes	
c:\window Manual	s\system32 Running		7xp32.sys Normal	Kernel Driv No	ver Yes	Yes	fips Driver	Fips Yes	c:\window System	s\system32		s.sys Normal	Kernel No
iviaiiual	Ruillillig	OIX.	romai	110	103		Yes	103	System	Running	OK.	romai	110

flpydisk Kernel Dr		sk Driver Yes	c:\window Manual	vs\system32 Running		ydisk.sys Normal
No ftdisk	Yes Volume N	lanager Dr	iver			
	vs\system32			Kernel Dr	river	Yes
Boot	Running	OK	Normal	No	Yes	
gpc	Generic P	acket Class	ifier			
	vs\system32	\drivers\ms	sgpc.sys	Kernel Dr	river	Yes
Manual	Running	OK	Normal	No	Yes	
hpn	hpn	Not Avail		Kernel Dr		No
Disabled	Stopped	OK	Normal	No	. No	N
hpt3xx Disabled	hpt3xx	Not Avail OK	Normal	Kernel Dr No	No No	No
http	Stopped HTTP		vs\system32	110		Kernel
Driver	No	Manual	Stopped	OK	Normal	No
No			~~~PF~~			
i2omgmt	i2omgmt	Not Avail	able	Kernel Dr	river	No
System	Stopped	OK	Normal	No	No	
i2omp	i2omp	Not Avail		Kernel Dr		No
Disabled	Stopped	OK	Normal	No	No	
i8042prt			PS/2 Mouse			Vas
	vs\system32 Running	OK	Normal	Kernel Dr No	Yes	Yes
System ibmhpa			vs\system32			Kernel
Driver	Yes	Manual	Running	OK	Normal	No
Yes						
ibmhpf	IBM Activ	ve PCI Filte	er Driver			
c:\window	vs\system32	\drivers\ibi	mhpf.sys	Kernel Dr	river	Yes
Boot	Running	OK	Normal	No	Yes	
iirsp	iirsp	Not Avail		Kernel Dr		No
Disabled	Stopped	OK na Eiltar D	Normal	No	No	
imapi	cD-Burni s\system32	ng Filter D		Kernel Dr	ivor	No
System	Stopped	OK	Normal	No	No	110
intelide	IntelIde	Not Avail		Kernel Dr		No
Disabled	Stopped	OK	Normal	No	No	
ipfilterdri			Filter Driv	er		
	vs\system32			Kernel Dr		No
Manual	Stopped	OK .	Normal	No	No	
ipinip Kernel Dr		unnel Drive No	erc:\window Manual		_	nıp.sys Normal
No	No	INO	Ivianuai	Stopped	OK	Normai
ipnat		rk Address	Translator			
	vs\system32			Kernel Dr	river	No
Manual	Stopped	OK	Normal	No	No	
ipsec	IPSEC dri	iver		vs\system32		
Kernel Dr		Yes	System	Running	OK	Normal
No	Yes	NT 4 A 11	1.1	V 1D		N
ipsraidn Disabled	ipsraidn Stopped	Not Avail OK	Normal	Kernel Dr No	No	No
irenum		rator Servi		INU	INU	
	vs\system32			Kernel Dr	iver	No
Manual	Stopped	OK	Normal	No	No	
isapnp	PnP ISA/I	EISA Bus I	Driver			
c:\window	vs\system32	2\drivers\isa	1 1 2	Kernel Dr		Yes
Boot	Running	OK	Critical	No	Yes	
kbdclass	2	Class Driv		V1 D		V
System	vs\system32 Running	OK	Normal	Kernel Dr No	Yes	Yes
ksecdd	KSecDD		vs\system32			Kernel
Driver	Yes	Boot	Running	OK	Normal	No
Yes						
lp6nds35	lp6nds35	Not Avail	able	Kernel Dr	river	No
Disabled	Stopped	OK	Normal	No	No	
macxp32	macxp32		vs\system32			Kernel
Driver	Yes	Boot	Running	OK	Normal	No
Yes						

mnmdd Driver Yes	mnmdd Yes	c:\window System	vs\system32 Running	c\drivers\mr OK	nmdd.sys Ignore	Kernel No
modem Driver	Modem No	c:\window Manual		OK	odem.sys Ignore	Kernel No
		ass Driver Yes	c:\window System	vs\system32 Running		ouclass.sys Normal
	Mount Po	int Manage Yes	r c:\window Boot	vs\system32 Running		ountmgr.sys Normal
	mraid35x Stopped	Not Avail OK Client Redi	Normal	Kernel Dr No	iver No	No
c:\window Manual	vs\system32 Stopped	2\drivers\mi OK	xdav.sys Normal	File Syste No	No	No
mrxsmb Driver Yes	MRXSMI Yes	Bc:\window System	vs\system32 Running	?\drivers\mr OK	xsmb.sys Normal	File System No
msfs Driver	Msfs Yes	c:\window System	vs\system32 Running	?\drivers\ms OK	sfs.sys Normal	File System No
Yes mup Driver	Mup Yes	c:\window Boot	vs\system32 Running	?∖drivers\mu OK	ıp.sys Normal	File System No
Yes nal Kernel Dr	Nal Servio	ce No	\??\c:\win Manual	dows\systei Stopped		s\iqvw32.sys Normal
No ndis Kernel Dr		tem Driver Yes	c:\window Boot	vs\system32 Running	\drivers\nd OK	is.sys Normal
No ndistapi	Yes		S TAPI Driv	Ü	OK	Normai
c:\window Manual ndisuio	Running	2\drivers\nd OK ermode I/O	Normal	Kernel Dr No	river Yes	Yes
c:\window Manual	vs\system32 Stopped	2\drivers\nd OK	isuio.sys Normal	Kernel Dr No	iver No	No
ndiswan c:\window		.ccess NDIS 2\drivers\nd	S WAN Dri iswan.sys	ver Kernel Dr	river	Yes
Manual	Running	OK	Normal	No	Yes	
ndproxy Kernel Dr		xy Yes	c:\window Manual	vs\system32 Running		proxy.sys Normal
No	Yes	T C	\ 1		v 1 · · · ·	a :
netbios File Syste No	NetBIOS m Driver Yes	Yes	System	vs\system32 Running		Normal
netbt Kernel Dr	NetBios o iver Yes	ver Tcpip Yes	c:\window System	vs\system32 Running		tbt.sys Normal
No nfrd6m		eRAID 6M	Device Dr	iver		
	•	2\drivers\nfi		Kernel Dr		Yes
Boot nfrd6mnf	Running IBM Serv	OK eraid 6m	Normal Performan	No ce Driver	Yes	
1				Kernel Dr	iver	Yes
Boot	Running	OK	Normal	No	Yes	
nfrd960 c:\window		eRAID 4M 2\drivers\nfi		Lx/5i/6M/6i Kernel Dr		river Yes
Boot	Running	OK	Normal	No	Yes	
npfs	Npfs		-	?\drivers\np	fs.sys Normal	File System
Driver Yes	Yes	System	Running	OK	inoimai	No
ntfs	Ntfs		-	!\drivers\ntf	-	File System
Driver Yes	Yes	Disabled	Running	OK	Normal	No
null	Null	c:\window	vs\system32	!\drivers\nu	ll.sys	Kernel
Driver	Yes	System	Running	OK	Normal	No
Yes						

parport Kernel Dr No	Parallel po iver Yes	ort driver Yes	c:\window Manual	s\system32 Running		port.sys Normal
partmgr Kernel Dr	Partition N	Aanager Yes	c:\window Boot	s\system32 Running	\drivers\par OK	tmgr.sys Normal
No parvdm Driver	Yes Parvdm Yes	c:\window Auto	s\system32 Running	\drivers\par	vdm.sys Ignore	Kernel No
Yes pci	PCI Bus D			s\system32	0	
Kernel Dr No	iver Yes	Yes	Boot	Running	OK	Critical
pciide	PCIIde Yes		•	\drivers\pci		Kernel
Driver Yes	res	Boot	Running	UK	Normal	No
pemeia Driver	Pemeia No	c:\window Disabled	s\system32 Stopped	\drivers\pci OK	ncia.sys Normal	Kernel No
No			• •			110
pdcomp		Not Availa OK		Kernel Dri No	iver No	No
Manual pdframe	Stopped PDFRAM		Ignore Not Availa		Kernel Dri	iver
No	Manual	Stopped	OK	Ignore	No	No
pdreli	PDRELI	Not Availa	able	Kernel Dr	iver	No
Manual	Stopped	OK	Ignore	No	No	
pdrframe	PDRFRAM		Not Availa		Kernel Dr	
No	Manual	Stopped	OK	Ignore	No	No
perc2	perc2	Not Availa		Kernel Dr		No
Disabled	Stopped	OK	Normal	No	No	N
perc2hib	perc2hib	Not Availa OK		Kernel Dri No	No	No
Disabled pptpminip	Stopped		Normal iport (PPTI		INO	
	s\system32			Kernel Dri	iver	Yes
Manual	Running	OK	Normal	No	Yes	1 05
	Processor			s\system32		cessr svs
Kernel Dr	ıver	Yes	Manual	Running	OK	Normal
No No	iver Yes	Yes	Manual	Running	OK	Normal
	Yes	Yes allel Link D		Running	OK	Normal
No ptilink	Yes	allel Link D	Priver	Running Kernel Dri		Normal Yes
No ptilink	Yes Direct Par	allel Link D	Priver	Kernel Dri	iver Yes	Yes
No ptilink c:\window Manual ql1080	Yes Direct Par- ys\system32 Running ql1080	allel Link D \drivers\ptil OK Not Availa	Oriver link.sys Normal able	Kernel Dri No Kernel Dri	iver Yes iver	
No ptilink c:\window Manual ql1080 Disabled	Yes Direct Parays\system32 Running ql1080 Stopped	allel Link D \drivers\ptil OK Not Availa OK	Oriver link.sys Normal able Normal	Kernel Dri No Kernel Dri No	iver Yes iver No	Yes No
No ptilink c:\window Manual ql1080 Disabled ql10wnt	Yes Direct Parassystem32 Running ql1080 Stopped Ql10wnt	allel Link D \drivers\ptil OK Not Availa OK Not Availa	Oriver link.sys Normal able Normal	Kernel Dri No Kernel Dri No Kernel Dri	iver Yes iver No iver	Yes
No ptilink c:\window Manual ql1080 Disabled ql10wnt Disabled	Yes Direct Park vs\system32 Running ql1080 Stopped Ql10wnt Stopped	allel Link D \drivers\ptil OK Not Availa OK Not Availa OK	Oriver link.sys Normal able Normal able Normal	Kernel Dri No Kernel Dri No Kernel Dri No	iver Yes iver No iver No	Yes No No
No ptilink c:\window Manual ql1080 Disabled ql10wnt Disabled ql12160	Yes Direct Par /s\system32 Running q11080 Stopped Q110wnt Stopped q112160	allel Link C \drivers\ptil OK Not Availa OK Not Availa OK Not Availa	Oriver link.sys Normal able Normal able Normal able Normal	Kernel Dri No Kernel Dri No Kernel Dri No Kernel Dri	iver Yes iver No iver No iver	Yes No
No ptilink c:\window Manual ql1080 Disabled ql10wnt Disabled ql12160 Disabled	Yes Direct Par- rs/system32 Running q11080 Stopped Q110wnt Stopped q112160 Stopped	allel Link D \drivers\ptil OK Not Availa OK Not Availa OK Not Availa OK Not Availa	Driver link.sys Normal able Normal able Normal able Normal	Kernel Dri No Kernel Dri No Kernel Dri No Kernel Dri No	iver Yes iver No iver No iver No iver No	Yes No No
No ptilink c:\window Manual ql1080 Disabled ql10wnt Disabled ql12160 Disabled ql1240	Yes Direct Par- zssystem32 Running ql1080 Stopped Ql10wnt Stopped ql12160 Stopped ql1240	allel Link D \drivers\ptil OK Not Availa	Driver link.sys Normal able Normal able Normal able Normal able Normal	Kernel Dri No Kernel Dri No Kernel Dri No Kernel Dri No Kernel Dri	iver Yes iver No iver No iver No iver No iver	Yes No No
No ptilink c:\window Manual ql1080 Disabled ql10wnt Disabled ql12160 Disabled ql1240 Disabled	Yes Direct Pare Sysystem Running ql 1080 Stopped Ql 10 wnt Stopped ql 12160 Stopped ql 1240 Stopped Stopped ql 1240 Stopped Stopped Running Run	allel Link D \drivers\ptil OK Not Availa	Driver link.sys Normal able Normal able Normal able Normal able Normal	Kernel Dri No Kernel Dri No Kernel Dri No Kernel Dri No Kernel Dri No	iver Yes iver No iver No iver No iver No	Yes No No No
No ptilink c:\window Manual ql1080 Disabled ql10wnt Disabled ql12160 Disabled ql1240	Yes Direct Pares/s\system32 Running ql1080 Stopped Ql10wnt Stopped ql12160 Stopped ql1240 Stopped ql1240	allel Link D \drivers\ptil OK Not Availa	Driver link.sys Normal able Normal able Normal able Normal able Normal	Kernel Dri No Kernel Dri No Kernel Dri No Kernel Dri No Kernel Dri	iver Yes iver No iver No iver No iver No	Yes No No
No ptilink c:\window Manual ql1080 Disabled ql10wnt Disabled ql12160 Disabled ql1240 Disabled ql1240	Yes Direct Pare Sysystem Running ql 1080 Stopped Ql 10 wnt Stopped ql 12160 Stopped ql 1240 Stopped Stopped ql 1240 Stopped Stopped Running Run	allel Link D \drivers\ptil OK Not Availa	Driver link.sys Normal able Normal able Normal able Normal able Normal able Normal	Kernel Dri No Kernel Dri No Kernel Dri No Kernel Dri No Kernel Dri No Kernel Dri	iver Yes iver No iver No iver No iver No iver No	Yes No No No
No ptilink c:\window Manual ql1080 Disabled ql10wnt Disabled ql12160 Disabled ql1240 Disabled ql1280 Disabled	Yes Direct Parassystem32 Running ql1080 Stopped Ql10wnt Stopped ql12160 Stopped ql1240 Stopped ql1280 Stopped ql1280 Stopped ql2100 Stopped	allel Link D \drivers\ptil OK Not Availa	Driver link.sys Normal able Normal able Normal able Normal able Normal able Normal	Kernel Dri No Kernel Dri No Kernel Dri No Kernel Dri No Kernel Dri No	iver Yes iver No iver No iver No iver No iver No	Yes No No No No
No ptilink c:\window Manual ql1080 Disabled ql10wnt Disabled ql12160 Disabled ql1240 Disabled ql1280 Disabled ql1280	Yes Direct Partys/system32 Running ql1080 Stopped Ql10wnt Stopped ql12160 Stopped ql1240 Stopped ql1280 Stopped ql1280 Stopped ql2100	allel Link D \drivers\ptil OK Not Availa	Driver link.sys Normal able Normal able Normal able Normal able Normal able Normal	Kernel Dri No Kernel Dri No Kernel Dri No Kernel Dri No Kernel Dri No Kernel Dri No Kernel Dri No	iver Yes iver No	Yes No No No No
No ptilink c:\window Manual ql1080 Disabled ql10wnt Disabled ql12160 Disabled ql1240 Disabled ql1280 Disabled ql2100 Disabled ql2200 Disabled	Yes Direct Part Sysystem32 Running ql1080 Stopped Ql10wnt Stopped ql12160 Stopped ql1240 Stopped ql1280 Stopped ql1280 Stopped ql2100 Stopped ql2200 Stopped	allel Link D \drivers\ptil OK Not Availa	Driver link.sys Normal able Normal able Normal able Normal able Normal able Normal able Normal	Kernel Dri No Kernel Dri No Kernel Dri No Kernel Dri No Kernel Dri No Kernel Dri No Kernel Dri No Kernel Dri No	iver Yes iver No	Yes No No No No No No No No No
No ptilink c:\window Manual ql1080 Disabled ql10wnt Disabled ql12160 Disabled ql1240 Disabled ql1280 Disabled ql2100 Disabled ql2200 Disabled ql2200	Yes Direct Part Sysystem32 Running ql1080 Stopped Ql10wnt Stopped ql12160 Stopped ql1240 Stopped ql1280 Stopped ql1280 Stopped ql2100 Stopped ql2200 Stopped ql2200 Stopped ql2300	allel Link D \drivers\ptil OK Not Availa OK	Driver link.sys Normal able Normal	Kernel Dri No Kernel Dri No No No No No No No No No No No No No	iver Yes iver No	Yes No No No No No No No
No ptilink c:\window Manual ql1080 Disabled ql110wnt Disabled ql12160 Disabled ql1240 Disabled ql1280 Disabled ql2100 Disabled ql2200 Disabled ql2300 Disabled	Yes Direct Part Sysystem32 Running ql1080 Stopped Ql10wnt Stopped ql12160 Stopped ql1240 Stopped ql1280 Stopped ql2100 Stopped ql2200 Stopped ql2300 Stopped ql2300 Stopped	allel Link D \drivers\ptil OK Not Availa OK	Driver link.sys Normal able Normal	Kernel Dri No Kernel Dri No Kernel Dri No Kernel Dri No Kernel Dri No Kernel Dri No Kernel Dri No Kernel Dri No Kernel Dri No	iver Yes iver No	Yes No No No No No No No No No
No ptilink c:\window Manual ql1080 Disabled ql10wnt Disabled ql12160 Disabled ql1280 Disabled ql2100 Disabled ql2200 Disabled ql2300 Disabled rasacd	Yes Direct Part Sysystem32 Running ql1080 Stopped Ql10wnt Stopped ql12160 Stopped ql1240 Stopped ql1280 Stopped ql2100 Stopped ql2200 Stopped ql2300 Stopped Remote A	allel Link D \drivers\ptil OK Not Availa OK CK Not Availa OK Not Availa	Driver link.sys Normal able Normal	Kernel Dri No Cernel Dri No Kernel Dri No Cernel Dri No Cernel Dri No Cernel Dri No Driver	iver Yes iver No	Yes No
No ptilink c:\window Manual q11080 Disabled q110wnt Disabled q112160 Disabled q11240 Disabled q12200 Disabled q12200 Disabled q12300 Disabled q12300 Disabled c12300 Disabled Dis	Yes Direct Parissystem32 Running ql1080 Stopped ql12160 Stopped ql1240 Stopped ql1280 Stopped ql1280 Stopped ql2200 Stopped ql2200 Stopped ql2300 Stopped ql2300 Stopped stopped ql2300 Stopped	allel Link D \drivers\ptil OK Not Availa	Driver link.sys Normal able Connection acd.sys	Kernel Dri No Kernel Dri No No No No No No No No No No No No No	iver Yes iver No	Yes No No No No No No No No No
No ptilink c:\window Manual ql1080 Disabled ql10wnt Disabled ql12160 Disabled ql1240 Disabled ql1280 Disabled ql2200 Disabled ql2200 Disabled cl2200 Disabled cl2300 Disabled Disabled cl2300 Disabled	Yes Direct Parissystem32 Running ql1080 Stopped Ql10wnt Stopped ql12160 Stopped ql1240 Stopped ql1280 Stopped ql2200 Stopped ql2300 Stopped ql2300 Stopped Remote Assystem32 Running	allel Link D \drivers\ptil OK Not Availa OK	Driver link.sys Normal able Connection acd.sys Normal	Kernel Dri No Cernel Dri No Kernel Dri No Cernel Dri No Cernel Dri No Cernel Dri No Driver	iver Yes iver No	Yes No
No ptilink c:\window Manual ql1080 Disabled ql10wnt Disabled ql12160 Disabled ql1240 Disabled ql1280 Disabled ql2200 Disabled ql2300 Disabled rasacd c:\window System rasl2tp	Yes Direct Part Sysystem32 Running ql1080 Stopped Ql10wnt Stopped ql12160 Stopped ql1240 Stopped ql1280 Stopped ql2200 Stopped ql2300 Stopped ql2300 Stopped Remote Acsysystem32 Running WAN Mir	allel Link D \drivers\ptil OK Not Availa OK	Driver link.sys Normal able Normal	Kernel Dri No Kernel Dri No No No No No No No No No No No No No	iver Yes iver No	Yes No No No No No No No No Yes
No ptilink c:\window Manual ql1080 Disabled ql12160 Disabled ql1240 Disabled ql1240 Disabled ql2200 Disabled ql2300 Disabled c:\window System rasl2tp c:\window	Yes Direct Part //s\system32 Running ql1080 Stopped Ql10wnt Stopped ql12160 Stopped ql1240 Stopped ql1280 Stopped ql2200 Stopped ql2300 Stopped ql2300 Stopped Remote A. /s\system32 Running WAN Mir /s\system32	allel Link D \drivers\ptil OK Not Availa OK	Driver link.sys Normal able Normal	Kernel Dri No	iver Yes iver No iver	Yes No
No ptilink c:\window Manual ql1080 Disabled ql12160 Disabled ql1240 Disabled ql12200 Disabled ql2200 Disabled ql2300 Disabled rasacd c:\window System rasl2tp c:\window Manual	Yes Direct Part Sysystem32 Running ql1080 Stopped Ql10wnt Stopped ql12160 Stopped ql1240 Stopped ql1280 Stopped ql2200 Stopped ql2300 Stopped ql2300 Stopped Remote A. Sysystem32 Running WAN Mir Sysystem32 Running	allel Link D \drivers\ptil OK Not Availa OK	Driver link.sys Normal able Normal	Kernel Dri No Kernel Dri No No No No No No No No No No No No No	iver Yes iver No	Yes No No No No No No No No Yes
No ptilink c:\window Manual ql1080 Disabled ql110wnt Disabled ql12160 Disabled ql1240 Disabled ql2200 Disabled ql2300 Disabled rasacd c:\window System rasl2tp c:\window Manual raspppoe	Yes Direct Part Sysystem32 Running ql1080 Stopped Ql10wnt Stopped ql12160 Stopped ql1240 Stopped ql1280 Stopped ql2200 Stopped ql2300 Stopped ql2300 Stopped Remote Acts Sysystem32 Running WAN Mir Sysystem32 Running Remote Acts	allel Link D \drivers\ptil OK Not Availa OK Ccess Auto \drivers\rasi OK \drivers\rasi OK \drivers\rasi OK \ccess PPPO	Driver link.sys Normal able Vormal able Normal able Vormal able Normal able Normal able Normal able Normal bbe Normal bbe Normal bbe Normal Connection acd.sys Normal P)	Kernel Dri No	iver Yes iver No iver Ves iver Yes	Yes No No No No No No No No Yes
No ptilink c:\window Manual ql1080 Disabled ql110wnt Disabled ql12160 Disabled ql1240 Disabled ql2200 Disabled ql2300 Disabled rasacd c:\window System rasl2tp c:\window Manual raspppoe	Yes Direct Part Sysystem32 Running ql1080 Stopped Ql10wnt Stopped ql12160 Stopped ql1240 Stopped ql1280 Stopped ql2200 Stopped ql2300 Stopped ql2300 Stopped Remote A. Sysystem32 Running WAN Mir Sysystem32 Running	allel Link D \drivers\ptil OK Not Availa OK Ccess Auto \drivers\rasi OK \drivers\rasi OK \drivers\rasi OK \ccess PPPO	Driver link.sys Normal able Vormal able Normal able Vormal able Normal able Normal able Normal able Normal bbe Normal bbe Normal bbe Normal Connection acd.sys Normal P)	Kernel Dri No Driver Kernel Dri No Kernel Dri No	iver Yes iver No iver Ves iver Yes	Yes No No No No No No No Yes Yes
No ptilink c:\window Manual ql1080 Disabled ql110wnt Disabled ql12160 Disabled ql1240 Disabled ql2100 Disabled ql2200 Disabled ql2300 Disabled rasacd c:\window System rasl2tp c:\window Manual rasppoe c:\window c:\window C:\window C:\window C:\window Manual rasppoe c:\window C:\window Manual rasppoe c:\window	Yes Direct Part Sysystem32 Running ql1080 Stopped Ql10wnt Stopped ql12160 Stopped ql1240 Stopped ql1280 Stopped ql2200 Stopped ql2300 Stopped ql2300 Stopped Remote Acts Sysystem32 Running WAN Mircs Sysystem32 Running Remote Acts Sysystem32	allel Link D \drivers\ptil OK Not Availa OK Ceess Auto \drivers\ras OK \drivers\ras OK \drivers\ras OK \drivers\ras OK \drivers\ras OK	Driver link.sys Normal able Vormal able Normal bble Normal acd.sys Normal bc) 12tp.sys Normal bc Driver pppoe.sys Normal	Kernel Dri No Cernel Dri No Cernel Dri No Cernel Dri No Kernel Dri No	iver Yes iver No iver Yes iver Yes	Yes No No No No No No No Yes Yes
No ptilink c:\window Manual ql1080 Disabled ql112160 Disabled ql12240 Disabled ql2200 Disabled ql2200 Disabled rasacd c:\window System rasl2tp c:\window Manual rasppoe c:\window Manual	Yes Direct Part Sysystem32 Running ql1080 Stopped Ql10wnt Stopped ql12160 Stopped ql1240 Stopped ql1280 Stopped ql2200 Stopped ql2300 Stopped ql2300 Stopped Remote Acts System32 Running WAN Mirt Sysystem32 Running Remote Acts Sysystem32 Running Remote Acts Sysystem32 Running Direct Part	allel Link D \drivers\ptil OK Not Availa OK Ceess Auto \drivers\ras OK \drivers\ras OK \drivers\ras OK \drivers\ras OK \drivers\ras OK	Driver link.sys Normal able Vormal able Normal bble Normal acd.sys Normal bc) 12tp.sys Normal bc Driver pppoe.sys Normal	Kernel Dri No Loriver Kernel Dri No Kernel Dri No Loriver Kernel Dri No	iver Yes iver No iver Yes iver Yes	Yes No No No No No No No Yes Yes
No ptilink c:\window Manual ql1080 Disabled ql110wnt Disabled ql12160 Disabled ql1240 Disabled ql12200 Disabled ql2200 Disabled ql2200 Disabled rasacd c:\window System rasl2tp c:\window Manual rasppoe c:\window Manual raspti	Yes Direct Part Sysystem32 Running ql1080 Stopped Ql10wnt Stopped ql12160 Stopped ql1240 Stopped ql1280 Stopped ql2200 Stopped ql2300 Stopped ql2300 Stopped Remote Acts System32 Running WAN Mirt Sysystem32 Running Remote Acts Sysystem32 Running Remote Acts Sysystem32 Running Direct Part	allel Link D \drivers\ptil OK Not Availa OK Coess Auto Ok drivers\ras OK coess PPPO \drivers\ras OK allel	Driver link.sys Normal able Tormal able Normal able Tormal able Normal able Tormal acd.sys Normal De) 12tp.sys	Kernel Dri No Loriver Kernel Dri No	iver Yes iver No iver No iver No iver No iver No iver No iver Ves iver Yes iver Yes iver	Yes No No No No No No No No Yes Yes Yes yes

rdbss	Rdbss	c:\window	vs\system32	\drivers\rd	oss.svs	File System
Driver Yes	Yes	System	Running		Normal	No
rdpcdd		c:\window				Kernel
Driver Yes	Yes	System	Running	OK	Ignore	No
rdpdr		Server Dev		tor Driver		
	vs\system32			Kernel Dr		Yes
Manual rdpwd	Running RDPWD	OK c:\window	Normal vs\system32	No \drivers\rds	Yes	Kernel
Driver	No	Manual	Stopped	OK	Ignore	No
No redbook	Digital CI	O Audio Pla	wback Filte	r Driver		
	vs\system32			Kernel Dr	iver	Yes
System	Running	OK	Normal	No	Yes	
secdrv	Secdry		vs\system32		-	Kernel
Driver No	No	Manual	Stopped	OK	Normal	No
serenum	Serenum I	Filter Drive	r c:\window	s\system32	!\drivers\sei	renum.sys
Kernel Di		Yes	Manual	Running	OK	Normal
No serial	Yes Serial por	t driver	o:\vvindov	ra\arratam2	?\drivers\sei	cial ava
Kernel Di		Yes	System	Running	OK	Ignore
No	Yes	100	System	114	011	18.1010
sfloppy	Sfloppy		vs\system32			Kernel
Driver No	No	System	Stopped	OK	Ignore	No
simbad	Simbad	Not Avail	able	Kernel Dr	river	No
Disabled	F F	OK	Normal	No	No	
sparrow	Sparrow	Not Avail		Kernel Dr		No
Disabled srv	Stopped Srv	OK	Normal vs\system32	No \drivers\ar	No	File System
Driver	Yes	Manual	Running	OK	Normal	No
Yes						
Yes swenum		Bus Driver				•
Yes swenum Kernel Di	river	Bus Driver Yes	c:\window Manual	vs\system32 Running	?\drivers\sw OK	renum.sys Normal
Yes swenum	Yes symc810		Manual		OK	•
Yes swenum Kernel Di No symc810 Disabled	Yes symc810 Stopped	Yes Not Avail OK	Manual able Normal	Running Kernel Dr No	OK iver No	Normal No
Yes swenum Kernel Di No symc810 Disabled symc8xx	Yes symc810 Stopped symc8xx	Yes Not Avail OK Not Avail	Manual able Normal able	Running Kernel Dr No Kernel Dr	OK iver No iver	Normal
Yes swenum Kernel Di No symc810 Disabled symc8xx Disabled	Yes symc810 Stopped symc8xx Stopped	Yes Not Avail OK Not Avail OK	Manual able Normal able Normal	Running Kernel Dr No Kernel Dr No	OK iver No iver No	Normal No No
Yes swenum Kernel Di No symc810 Disabled symc8xx	Yes symc810 Stopped symc8xx	Yes Not Avail OK Not Avail OK	Manual able Normal able	Running Kernel Dr No Kernel Dr No	OK iver No iver No	Normal No
Yes swenum Kernel Dr No symc810 Disabled symc8xx Disabled symmpi Driver Yes	Yes symc810 Stopped symc8xx Stopped symmpi Yes	Yes Not Avail OK Not Avail OK c:\window Boot	Manual able Normal able Normal vs\system32 Running	Running Kernel Dr No Kernel Dr No Adrivers\sy OK	OK iver No iver No mmpi.sys Normal	Normal No No Kernel No
Yes swenum Kernel Di No symc810 Disabled symc8xx Disabled symmpi Driver Yes sym_hi	Yes symc810 Stopped symc8xx Stopped symmpi Yes sym_hi	Yes Not Avail OK Not Avail OK c:\window Boot Not Avail	Manual able Normal able Normal vs\system32 Running able	Running Kernel Dr No Kernel Dr No Adrivers\sy OK Kernel Dr	OK iver No iver No mmpi.sys Normal	Normal No No Kernel
Yes swenum Kernel Dr No symc810 Disabled symc8xx Disabled symmpi Driver Yes	Yes symc810 Stopped symc8xx Stopped symmpi Yes	Yes Not Avail OK Not Avail OK c:\window Boot	Manual able Normal able Normal vs\system32 Running able Normal	Running Kernel Dr No Kernel Dr No Adrivers\sy OK	OK iver No iver No mmpi.sys Normal iver No	Normal No No Kernel No
Yes swenum Kernel Di No symc810 Disabled symc8xx Disabled symmpi Driver Yes sym_hi Disabled	Yes symc810 Stopped symc8xx Stopped symmpi Yes sym_hi Stopped sym_u3 Stopped	Yes Not Avail OK Not Avail OK e:\window Boot Not Avail OK Not Avail OK	Manual able Normal able Normal ys\system32 Running able Normal able Normal	Running Kernel Dr No Kernel Dr No Adrivers\sy OK Kernel Dr No	OK iver No iver No mmpi.sys Normal iver No	Normal No No Kernel No No
Yes swenum Kernel Dr No symc810 Disabled symc8xx Disabled symmpi Driver Yes sym_hi Disabled sym_u3 Disabled tcpip	Yes symc810 Stopped symc8xx Stopped symmpi Yes sym_hi Stopped sym_u3 Stopped TCP/IP Pr	Yes Not Avail OK Not Avail OK e:\window Boot Not Avail OK Not Avail OK rotocol Driv	Manual able Normal able Normal zs\system32 Running able Normal able Normal	Running Kernel Dr No Kernel Dr No Adrivers\sy OK Kernel Dr No Kernel Dr No Kernel Dr No	OK iver No iver No mmpi.sys Normal iver No iver No	Normal No No Kernel No No No
Yes swenum Kernel Dr No symc810 Disabled symc8xx Disabled symmpi Driver Yes sym_hi Disabled sym_u3 Disabled tcpip c:\windov	Yes symc810 Stopped symc8xx Stopped symmpi Yes sym_hi Stopped sym_u3 Stopped TCP/IP Pr vs\system32	Yes Not Avail OK Not Avail OK c:\window Boot Not Avail OK Not Avail OK rotocol Driv c\drivers\tcp	Manual able Normal able Normal zs\system32 Running able Normal able Normal	Running Kernel Dr No Kernel Dr	OK iver No iver No mmpi.sys Normal iver No iver No	Normal No No Kernel No No
Yes swenum Kernel Dr No symc810 Disabled symc8xx Disabled symmpi Driver Yes sym_hi Disabled sym_u3 Disabled tcpip c:\window System	Yes symc810 Stopped symc8xx Stopped symmpi Yes sym_hi Stopped sym_u3 Stopped TCP/IP Pr	Yes Not Avail OK Not Avail OK c:\window Boot Not Avail OK Not Avail OK rotocol Driv Okrivers\tcp	Manual able Normal able Normal zs\system32 Running able Normal able Normal	Running Kernel Dr No Kernel Dr No Adrivers\sy OK Kernel Dr No Kernel Dr No Kernel Dr No	OK iver No iver No mmpi.sys Normal iver No iver No iver Yes	Normal No No Kernel No No No
Yes swenum Kernel Dr No symc810 Disabled symc8xx Disabled symmpi Driver Yes sym_hi Disabled sym_u3 Disabled tcpip c:\windov System tdpipe Driver	Yes symc810 Stopped symc8xx Stopped symmpi Yes sym_hi Stopped sym_u3 Stopped TCP/IP Pr vs\system32 Running	Yes Not Avail OK Not Avail OK c:\window Boot Not Avail OK Not Avail OK rotocol Driv Okrivers\tcp	Manual able Normal able Normal zs\system32 Running able Normal able Normal zer pip.sys Normal	Running Kernel Dr No Kernel Dr No Adrivers\sy OK Kernel Dr No Kernel Dr No Kernel Dr No	OK iver No iver No mmpi.sys Normal iver No iver No iver Yes	Normal No No Kernel No No Yes
Yes swenum Kernel Dr No symc810 Disabled symc8xx Disabled symmpi Driver Yes sym_hi Disabled sym_u3 Disabled tcpip c:\windov System tdpipe Driver No	Yes symc810 Stopped symc8xx Stopped symmpi Yes sym_hi Stopped sym_u3 Stopped TCP/IP Pr vs\system32 Running TDPIPE No	Yes Not Avail OK Not Avail OK c:\window Boot Not Avail OK Not Avail OK votocol Driv Cdrivers\tcp OK c:\window Manual	Manual able Normal able Normal zs\system32 Running able Normal able Normal able Normal zs\system32	Running Kernel Dr No Chdrivers\tdp	OK iver No iver No mmpi.sys Normal iver No iver No iver Yes pipe.sys Ignore	Normal No No Kernel No No Ves Kernel No
Yes swenum Kernel Di No symc810 Disabled symc8xx Disabled symmpi Driver Yes sym_hi Disabled sym_u3 Disabled tcpip c:\window System tdpipe Driver No tdtcp	river Yes symc810 Stopped symc8xx Stopped symmpi Yes sym_hi Stopped sym_u3 Stopped TCP/IP Pr vs\system32 Running TDPIPE No TDTCP	Yes Not Avail OK Not Avail OK c:\window Boot Not Avail OK Not Avail OK votocol Driv chrivers\tcp OK c:\window Manual c:\window	Manual able Normal able Normal ss\system32 Running able Normal able Normal ver pip.sys Normal zs\system32 Stopped	Running Kernel Dr No Kernel Dr No Adrivers\sy OK Kernel Dr No Kernel Dr No Kernel Dr No Adrivers\tdl OK	OK iver No iver No mmpi.sys Normal iver No iver No iver Yes pipe.sys Ignore cp.sys	Normal No No Kernel No No Ves Kernel No Kernel
Yes swenum Kernel Dr No symc810 Disabled symc8xx Disabled symmpi Driver Yes sym_hi Disabled sym_u3 Disabled tcpip c:\windov System tdpipe Driver No	Yes symc810 Stopped symc8xx Stopped symmpi Yes sym_hi Stopped sym_u3 Stopped TCP/IP Pr vs\system32 Running TDPIPE No	Yes Not Avail OK Not Avail OK c:\window Boot Not Avail OK Not Avail OK votocol Driv Cdrivers\tcp OK c:\window Manual	Manual able Normal able Normal zs\system32 Running able Normal able Normal able Normal zs\system32	Running Kernel Dr No Chdrivers\tdp	OK iver No iver No mmpi.sys Normal iver No iver No iver Yes pipe.sys Ignore	Normal No No Kernel No No Ves Kernel No
Yes swenum Kernel Dr No symc810 Disabled symc8xx Disabled symmpi Driver Yes sym_hi Disabled sym_u3 Disabled tcpip c:\window System tdpipe Driver No tdtcp Driver No termdd	yes symc810 Stopped symc8xx Stopped symmpi Yes sym_hi Stopped sym_u3 Stopped TCP/IP Prossystem32 Running TDPIPE No TDTCP No	Yes Not Avail OK Not Avail OK e:\window Boot Not Avail OK Not Avail OK rotocol Driv Notivers\tep OK e:\window Manual c:\window Manual Device Driv	Manual able Normal able Normal zs\system32 Running able Normal able Normal zer pip.sys Normal zs\system32 Stopped zs\system32 Stopped ver	Running Kernel Dr No No No No Kernel Dr No	OK iver No iver No mmpi.sys Normal iver No iver No iver Yes pipe.sys Ignore cp.sys	Normal No No No Kernel No No Yes Kernel No Kernel No
Yes swenum Kernel Dr No symc810 Disabled symc8xx Disabled symmpi Driver Yes sym_hi Disabled sym_u3 Disabled tcpip c:\window System tdpipe Driver No tdtcp Driver No termdd c:\window color window cywindow cywindow cywindow color window color window color window color window color window system tdpipe Driver No tdtcp Driver No tdtcp Driver No termdd ccolor window color	yes symc810 Stopped symc8xx Stopped symmpi Yes sym_hi Stopped sym_u3 Stopped TCP/IP Prossystem32 Running TDPIPE No TDTCP No Terminal 1 Terminal 1	Yes Not Avail OK Not Avail OK e:\window Boot Not Avail OK Not Avail OK rotocol Driv Notivers\tep OK e:\window Manual c:\window Manual Device Driv Notivers\tep	Manual able Normal able Normal rs\system32 Running able Normal able Normal rer pip.sys Normal rs\system32 Stopped rs\system32 Stopped ver mdd.sys	Running Kernel Dr No No No No No Kernel Dr	OK iver No iver No mmpi.sys Normal iver No iver Yes pipe.sys Ignore cp.sys Ignore	Normal No No Kernel No No Ves Kernel No Kernel
Yes swenum Kernel Dr No symc810 Disabled symc8xx Disabled symmpi Driver Yes sym_hi Disabled sym_u3 Disabled tcpip c:\window System tdpipe Driver No tdtcp Driver No termdd c:\window System	yes symc810 Stopped symc8xx Stopped symmpi Yes sym_hi Stopped sym_u3 Stopped TCP/IP Prossystem32 Running TDPIPE No TDTCP No Terminal 1 vs/system32 Running	Yes Not Avail OK Not Avail OK e:\window Boot Not Avail OK Not Avail OK rotocol Driv c\drivers\tcp OK e:\window Manual c:\window Manual Device Driv c\drivers\ter OK	Manual able Normal able Normal zs\system32 Running able Normal able Normal zer pip.sys Normal zs\system32 Stopped zs\system32 Stopped ver mdd.sys Normal	Running Kernel Dr No No No No Kernel Dr No	OK iver No iver No mmpi.sys Normal iver No iver Yes pipe.sys Ignore cp.sys Ignore iver Yes	Normal No No No Kernel No No Yes Kernel No Kernel No Yes
Yes swenum Kernel Dr No symc810 Disabled symc8xx Disabled symmpi Driver Yes sym_hi Disabled sym_u3 Disabled tcpip c:\window System tdpipe Driver No tdtcp Driver No termdd c:\window color window cywindow cywindow cywindow color window color window color window color window color window system tdpipe Driver No tdtcp Driver No tdtcp Driver No termdd ccolor window color	yes symc810 Stopped symc8xx Stopped symmpi Yes sym_hi Stopped sym_u3 Stopped TCP/IP Prossystem32 Running TDPIPE No TDTCP No Terminal 1 Terminal 1	Yes Not Avail OK Not Avail OK e:\window Boot Not Avail OK Not Avail OK rotocol Driv Notivers\tep OK e:\window Manual c:\window Manual Device Driv Notivers\tep	Manual able Normal able Normal zs\system32 Running able Normal able Normal zer pip.sys Normal zs\system32 Stopped zs\system32 Stopped ver mdd.sys Normal	Running Kernel Dr No	OK iver No iver No mmpi.sys Normal iver No iver Yes pipe.sys Ignore cp.sys Ignore iver Yes	Normal No No No Kernel No No Yes Kernel No Kernel No
Yes swenum Kernel Dr No symc810 Disabled symc8xx Disabled symmpi Driver Yes sym_hi Disabled sym_u3 Disabled tcpip c:\window System tdpipe Driver No tdtcp Driver No termdd c:\window System toside Disabled udfs	river Yes symc810 Stopped symc8xx Stopped symmpi Yes sym_hi Stopped sym_u3 Stopped TCP/IP Pr vs\system32 Running TDPIPE No TDTCP No Terminal vs\system32 Running TosIde Stopped Udfs	Yes Not Avail OK Not Avail OK c:\window Boot Not Avail OK Not Avail OK Not Avail OK cotocol Driv Odrivers\tcp OK c:\window Manual c:\window Manual Device Driv Odrivers\ter OK Not Avail OK C:\window Not Avail OK C:\window	Manual able Normal able Normal zs\system32 Running able Normal able Normal zer bip.sys Normal zs\system32 Stopped zs\system32 Stopped ver mdd.sys Normal able Normal	Running Kernel Dr No	OK iver No iver No mmpi.sys Normal iver No iver No iver Yes pipe.sys Ignore cp.sys Ignore iver Yes iver No fs.sys	Normal No No Kernel No No No Yes Kernel No Kernel No Kernel No File System
Yes swenum Kernel Dr No symc810 Disabled symc8xx Disabled symmpi Driver Yes sym_hi Disabled sym_u3 Disabled tcpip c:\window System tdpipe Driver No tdtcp Driver No termdd c:\window System toside Disabled udfs Driver	yes symc810 Stopped symc8xx Stopped symmpi Yes sym_hi Stopped sym_u3 Stopped TCP/IP Pr vs\system32 Running TDPIPE No TDTCP No Terminal vs\system32 Running Toslde Stopped	Yes Not Avail OK Not Avail OK c:\window Boot Not Avail OK Not Avail OK Not Avail OK cotocol Driv Odrivers\tcp OK c:\window Manual c:\window Manual Device Driv Odrivers\ter OK Not Avail OK Not Avail OK	Manual able Normal able Normal zs\system32 Running able Normal able Normal zer pip.sys Normal zer stopped zs\system32 Stopped zs\system32 Stopped ver mdd.sys Normal able Normal	Running Kernel Dr No	OK iver No iver No mmpi.sys Normal iver No iver No iver Yes pipe.sys Ignore cp.sys Ignore iver Yes iver No	Normal No No No Kernel No No Yes Kernel No Kernel No Yes No
Yes swenum Kernel Dr No symc810 Disabled symc8xx Disabled symmpi Driver Yes sym_hi Disabled sym_u3 Disabled tcpip c:\window System tdpipe Driver No tdtcp Driver No termdd c:\window System toside Disabled udfs	river Yes symc810 Stopped symc8xx Stopped symmpi Yes sym_hi Stopped sym_u3 Stopped TCP/IP Pr vs\system32 Running TDPIPE No TDTCP No Terminal vs\system32 Running TosIde Stopped Udfs	Yes Not Avail OK Not Avail OK c:\window Boot Not Avail OK Not Avail OK Not Avail OK cotocol Driv Odrivers\tcp OK c:\window Manual c:\window Manual Device Driv Odrivers\ter OK Not Avail OK C:\window Not Avail OK C:\window	Manual able Normal able Normal zs\system32 Running able Normal able Normal zer sip.sys Normal zs\system32 Stopped zs\system32 Stopped ver mdd.sys Normal able Normal able Normal zs\system32 Stopped	Running Kernel Dr No	OK iver No iver No mmpi.sys Normal iver No iver No iver Yes pipe.sys Ignore cp.sys Ignore iver Yes iver No iver No iver Yes pipe.sys Ignore	Normal No No Kernel No No No Yes Kernel No Kernel No Kernel No File System

		Update Di					Default Monitor	No	MONITO		5.1.2001.0	
c:\windows\s	-		-	Kernel Dr		Yes	(Standard monitor typ		monitor.in		Not Availa	
	unning		Normal	No	Yes		DISPLAY\DEFAULT					
				vs\system32			ServerWorks Champi 5.2.3790.0 10/1/2002				No	SYSTEM
Kernel Driver	es	Yes	Manual	Running	UK	Normal	3.2.3/90.0 10/1/2002 Available	Serverwo	rks (RCC)	macnine.ir	11	Not
		USD Open	Host Cont	roller Minip	ort Driver		PCI\VEN_1166&DE	V 02018-5	IIDCVC 00	ሰበበበበበ <u>የ</u> -	EV 02\2 8r	267461648
c:\windows\s				Kernel Dr		Yes	0&78	v_0201&S	06313_00	UUUUUUXK	E V_93\3&.	20/A010A&
	unning		Normal	No	Yes	1 68	Motherboard resource	20	No	SVSTEM	5.2.3790.0	10/1/2002
		lay Control		110	1 03		(Standard system dev		machine.ir		Not Availa	
c:\windows\s				Kernel Dr	iver	Yes	ACPI\PNP0C02\2	1003)	macimic.n	11	1 vot 11 vanc	aoic
	unning		Ignore	No	Yes	105	Standard 101/102-Ke	v or Micros	soft Natural	PS/2 Kevb	oard	No
-	iaIde	Not Availa	_	Kernel Dr		No	KEYBOARD					keyboard.inf
Disabled St			Normal	No	No		Not Available		20303\4&35	,	-	,
	torage vo			vs\system32	\drivers\vo	lsnap.svs	PS/2 Compatible Mou		No		5.2.3790.0	10/1/2002
Kernel Drive	_	Yes	Boot	Running		Normal	Microsoft msmouse.		Not Availa			
No Yo	es						ACPI\PNP0F13\4&3	5118DFF&	0			
wanarp Re	emote A	cess IP AF	RP Driver				Standard floppy disk	controller	No	FDC	5.2.3790.0	10/1/2002
c:\windows\s	system32	drivers\wa	narp.sys	Kernel Dr	iver	Yes	(Standard floppy disk	controllers	3)	fdc.inf	Not Availa	able
Manual Ru	unning	OK	Normal	No	Yes		ACPI\PNP0700\4&35	5118DFF&	0			
wdica W	/DICA	Not Availa	able	Kernel Dr	iver	No	Floppy disk drive	No	FLOPPYD	DISK	5.2.3790.0	10/1/2002
Manual St	topped	OK	Ignore	No	No		(Standard floppy disk	drives)	flpydisk.in	ıf	Not Availa	able
wlbs No	etwork L	oad Balanc	eing				FDC\GENERIC_FLC	OPPY_DRI				
c:\windows\s	system32	drivers\wll	bs.sys	Kernel Dr	iver	No	Printer Port	No			10/1/2002	(Standard
Manual St	topped	OK	Normal	No	No		port types) msports.in		Not Availa		ACPI\PNF	
							Printer Port Logical In	nterface	No	SYSTEM	5.2.3790.0	10/1/2002
[Signed Drive	ers]						(Standard system dev		machine.ir		Not Availa	able
							LPTENUM\MICROS					
Device Name		Signed	Device Cl		Driver Ve		Communications Port				10/1/2002	`
Driver Date		Manufactu	ırer	INF Name	Driver Na	me	port types) msports.in		Not Availa		ACPI\PNF	
Device ID							Advanced programma					5.2.3790.0
Not Available		Not Availa		Not Availa		Not	10/1/2002 (Standard	-	/	machine.ir	1f	Not
Available No			Not Avail	able	Not Availa	able	Available ACPI\PNI				5.0.05 00.0	10/1/2002
Not Available		HTREE\R		COLUMNIA	ED	5.0.0700.0	Direct memory access				5.2.3790.0	
ACPI Multip			No	COMPUT		5.2.3790.0	(Standard system dev		machine.ir	1İ	Not Availa	able
10/1/2002 (S			nai.ini	Not Availa	able		ACPI\PNP0200\4&35			5 2 2700 0	10/1/2002	(C+
ROOT\ACPI				M-	CVCTEM	5 2 2700 O	System timer	No			10/1/2002	(Standard
Microsoft AC 10/1/2002 M				No	SISIEM	5.2.3790.0	system devices) ACPI\PNP0100\4&35	machine.ii		Not Availa	able	
ACPI HAL\I		-	Not Avaii	abic			System CMOS/real ti			CVCTEM	5.2.3790.0	10/1/2002
Processor No		PROCESS	SOR.	5 2 3790 0	10/1/2002	(Standard	(Standard system dev		machine.ir		Not Availa	
processor typ		cpu.inf	Not Avail		10/1/2002	(Standard	ACPI\PNP0B00\4&3	/		11	Not Availe	aoic
ACPI\GENU					I. 2\ 0		System speaker	No		5 2 3790 0	10/1/2002	(Standard
Processor No		PROCESS			0 10/1/2002	(Standard	system devices)	machine.ii		Not Availa		(Standard
processor typ		cpu.inf	Not Avail		. 10, 1, 2002	(Suman a	ACPI\PNP0800\4&35			110011114111		
ACPI\GENU					L 2\ 1		Numeric data process		No	SYSTEM	5.2.3790.0	10/1/2002
PCI bus No				0 10/1/2002		system	(Standard system dev		machine.ir	nf	Not Availa	able
	achine.in		Not Avail		ACPI\PNI		ACPI\PNP0C04\4&3					
ServerWorks	(RCC)	CMIC LE I	Processor to	o PCI Bridg	ge(*)	No	Motherboard resource	es	No	SYSTEM	5.2.3790.0	10/1/2002
SYSTEM 5.3	2.3790.0	10/1/2002	ServerWo	rks (RCC)	machine.ii	nf	(Standard system dev	ices)	machine.ir	nf	Not Availa	able
Not Available	e						ACPI\PNP0C02\3					
PCI\VEN_11	166&DEV	/_0014&S	UBSYS_0	0000000&R	EV_31\3&	267A616A&	CSB5 IDE Controller	No	HDC	5.2.3790.0	10/1/2002	
0&00							ServerWorks	mshdc.inf	Not Availa	able		
ServerWorks	(RCC)	CMIC_LE I	Processor to	o PCI Bridg	ge(*)	No	PCI\VEN_1166&DE	V_0212&S	UBSYS_02	121166&R	EV_93\3&2	267A616A&
SYSTEM 5.3	2.3790.0	10/1/2002	ServerWo	orks (RCC)	machine.ii	nf	0&79					
Not Available							Primary IDE Channel		HDC	5.2.3790.0	10/1/2002	(Standard
PCI\VEN_11	166&DEV	/_0014&S	UBSYS_0	0000000&R	EV_00\3&	267A616A&	IDE ATA/ATAPI cor	ntrollers)	mshdc.inf	Not Availa	able	
0&01							PCIIDE\IDECHANN	EL\4&102	4D5C6&0&	0.0		
ServerWorks						No	CD-ROM Drive	No			10/1/2002	(Standard
SYSTEM 5.		10/1/2002	ServerWo	orks (RCC)	machine.ii	nf	CD-ROM drives)		Not Availa	able		
Not Available							IDE\CDROMLG_CD	_	D-8484B_			2.01\5
PCI\VEN_11	166&DEV	/_0014&S	UBSYS_0	0000000&R	EV_00\3&	267A616A&	&FB0C83D&0&0.0.0					
0&02	OY E	2.5				Diani :	Secondary IDE Chang		No	HDC		10/1/2002
RAGE XL PO			-		No	DISPLAY	(Standard IDE ATA/A				Not Availa	able
5.10.2600.60		8/8/2001	ATI Tech	nologies Inc	Э.	atiixpad.inf	PCIIDE\IDECHANN				N T	HOD
Not Available		7 47500 C	IIDOVO A	340101405	EV 05/00	267461646	ServerWorks (RCC) I					USB
PCI\VEN_10 0&48	JU2&DEV	/_4/52&S	OBSAS_02	2401014&R	EV_27\3&	20/A616A&	5.2.3790.0 10/1/2002	ServerWo	rks (RCC)	usbport.in	ı Not Avaıla	able

0&48

```
PCI\VEN 1166&DEV 0220&SUBSYS 02201166&REV 05\3&267A616A&
                                                                                    DISKDRIVE
                                                                                                      5.2.3790.0 10/1/2002 (Standard
                                                                 Disk drive No
0&7A
                                                                  disk drives)
                                                                                    disk.inf Not Available
USB Root Hub
                                                                  SCSI\DISK&VEN IBM-ESXS&PROD ST318305LC !#&REV B244\4&
                  No
                           USB
                                    5.2.3790.0 10/1/2002 (Standard
USB Host Controller) usbport.inf Not Available
                                                                  30BEF77E&0&000
USB\ROOT HUB\4&AF5358C&0
                                                                  Disk drive No
                                                                                    DISKDRIVE
                                                                                                      5.2.3790.0 10/1/2002 (Standard
Serverworks Champion CSB5 - SouthBridge 5 LPC
                                                      SYSTEM
                                                                  disk drives)
                                                                                    disk.inf Not Available
                                                                  SCSI \backslash DISK\&VEN\_LSILOGIC\&PROD\_1030\_IM\&REV\_1000 \backslash 4\&30BEF77E
5.2.3790.0 10/1/2002 ServerWorks (RCC) machine.inf
                                                      Not
Available
                                                                  &0&020
PCI\VEN 1166&DEV 0225&SUBSYS 00000000&REV 00\3&267A616A&
                                                                  SCSI Processor Device
                                                                                             No
                                                                                                      SYSTEM 5.2.3790.0 10/1/2002
0&7B
                                                                  IBM
                                                                           scsidev.inf Not Available
                                    SYSTEM 5.2.3790.0 10/1/2002
                                                                  SCSI\PROCESSOR&VEN IBM&PROD 02R0962A S320 1&REV 1\4&30
ISAPNP Read Data Port
                           No
(Standard system devices)
                           machine.inf
                                             Not Available
                                                                  BEF77E&0&080
ISAPNP\READDATAPORT\0
                                                                  IBM Dummy Device No
                                                                                             SYSTEM 5.2.3790.0 10/1/2002 IBM
ServerWorks Grand Champion CIOB_X2 - I/O Bridge 133 Mhz No
                                                                  scsidev.inf Not Available
SYSTEM 5.2.3790.0 10/1/2002 ServerWorks (RCC) machine.inf
                                                                  SCSI\BRIDGE&VEN_IBM&PROD_DUMMY_DEVICE&REV_4.80\4&30B
Not Available
                                                                  EF77E&0&100
PCI\VEN 1166&DEV 0101&SUBSYS 00000000&REV 03\3&267A616A&
                                                                  LSI Logic PCI-X Ultra320 SCSI Host Adapter
0&80
                                                                  SCSIADAPTER
                                                                                    5.2.3790.0 10/1/2002 LSI Logic Inc.
                                                                                                                        pnpscsi.inf
ServerWorks Grand Champion CIOB X2 - I/O Bridge 133 Mhz No
                                                                  Not Available
                                                                  PCI\VEN 1000&DEV 0030&SUBSYS 026C1014&REV 07\3&1070020&0
SYSTEM 5.2.3790.0 10/1/2002 ServerWorks (RCC) machine.inf
Not Available
                                                                  &39
PCI\VEN 1166&DEV 0101&SUBSYS 00000000&REV 03\3&267A616A&
                                                                  IBM Dummy Device No
                                                                                             SYSTEM 5.2.3790.0 10/1/2002 IBM
0&82
                                                                  scsidev.inf Not Available
                                                                  SCSI\BRIDGE&VEN_IBM&PROD_DUMMY_DEVICE&REV_4.80\4&7DD
ServerWorks Grand Champion CIOB X2 - I/O Bridge 133 Mhz No
SYSTEM 5.2.3790.0 10/1/2002 ServerWorks (RCC) machine.inf
                                                                  DE1C&0&100
                                                                  PCI bus No
                                                                                    SYSTEM 5.2.3790.0 10/1/2002 (Standard system
Not Available
PCI\VEN_1166&DEV_0101&SUBSYS_00000000&REV 03\3&267A616A&
                                                                  devices)
                                                                           machine.inf
                                                                                             Not Available
                                                                                                               ACPI\PNP0A03\3
                                                                  Mylex AcceleRAID 352 Controller (Accelerated)
0&88
                                                                                                               Nο
ServerWorks Grand Champion CIOB X2 - I/O Bridge 133 Mhz No
                                                                  SCSIADAPTER
                                                                                    7.5.1.0
                                                                                             12/13/2002
                                                                                                               Mylex
                                                                                                                        oem7.inf
SYSTEM 5.2.3790.0 10/1/2002 ServerWorks (RCC) machine.inf
                                                                  Not Available
Not Available
                                                                  PCI\VEN 1069&DEV 0050&SUBSYS 00501069&REV 02\3&29E81982&0
PCI\VEN_1166&DEV_0101&SUBSYS_00000000&REV_03\3&267A616A&
                                                                  &28
0&8A
                                                                  SCSI Processor Device
                                                                                             No
                                                                                                      SYSTEM 5.2.3790.0 10/1/2002
PCI bus
                  SYSTEM 5.2.3790.0 10/1/2002 (Standard system
                                                                           scsidev.inf Not Available
                                             ACPI\PNP0A03\1
                                                                  SCSI\PROCESSOR&VEN_IBM&PROD_EXP300___S160&REV_D014\4&8
devices) machine.inf
                           Not Available
Mylex AcceleRAID 352 Controller (Accelerated)
                                                                  B4695D&0&0F0
                                             No
                                                                  SCSI Processor Device
                                                                                                      SYSTEM 5.2.3790.0 10/1/2002
SCSIADAPTER
                  7.5.1.0
                          12/13/2002
                                             Mylex
                                                      oem7.inf
                                                                                             No
Not Available
                                                                           scsidev.inf Not Available
PCI\VEN_1069&DEV_0050&SUBSYS_00501069&REV_02\3&13C0B0C5&
                                                                  SCSI\PROCESSOR&VEN IBM&PROD EXP300 S160&REV D014\4&8
0&08
                                                                  B4695D&0&1F0
SCSI Processor Device
                                    SYSTEM 5.2.3790.0 10/1/2002
                                                                  Mylex RAID Disk Device
                                                                                             No
                                                                                                      DISKDRIVE
                                                                                                                        5.2.3790.0
                                                                                    disk.inf
                                                                                            Not Available
         scsidev.inf Not Available
                                                                  10/1/2002 Mylex
SCSI\PROCESSOR&VEN IBM&PROD EXP300 S160&REV D014\4&3
                                                                  SCSI\DISK&VEN MYLEX&PROD ACCELERAID 352&REV 0701\4&8B
8A862AB&0&0F0
                                                                  4695D&0&200
                                    SYSTEM 5.2.3790.0 10/1/2002
                                                                  Mylex GAM Device No
                                                                                             SYSTEM 5.2.3790.0 10/1/2002 Mylex
SCSI Processor Device
                           No
         scsidev.inf Not Available
                                                                  scsidev.inf Not Available
IBM
SCSI\PROCESSOR&VEN IBM&PROD EXP300 S160&REV D014\4&3
                                                                  SCSI\PROCESSOR&VEN MYLEX&PROD GAM DEVICE&REV \4&8B4
8A862AB&0&1F0
                                                                  695D&0&460
                                    DISKDRIVE
                                                      5.2.3790.0
                                                                                    SYSTEM 5.2.3790.0 10/1/2002 (Standard system
Mylex RAID Disk Device
                                                                  PCI bus No
                           Nο
10/1/2002 Mylex
                  disk.inf
                           Not Available
                                                                                             Not Available
                                                                                                               ACPI\PNP0A03\4
                                                                  devices)
                                                                           machine.inf
SCSI\DISK&VEN_MYLEX&PROD_ACCELERAID_352&REV_0701\4&38
                                                                  Mylex AcceleRAID 352 Controller (Accelerated)
                                                                                                               Nο
                                                                  SCSIADAPTER
A862AB&0&200
                                                                                    7.5.1.0
                                                                                             12/13/2002
                                                                                                               Mylex
                                                                                                                        oem7.inf
                           SYSTEM 5.2.3790.0 10/1/2002 Mylex
Mylex GAM Device No
                                                                  Not Available
scsidev.inf Not Available
                                                                  PCI\VEN 1069&DEV 0050&SUBSYS 00501069&REV 02\3&172E68DD&
SCSI\PROCESSOR&VEN_MYLEX&PROD_GAM_DEVICE&REV_\4&38A
                                                                  0&30
862AB&0&460
                                                                  SCSI Processor Device
                                                                                                      SYSTEM 5.2.3790.0 10/1/2002
                                                                                             No
Broadcom NetXtreme Gigabit Ethernet
                                                      2.91.0.0
                                                                           scsidev.inf Not Available
                                                                  SCSI\PROCESSOR&VEN_IBM&PROD_EXP300___S160&REV_D014\4&3
10/1/2002 Broadcom netb57xp.inf
                                    Not Available
PCI\VEN 14E4&DEV 16A7&SUBSYS 026F1014&REV 02\3&13C0B0C5
                                                                  4FC159&0&0F0
&0&40
                                                                  SCSI Processor Device
                                                                                                      SYSTEM 5.2.3790.0 10/1/2002
PCI bus
                  SYSTEM 5.2.3790.0 10/1/2002 (Standard system
                                                                           scsidev.inf Not Available
                                             ACPI\PNP0A03\2
devices) machine.inf
                           Not Available
                                                                  SCSI\PROCESSOR&VEN IBM&PROD EXP300 S160&REV D014\4&3
LSI Logic PCI-X Ultra320 SCSI Host Adapter
                                                                  4FC159&0&1F0
                                             No
SCSIADAPTER
                  5.2.3790.0 10/1/2002 LSI Logic Inc.
                                                      pnpscsi.inf
                                                                  Mylex RAID Disk Device
                                                                                             No
                                                                                                      DISKDRIVE
                                                                  10/1/2002 Mylex
                                                                                    disk.inf
                                                                                            Not Available
Not Available
PCI\VEN 1000&DEV 0030&SUBSYS 026C1014&REV 07\3&1070020&0
                                                                  SCSI\DISK&VEN MYLEX&PROD ACCELERAID 352&REV 0701\4&34
&38
                                                                  FC159&0&200
```

Myday CAM Daviga No. SYSTEM 5.2.2700.0.10/1/2002 Myday	CDC Diele Eilten Deivon Not Aveilable LECA CVDDIVED
Mylex GAM Device No SYSTEM 5.2.3790.0 10/1/2002 Mylex	CRC Disk Filter Driver Not Available LEGACYDRIVER Not Available Not Available Not Available Not
scsidev.inf Not Available SCSI\PROCESSOR&VEN MYLEX&PROD GAM DEVICE&REV \4&34F	Available Not Available ROOT\LEGACY_CRCDISK\0000
C159&0&460	dmboot Not Available LEGACYDRIVER Not Available
Not Available Not Available Not Available Not	Not Available Not Available Not Available Not
Available Not Available Not Available Not Available	Available ROOT\LEGACY DMBOOT\0000
Not Available ACPI\IBM37D4\2&DABA3FF&0	dmload Not Available LEGACYDRIVER Not Available
IBM Active PCI Device No SYSTEM 5.1.1.1 2/7/2003	Not Available Not Available Not Available Not
IBM Corporation oeml.inf Not Available	Available ROOT\LEGACY_DMLOAD\0000
ACPI\ASF0001\2&DABA3FF&0	Fips Not Available LEGACYDRIVER Not Available
ACPI Fixed Feature Button No SYSTEM 5.2.3790.0 10/1/2002	Not Available Not Available Not Available Not
(Standard system devices) machine.inf Not Available	Available ROOT\LEGACY_FIPS\0000
ACPI\FIXEDBUTTON\2&DABA3FF&0	Generic Packet Classifier Not Available LEGACYDRIVER
Logical Disk ManagerNo SYSTEM 5.2.3790.0 10/1/2002 (Standard	Not Available Not Available Not Available Not
system devices) machine.inf Not Available	Available Not Available ROOT\LEGACY GPC\0000
ROOT\DMIO\0000	IPSEC driver Not Available LEGACYDRIVER Not
Volume Manager No SYSTEM 5.2.3790.0 10/1/2002 (Standard	Available Not Available Not Available Not Available
system devices) machine.inf Not Available	Not Available ROOT\LEGACY IPSEC\0000
ROOT\FTDISK\0000	ksecdd Not Available LEGACYDRIVER Not Available
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft	Not Available Not Available Not Available Not
volume.inf Not Available	Available ROOT\LEGACY_KSECDD\0000
STORAGE\VOLUME\1&30A96598&0&SIGNATURE3F2CE4BAOFFSET7E	macxp32 Not Available LEGACYDRIVER Not Available
0000LENGTH722B99E00	Not Available Not Available Not Available Not
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft	Available ROOT\LEGACY_MACXP32\0000
volume.inf Not Available	mnmdd Not Available LEGACYDRIVER Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURE3F2CE4BAOFFSET72	Not Available Not Available Not Available Not
3381C00LENGTH36E8D8200	Available ROOT\LEGACY_MNMDD\0000
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft	mountmgr Not Available LEGACYDRIVER Not Available
volume.inf Not Available	Not Available Not Available Not
STORAGE\VOLUME\1&30A96598&0&SIGNATURE3F2CE4BAOFFSETA9	Available ROOT\LEGACY_MOUNTMGR\0000
1C61C00LENGTH138835B000	Nal Service Not Available LEGACYDRIVER Not
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft	Available Not Available Not Available Not Available
volume.inf Not Available	Not Available ROOT\LEGACY_NAL\0000
STORAGE\VOLUME\1&30A96598&0&SIGNATURE1CC73790OFFSET7E	NDIS System Driver Not Available LEGACYDRIVER Not
00LENGTH43BF94800	Available Not Available Not Available Not Available
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft	Not Available ROOT\LEGACY NDIS\0000
volume.inf Not Available	Remote Access NDIS TAPI Driver Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURE1OFFSET7E0000LEN	LEGACYDRIVER Not Available Not Available Not
GTHB00146200	Available Not Available Not Available
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft	ROOT\LEGACY_NDISTAPI\0000
volume.infNot Available	NDIS Usermode I/O Protocol Not Available LEGACYDRIVER
STORAGE\VOLUME\1&30A96598&0&SIGNATURE3F2CE4BCOFFSET7E	Not Available Not Available Not
0000LENGTH722B99E00	Available Not Available ROOT\LEGACY_NDISUIO\0000
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft	NDProxy Not Available LEGACYDRIVER Not Available
volume.inf Not Available	Not Available Not Available Not
STORAGE\VOLUME\1&30A96598&0&SIGNATURE3F2CE4BCOFFSET72	Available ROOT\LEGACY_NDPROXY\0000
3381C00LENGTH36E8D8200	NetBios over Tcpip Not Available LEGACYDRIVER Not
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft	Available Not Available Not Available Not Available
volume.inf Not Available	Not Available ROOT\LEGACY_NETBT\0000
STORAGE\VOLUME\1&30A96598&0&SIGNATURE3F2CE4BBOFFSET7E	IBM ServeRAID 6M Device Driver Not Available
0000LENGTH722B99E00	LEGACYDRIVER Not Available Not Available Not
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft	Available Not Available Not Available
volume.inf Not Available	ROOT\LEGACY_NFRD6M\0000
STORAGE\VOLUME\1&30A96598&0&SIGNATURE3F2CE4BBOFFSET72	IBM ServeRAID 6M Performance Driver Not Available
3381C00LENGTH36E8D8200	LEGACYDRIVER Not Available Not Available Not
Generic volume No VOLUME 5.2.3790.0 10/1/2002 Microsoft	Available Not Available Not Available
volume.inf Not Available	ROOT\LEGACY_NFRD6MPF\0000
STORAGE\VOLUME\1&30A96598&0&SIGNATURE3F2CE4BBOFFSETA9	IBM ServeRAID 4M/4Mx/4L/4Lx/5i/6M/6i Device Driver Not
1C61C00LENGTH138835B000	Available LEGACYDRIVER Not Available Not Available
AFD Networking Support Environment Not Available	Not Available Not Available Not Available
LEGACYDRIVER Not Available Not Available Not	ROOT\LEGACY_NFRD960\0000
Available Not Available Not Available	Null Not Available LEGACYDRIVER Not Available
ROOT\LEGACY_AFD\0000	Not Available Not Available Not Available Not
Beep Not Available LEGACYDRIVER Not Available	Available ROOT\LEGACY_NULL\0000
Not Available Not Available Not Available Not	Partition Manager Not Available LEGACYDRIVER Not
Available ROOT\LEGACY_BEEP\0000	Available Not Available Not Available Not Available
	Not Available ROOT\LEGACY_PARTMGR\0000

Parvdm Not Available LEGACYDRIVER Not Available	Variable Value User Name
Not Available Not Available Not Available Not	ClusterLog C:\WINDOWS\Cluster\cluster.log <system></system>
Available ROOT\LEGACY_PARVDM\0000	ComSpec %SystemRoot%\system32\cmd.exe <system></system>
Remote Access Auto Connection Driver Not Available	NUMBER_OF_PROCESSORS 2 <system></system>
LEGACYDRIVER Not Available Not Available Not	OS Windows_NT <system></system>
Available Not Available Not Available	Path
ROOT\LEGACY_RASACD\0000	%SystemRoot%\system32;%SystemRoot%;%SystemRoot%\System32\Wbem;
RDPCDD Not Available LEGACYDRIVER Not Available	C:\Program Files\Microsoft SQL Server\MSSQL\BINN;C:\Program
Not Available Not Available Not Available Not	Files\Microsoft SQL Server\80\Tools\BINN <system></system>
Available ROOT\LEGACY_RDPCDD\0000	PATHEXT
TCP/IP Protocol Driver Not Available LEGACYDRIVER	.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH <system></system>
Not Available Not Available Not Available Not	PROCESSOR_ARCHITECTURE x86 <system></system>
Available Not Available ROOT\LEGACY_TCPIP\0000	PROCESSOR_IDENTIFIER x86 Family 15 Model 2 Stepping 7,
VGA Display Controller. Not Available LEGACYDRIVER	GenuineIntel <system></system>
Not Available Not Available Not Available Not	PROCESSOR LEVEL 15 <system></system>
Available Not Available ROOT\LEGACY_VGASAVE\0000	PROCESSOR REVISION 0207 <system></system>
volsnap Not Available LEGACYDRIVER Not Available	TEMP %SystemRoot%\TEMP <system></system>
Not Available Not Available Not	TMP %SystemRoot%\TEMP <system></system>
Available ROOT\LEGACY VOLSNAP\0000	windir %SystemRoot% <system></system>
Remote Access IP ARP Driver Not Available LEGACYDRIVER	TEMP %USERPROFILE%\Local Settings\Temp NT
Not Available Not Available Not Available Not	AUTHORITY\SYSTEM
Available Not Available ROOT\LEGACY_WANARP\0000	TMP %USERPROFILE%\Local Settings\Temp NT
Audio Codecs No MEDIA 5.2.3790.0 10/1/2002 (Standard	AUTHORITY\SYSTEM
system devices) wave.inf Not Available	TEMP %USERPROFILE%\Local Settings\Temp NT
ROOT\MEDIA\MS MMACM	AUTHORITY\LOCAL SERVICE
Legacy Audio Drivers No MEDIA 5.2.3790.0 10/1/2002 (Standard	TMP %USERPROFILE%\Local Settings\Temp NT
system devices) wave.inf Not Available	AUTHORITY\LOCAL SERVICE
ROOT\MEDIA\MS MMDRV	TEMP %USERPROFILE%\Local Settings\Temp NT
Media Control Devices No MEDIA 5.2.3790.0 10/1/2002	AUTHORITY\NETWORK SERVICE
(Standard system devices) wave.inf Not Available	TMP %USERPROFILE%\Local Settings\Temp NT
ROOT\MEDIA\MS_MMMCI	AUTHORITY\NETWORK SERVICE
Legacy Video Capture Devices No MEDIA 5.2.3790.0 10/1/2002	TEMP %USERPROFILE%\Local Settings\Temp
(Standard system devices) wave.inf Not Available	IBMSERVER3\Administrator
ROOT\MEDIA\MS_MMVCD	TMP %USERPROFILE%\Local Settings\Temp
Video Codecs No MEDIA 5.2.3790.0 10/1/2002 (Standard	IBMSERVER3\Administrator
system devices) wave.inf Not Available	1DIVISER VERS (Administrator
ROOT\MEDIA\MS MMVID	[Print Jobs]
WAN Miniport (L2TP) No NET 5.2.3790.0 10/1/2002	[11111 3003]
Microsoft netrasa.inf Not Available	Document Size Owner Notify Status Time Submitted
ROOT\MS_L2TPMINIPORT\0000	Start Time Until Time Elapsed Time Pages Printed Job ID
WAN Miniport (IP) No NET 5.2.3790.0 10/1/2002 Microsoft	Priority Parameters Driver Print Processor Host Print Queue
netrasa.inf Not Available ROOT\MS NDISWANIP\0000	Data Type Name
WAN Miniport (PPPOE) No NET 5.2.3790.0 10/1/2002	Data Type Name
Microsoft netrasa.inf Not Available	[Network Connections]
ROOT\MS PPPOEMINIPORT\0000	[Network Connections]
WAN Miniport (PPTP) No NET 5.2.3790.0 10/1/2002	Local Name Remote Name Type Status User Name
Microsoft netrasa.inf Not Available	X: \\192.168.132.253\\\ Disk Current Connection
ROOT\MS PPTPMINIPORT\0000	IBMSERVER3\Administrator
Direct Parallel No NET 5.2.3790.0 10/1/2002 Microsoft	IDIVISER VERS (Administrator
netrasa.inf Not Available ROOT\MS PTIMINIPORT\0000	[Running Tasks]
Terminal Server Device Redirector No SYSTEM 5.2.3790.0	[Kulling 1d5k5]
10/1/2002 (Standard system devices) machine.inf Not	Name Path Process ID Priority Min Working Set Max
Available ROOT\RDPDR\0000	Working Set Start Time Version Size File Date
Terminal Server Keyboard Driver No SYSTEM 5.2.3790.0	system idle process Not Available 0 Not
10/1/2002 (Standard system devices) machine.inf Not	Available Not Available Not Available Not Available
,	
Available ROOT\RDP_KBD\0000	Not Available Not Available
Terminal Server Mouse Driver No SYSTEM 5.2.3790.0 10/1/2002	system Not Available 4 8 0 1413120
(Standard system devices) machine.inf Not Available	Not Available Not Available Not Available Not
ROOT\RDP_MOU\0000	Available
Plug and Play Software Device Enumerator No SYSTEM 5.2.3790.0	smss.exe Not Available 344 11 204800 1413120
10/1/2002 (Standard system devices) machine.inf Not	7/7/2003 2:51 PM Not Available Not Available Not
Available ROOT\SYSTEM\0000	Available
Microcode Update Device No SYSTEM 5.2.3790.0 10/1/2002	csrss.exe Not Available 468 13 Not Available
(Standard system devices) machine.inf Not Available	Not Available 7/7/2003 2:51 PM Not Available Not
ROOT\SYSTEM\0001	Available Not Available
[F V V V	winlogon.exe c:\windows\system32\winlogon.exe 492
[Environment Variables]	13 204800 1413120 7/7/2003 2:51 PM 5.2.3790.0

700 AM	Services.exe	10324-2048 720.00 KB (737,280 bytes) 6/9/2003 10 10 10 10 10 11 12 10 10 12 13 14 15 13 14 15 16 14 15 16 15 16 16 16 17 17 16 18 18 17 18 18 18 18 18 18 18
143120 77/2003 2-51 PM 5.23790.0 (sero) 3 trus 030324-2048 13.00 KB (13.12 bytes) 3252003 13.00 KB (13.12 bytes) 3252003 13.00 KB (13.12 bytes) 3252003 14.00 KB (13.12 byt	12:46 PM	rsion Size File Date Manufacturer Path .3790.0 (srv03_rtm.030324-2048) 536.50 KB (549,376 5/2003 7:00 AM Microsoft Corporation stem32\winlogon.exe .3790.0 (srv03_rtm.030324-2048) 722.50 KB (739,840 5/2003 7:00 AM Microsoft Corporation stem32\ntdll.dll .3790.0 (srv03_rtm.030324-2048) 965.00 KB (988,160 5/2003 7:00 AM Microsoft Corporation stem32\wedge kernel32.dll .3790.0 (srv03_rtm.030324-2048) 319.50 KB (327,168 5/2003 7:00 AM Microsoft Corporation stem32\wedge kernel32\windersymbol{microsoft} Corporation stem32\windersymbol{microsoft} Corporation Stem32microsoft
Tool AM	Loaded Mode Sass.exe Cwindows/system32\sass.exe 548 9 204800 1413120 777/2003 2:51 PM 5.2.3790.0 (srv03 rtm.030324-2048) 3/25/2003 7:00 AM svchost.exe c.\windows/system32\svchost.exe 680 8 204800 1413120 777/2003 2:51 PM 5.2.3790.0 5.2 5003 7:00 AM 5.2	rsion Size File Date Manufacturer Path .3790.0 (srv03_rtm.030324-2048) 536.50 KB (549,376 5/2003 7:00 AM Microsoft Corporation stem32\winlogon.exe .3790.0 (srv03_rtm.030324-2048) 722.50 KB (739,840 5/2003 7:00 AM Microsoft Corporation stem32\ntdll.dll .3790.0 (srv03_rtm.030324-2048) 965.00 KB (988,160 5/2003 7:00 AM Microsoft Corporation stem32\wternel32.dll .3790.0 (srv03_rtm.030324-2048) 319.50 KB (327,168 5/2003 7:00 AM Microsoft Corporation stem32\wternel32.dll .3790.0 (srv03_rtm.030324-2048) 559.50 KB (572,928 5/2003 7:00 AM Microsoft Corporation
Subsect Cwindows/system32 Subsect S4 9 204800	Sass.exe C:\windows\system32\sass.exe 548 9 204800	rsion Size File Date Manufacturer Path .3790.0 (srv03_rtm.030324-2048) 536.50 KB (549,376 5/2003 7:00 AM Microsoft Corporation stem32\winlogon.exe .3790.0 (srv03_rtm.030324-2048) 722.50 KB (739,840 5/2003 7:00 AM Microsoft Corporation stem32\ntdll.dll .3790.0 (srv03_rtm.030324-2048) 965.00 KB (988,160 5/2003 7:00 AM Microsoft Corporation stem32\wternel32.dll .3790.0 (srv03_rtm.030324-2048) 319.50 KB (327,168 5/2003 7:00 AM Microsoft Corporation stem32\wternel32.dll .3790.0 (srv03_rtm.030324-2048) 559.50 KB (572,928 5/2003 7:00 AM Microsoft Corporation
Hal 130 OR Bit (13.15 bytes) 3.257003 7:00 AM 680 or 8	1413120	.3790.0 (srv03_rtm.030324-2048) 536.50 KB (549,376 5/2003 7:00 AM Microsoft Corporation stem32\winlogon.exe .3790.0 (srv03_rtm.030324-2048) 722.50 KB (739,840 5/2003 7:00 AM Microsoft Corporation stem32\ntdll.dll .3790.0 (srv03_rtm.030324-2048) 965.00 KB (988,160 5/2003 7:00 AM Microsoft Corporation stem32\kernel32.dll .3790.0 (srv03_rtm.030324-2048) 319.50 KB (327,168 5/2003 7:00 AM Microsoft Corporation stem32\kernel32\kernel32.dll .3790.0 (srv03_rtm.030324-2048) 559.50 KB (572,928 5/2003 7:00 AM Microsoft Corporation stem32\kernel3032\kernel32\kernel32\kernel32\kernel3032\kernel32\kernel32\kernel3032\kernel32\kernel32\kernel3032\kernel32\kernel3032\kernel32\kernel30
Symbol 12 12 12 12 12 12 12 1	Sychost.exe C:\windows\system32\sychost.exe 680 8 204800 1413120 7/7/2003 2:51 PM 5.2.3790.0	5/2003 7:00 AM Microsoft Corporation stem32\winlogon.exe .3790.0 (srv03_rtm.030324-2048) 722.50 KB (739,840 5/2003 7:00 AM Microsoft Corporation stem32\ntdll.dll .3790.0 (srv03_rtm.030324-2048) 965.00 KB (988,160 5/2003 7:00 AM Microsoft Corporation stem32\kernel32.dll .3790.0 (srv03_rtm.030324-2048) 319.50 KB (327,168 5/2003 7:00 AM Microsoft Corporation stem32\msvcrt.dll .3790.0 (srv03_rtm.030324-2048) 559.50 KB (572,928 5/2003 7:00 AM Microsoft Corporation
	Chail Color Colo	stem32\winlogon.exe .3790.0 (srv03_rtm.030324-2048) 722.50 KB (739,840 5/2003 7:00 AM Microsoft Corporation stem32\ntdll.dll .3790.0 (srv03_rtm.030324-2048) 965.00 KB (988,160 5/2003 7:00 AM Microsoft Corporation stem32\kernel32.dll .3790.0 (srv03_rtm.030324-2048) 319.50 KB (327,168 5/2003 7:00 AM Microsoft Corporation stem32\windextrace{\text{me}} 319.50 KB (327,168 5/2003 7:00 AM Microsoft Corporation stem32\windextrace{\text{me}} 319.50 KB (572,928 5/2003 7:00 AM Microsoft Corporation
Serior 1900 A	Str. 03 _ Ttm. 030324-2048 13.00 KB (13,312 bytes) 3/25/2003	.3790.0 (srv03_rtm.030324-2048) 722.50 KB (739,840 5/2003 7:00 AM Microsoft Corporation stem32\ntdll.dll .3790.0 (srv03_rtm.030324-2048) 965.00 KB (988,160 5/2003 7:00 AM Microsoft Corporation stem32\kernel32.dll .3790.0 (srv03_rtm.030324-2048) 319.50 KB (327,168 5/2003 7:00 AM Microsoft Corporation stem32\msvcrt.dll .3790.0 (srv03_rtm.030324-2048) 559.50 KB (572,928 5/2003 7:00 AM Microsoft Corporation
Second S	Svchost.exe C:\windows\system32\svchost.exe 748 8 C:\windows\system32\svchost.exe 748 8 C:\windows\system32\svchost.exe 3/25/2003 7:00 AM Svchost.exe Not Available 880 8 Not Available Not Av	stem32\ntdll.dll .3790.0 (srv03_rtm.030324-2048) 965.00 KB (988,160 5/2003 7:00 AM Microsoft Corporation stem32\kernel32.dll .3790.0 (srv03_rtm.030324-2048) 319.50 KB (327,168 5/2003 7:00 AM Microsoft Corporation stem32\msvcrt.dll .3790.0 (srv03_rtm.030324-2048) 559.50 KB (572,928 5/2003 7:00 AM Microsoft Corporation
	204800	.3790.0 (srv03_rtm.030324-2048) 965.00 KB (988,160 5/2003 7:00 AM Microsoft Corporation stem32\kernel32.dll 319.50 KB (327,168 5/2003 7:00 AM Microsoft Corporation stem32\msvcrt.dll 319.50 KB (327,168 5/2003 7:00 AM Microsoft Corporation stem32\msvcrt.dll 5/2003 7:00 AM Microsoft Corporation 5/2003 7:00 AM Microsoft Corporation
Comparison Co	7:00 AM svchost.exe Not Available Svchost.exe c:\windows\system32\svchost.exe 912 8 204800 1413120 7/7/2003 2:51 PM 5.2.3790.0 (srv03_rtm.030324-2048) 13.00 KB (13,312 bytes) 3/25/2003 7:00 AM msdtc.exe Not Available Not Ava	stem32\kernel32.dll .3790.0 (srv03_rtm.030324-2048) 5/2003 7:00 AM Microsoft Corporation stem32\msvcrt.dll .3790.0 (srv03_rtm.030324-2048) 5/2003 7:00 AM Microsoft Corporation
Such as Such	Svchost.exe	.3790.0 (srv03_rtm.030324-2048) 319.50 KB (327,168 5/2003 7:00 AM Microsoft Corporation stem32\msvcrt.dll 5790.0 (srv03_rtm.030324-2048) 559.50 KB (572,928 5/2003 7:00 AM Microsoft Corporation
Available Not Available	Available Not Available	5/2003 7:00 AM Microsoft Corporation stem32\msvcrt.dll .3790.0 (srv03_rtm.030324-2048) 559.50 KB (572,928 5/2003 7:00 AM Microsoft Corporation
Such State Suc	Svchost.exe C:\windows\system32\svchost.exe 912 8 204800 1413120 7/7/2003 2:51 PM 5.2.3790.0 5.2	.3790.0 (srv03_rtm.030324-2048) 559.50 KB (572,928 5/2003 7:00 AM Microsoft Corporation
204800	204800 1413120 7/7/2003 2:51 PM 5.2.3790.0 (srv03_rtm.030324-2048) 13.00 KB (13,312 bytes) 3/25/2003 7:00 AM msdtc.exe Not Available 1004 8 Not Available Not Available 7/7/2003 2:51 PM Not Available	5/2003 7:00 AM Microsoft Corporation
Carrology at 1.00 kB 13.312 bytes 3/25/2003 17.00 AM 1.00 kB 13.210 17.72003 17.38 bytes 1.00 kB 13.20 17.72003 17.38 bytes 1.00 kB 13.20 17.72003 17.00 AM 13.20 17.72003 17.00 AM 13.120 17.72003 17.00 AM	(srv03_rtm.030324-2048) 13.00 KB (13,312 bytes) 3/25/2003 c:\windows\syrcon AM rpcrt4 5.2 sytes) 3/2 for the color of	*
msdtc ex Not Available	Most Available	
Not Available Not Available Not Available Not Available Sevenost exe clivindows/system32/sychost.exe 1108 system52 syste	Not Available	
Available Not Available Not Available Sevents Event Sevents Se	Available Not Available svchost.exe	
204800	204800 1413120 7/7/2003 2:51 PM 5.2.3790.0	.3790.0 (srv03_rtm.030324-2048) 562.00 KB (575,488
Serviol 3 TIII 300 CR 13,312 bytes 3/25/2003 3/25/2003 7/20 AM Microsoft Corporation bytes 6/9/2003 1/38 PM 5.1.1 14.50 KB (14,848 bytes 6/9/2003 1/38 PM 5.2.3790.0 (8703 TIIII 030324-2048) 3/25/2003 7/00 AM Microsoft Corporation c/windows/system32/udeapi.dll cryp132 5.131,3790.0 (srv03 TIIII 030324-2048) 598.00 KB (612,352 bytes 3/25/2003 7/00 AM Microsoft Corporation c/windows/system32/udeapi.dll mssan 5.2.3790.0 (srv03 TIIII 030324-2048) 5.2.3790.0 (srv03	Srv03_rtm.030324-2048 13.00 KB (13,312 bytes	<u> </u>
ibmhpasv_exe	ibmhpasv.exe c:\windows\system32\ibmhpasv.exe 1120 8 204800 1413120 7/7/2003 2:51 PM 5.1.1.1 14.50 KB (14,848 userenv 5.2 bytes) 6/9/2003 1:38 PM explorer.exe c:\windows\explorer.exe 1420 8 204800 1413120 7/7/2003 2:52 PM 6.00.3790.0 nddeapi 5.2 5.2 5.2 5.2 5.2 6.2 5.3 5.3 5.2 5.3	
204800 1413120 7/7/2003 2:52 PM 5.1.1.1 14.50 KB (14,848 bytes) 6/9/2003 1:38 PM 204800 1413120 7/7/2003 2:52 PM 6.00.3790.0 (srv03.7tm.030324-2048) 1,008.50 KB (1,032,704 bytes) 3/25/2003 7:00 AM Microsoft Corporation ecwindows/system32 lemd.exe 150 180 4 8 Not Available Not Avail	204800 1413120 7/7/2003 2:51 PM 5.1.1.1 14.50 KB (14,848 bytes) 6/9/2003 1:38 PM bytes) 6/9/2003 1:38 PM bytes) 6/9/2003 1:38 PM bytes) 6/9/2003 1:38 PM bytes) 3/2 c:\windows\system32\cmd.exe c:\windows\system32\cmd.exe bytes) 3/2 bytes) 3/2 bytes) 3/2 c:\windows\system32\cmd.exe bytes) 3/2 bytes	<u> </u>
bytes 6/9/2003 1:38 PM	bytes) 6/9/2003 1:38 PM explorer.exe c:\windows\explorer.exe 1420 8 204800 1413120 7/7/2003 2:52 PM 6.00.3790.0 (srv03_rtm.030324-2048) 1,008.50 KB (1,032,704 bytes) 3/25/2003 7:00 AM pronomgr.exe c:\program files\intel\ncs\proset\pronomgr.exe 1480 8 204800 1413120 7/7/2003 2:52 PM 6.2.35.0 84.00 KB (86,016 bytes) 3/11/2003 3:24 PM cri_\kindows\system32\cmd.exe 1596 8 204800 1413120 7/7/2003 2:52 PM 5.2.3790.0 (srv03_rtm.030324-2048) 374.00 KB (382,976 bytes) 3/25/2003 7:00 AM wmiprvse.exe Not Available 1804 8 Not Available Not Available 7/7/2003 2:53 PM Not Available Not Available Not Available 7/7/2003 2:53 PM Not Available Cri_\kindows\system32\cmd.exe 1912 8 204800 1413120 7/7/2003 3:03 PM 5.2.3790.0 (srv03_rtm.030324-2048) 374.00 KB (382,976 bytes) 3/25/2003 7:00 AM wmash 2 204800 2 c:\windows\system32\cmd.exe 1912 8 204800 3 2 204800 3 2 204800 3 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	
204800	204800 1413120 7/7/2003 2:52 PM 6.00.3790.0 (srv03_rtm.030324-2048) 1,008.50 KB (1,032,704 bytes) 3/25/2003 bytes) 3/2 5.0 AM pronomgr.exe c:\program files\intel\ncs\proset\pronomgr.exe c:\program files\intel\ncs\proset\program files\intel\ncs\proset\program files\intel\ncs\proset\program files\intel\ncs\proset\program files\intel\ncs\proset\program files\intel\ncs\proset\program files\intel\ncs\pr	
System 30324-2048 1,008.50 KB (1,032,704 bytes) 3/25/2003 rtm.030324-2048 1,008.50 KB (1,032,704 bytes) 3/25/2003 rtm.030324-2048 5.98.00 KB (86,016 bytes) 3/11/2003 2:52 PM 6.2.35.0	Syr03_rtm.030324-2048 1,008.50 KB (1,032,704 bytes 3/25/2003 7:00 AM pronomgr.exe c:\program files\intel\ncs\proset\pronomgr.exe c:\program files\intel\ncs\proset\program files\intel\ncs\program files\intel\ncs\proset\program files\intel\ncs\proset\progra	
7:00 AM pronomgr.exe chyprogram fileshintel\ncs\proset\pronomgr.exe logologologologologologologologologolog	7:00 AM pronomgr.exe c:\program files\intel\ncs\proset\pronomgr.exe c:\program files\intel\ncs\proset\program files\intel\ncs\proset\program files\intel\ncs\proset\program files\intel\ncs\program file	, =
1480	1480 8 204800 1413120 7/7/2003 2:52 PM 6.2.35.0 bytes) 3/2 84.00 KB (86,016 bytes) 3/11/2003 3:24 PM c:\windows\system32\cmd.exe 1596 8 204800 msasn1 5.2 1413120 7/7/2003 2:52 PM 5.2.3790.0 (srv03_rtm.030324-2048) bytes) 3/2 374.00 KB (382,976 bytes) 3/25/2003 7:00 AM secur32 5.2 Available Not Available Not Available Not Available Poth Available Not Available Not Available Poth Available Poth Available System32\cmd.exe 1912 8 204800 winsta 5.2 1413120 7/7/2003 3:03 PM 5.2.3790.0 (srv03_rtm.030324-2048) bytes) 3/2 374.00 KB (382,976 bytes) 3/25/2003 7:00 AM bytes) 3/2 winsta 5.2 bytes) 3/2 c:\windows\system32\cmd.exe 1912 8 204800 bytes) 3/2 374.00 KB (382,976 bytes) 3/25/2003 7:00 AM c:\windows\system32\cmd.exe c:\windows\system32\cmd.exe bytes) 3/2 bytes) 3/2 bytes) 3/2 bytes) 3/2 c:\windows\system32\cmd.exe c:\windows\system32\cmd.exe <	ı
84.00 KB (86,016 bytes)	84.00 KB (86,016 bytes)	
cmd.exe c:\windows\system32\cmd.exe 1596 8 204800 1413120 7/7/2003 2:52 PM 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM 3/	cmd.exe c:\windows\system32\cmd.exe 1596 8 204800 msasn1 5.2 1413120 7/7/2003 2:52 PM 5.2.3790.0 (srv03_rtm.030324-2048) bytes) 3/2 374.00 KB (382,976 bytes) 3/25/2003 7:00 AM c:\windows\system32\square c:\windows\system32 5.2 Available Not Available Not Available bytes) 3/2 Not Available Not Available c:\windows\system32\cmd.exe 1912 8 204800 winsta 5.2 1413120 7/7/2003 3:03 PM 5.2.3790.0 (srv03_rtm.030324-2048) bytes) 3/2 374.00 KB (382,976 bytes) 3/25/2003 7:00 AM c:\windows\system32\wpabln.exe c:\windows\system32\wpabln.exe 5.2	
374.00 KB (382,976 bytes)	374.00 KB (382,976 bytes) 3/25/2003 7:00 AM c:\windows\sy wmiprvse.exe Not Available 1804 8 Not Available Not Available 7/7/2003 2:53 PM Not Available bytes) 3/2 Not Available Not Available cmd.exe c:\windows\system32\cmd.exe 1912 8 204800 winsta 5.2 1413120 7/7/2003 3:03 PM 5.2.3790.0 (srv03_rtm.030324-2048) bytes) 3/2 374.00 KB (382,976 bytes) 3/25/2003 7:00 AM c:\windows\system32\wpabaln.exe c:\windows\system32\wpabaln.exe 396 8	.3790.0 (srv03_rtm.030324-2048) 58.00 KB (59,392
wmiprvse.exe	wmiprvse.exe Not Available 1804 8 Not secur32 5.2 Available Not Available Not Available Not Available Not Available cmd.exe not Available Not Available c:\windows\system32\cmd.exe 1912 8 204800 winsta 5.2 1413120 7/7/2003 3:03 PM 5.2.3790.0 (srv03_rtm.030324-2048) bytes) 3/2 374.00 KB (382,976 bytes) 3/25/2003 7:00 AM c:\windows\system32\wpabaln.exe 396 8	
Not Available	Not Available Not Available c:\windows\sy cmd.exe c:\windows\system32\cmd.exe 1912 8 204800 winsta 5.2 1413120 7/7/2003 3:03 PM 5.2.3790.0 (srv03_rtm.030324-2048) bytes) 3/2 374.00 KB (382,976 bytes) 3/25/2003 7:00 AM c:\windows\sy wpabaln.exe c:\windows\system32\wpabaln.exe 396 8	
cmd.exe c:\windows\system32\cmd.exe 1912 8 204800	cmd.exe c:\windows\system32\cmd.exe 1912 8 204800 winsta 5.2 1413120 7/7/2003 3:03 PM 5.2.3790.0 (srv03_rtm.030324-2048) bytes) 3/2 374.00 KB (382,976 bytes) 3/25/2003 7:00 AM c:\windows\system32\wpabaln.exe c:\windows\system32 wpabaln.exe c:\windows\system32\wpabaln.exe 396 8 netapi32 5.2	ı
1413120	1413120 7/7/2003 3:03 PM 5.2.3790.0 (srv03_rtm.030324-2048) bytes) 3/2 374.00 KB (382,976 bytes) 3/25/2003 7:00 AM c:\windows\system32\wpabaln.exe c:\windows\system32\wpabaln.exe 396 8 netapi32 5.2	
wpabaln.exe c:\windows\system32\wpabaln.exe 396 8 204800 1413120 7/7/2003 6:54 PM 5.2.3790.0 5.2.3790.0 5/25/2003 7:00 AM 317.00 KB (324,608 bytes) 317.00 KB (324,608 bytes) 325/2003 7:00 AM 317.00 KB (324,608 bytes) 325/2003 7:00 AM 325/2003 7:00 AM Microsoft Corporation c:\windows\system32\netapi32.dll c:\windows\system32\netapi32.dll profmap 5.2.3790.0 (srv03_rtm.030324-2048) 22.00 KB (22,528 bytes) 22.00 KB (22,528 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\profmap.dll c:\windows\system32\profmap.dll regapi 5.2.3790.0 (srv03_rtm.030324-2048) 22.00 KB (22,528 bytes) 6/9/2003 1:08 PM 5.2.3790.0 (srv03_rtm.030324-2048) 22.00 KB (22,528 bytes) 6/9/2003 1:08 PM 5.2.3790.0 (srv03_rtm.030324-2048) 22.00 KB (22,528 bytes) 6/9/2003 1:08 PM c:\windows\system32\profmap.dll regapi 5.2.3790.0 (srv03_rtm.030324-2048) 48.50 KB (49,664 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\profmap.dll ws2_32 5.2.3790.0 (srv03_rtm.030324-2048) 87.50 KB (89,600 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\profmap.dll ws2_32 5.2.3790.0 (srv03_rtm.030324-2048) 8	wpabaln.exe c:\windows\system32\wpabaln.exe 396 8 netapi32 5.2	
204800 1413120 7/7/2003 6:54 PM 5.2.3790.0 (srv03_rtm.030324-2048) 31.00 KB (31,744 bytes) 3/25/2003 7:00 AM	1 1	
C:\windows\system32\netapi32.dll profmap 5.2.3790.0 (srv03_rtm.030324-2048) 31.00 KB (31,744 bytes) 3/25/2003 3/25	- ANTONO - 191 1 1 ANT - 1 ANT	, ,
sqlservr.exe c:\program files\microsoft sql bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\profmap.dll regapi 5.2.3790.0 (srv03_rtm.030324-2048) 48.50 KB (49,664 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\profmap.dll regapi 5.2.3790.0 (srv03_rtm.030324-2048) 48.50 KB (49,664 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\profmap.dll regapi 5.2.3790.0 (srv03_rtm.030324-2048) 48.50 KB (49,664 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\profmap.dll regapi 5.2.3790.0 (srv03_rtm.030324-2048) 87.50 KB (89,600 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\profmap.dll regapi 5.2.3790.0 (srv03_rtm.030324-2048) 87.50 KB (89,600 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\profmap.dll regapi 5.2.3790.0 (srv03_rtm.030324-2048) 87.50 KB (89,600 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\profmap.dll regapi 5.2.3790.0 (srv03_rtm.030324-2048) 87.50 KB (89,600 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\profmap.dll regapi 5.2.3790.0 (srv03_rtm.030324-2048) 87.50 KB (89,600 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\profmap.dll regapi 5.2.3790.0 (srv03_rtm.030324-2048) 87.50 KB (89,600 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\profmap.dll regapi 5.2.3790.0 (srv03_rtm.030324-2048) 87.50 KB (89,600 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\profmap.dll regapi 5.2.3790.0 (srv03_rtm.030324-2048) 87.50 KB (89,600 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\profmap.dl regapi 5.2.3790.0 (srv03_rtm.030324-2048) 87.50 KB (89,600 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\profmap.dl regapi 5.2.3790.0 (srv03_rtm.030324-2048) 87.50 KB (89,600 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\profmap.dl regapi 5.2.3790.0 (srv03_rtm.030324-2048) 87.50 KB (89,600 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\profma	(srv03_rtm.030324-2048) 31.00 KB (31,744 bytes) 3/25/2003 c:\windows\sy	<u> </u>
server\mssql\binn\sqlservr.exe 1256 13 204800 1413120 7/7/2003 7:35 PM 2000.080.0760.00 7.17 MB (7,520,337 bytes) 204800 1413120 7.17 MB (7,520,337 bytes) 5.2.3790.0 (srv03_rtm.030324-2048) 48.50 KB (49,664 bytes) 3/25/2003 7:00 AM Microsoft Corporation 5.2.3790.0 (srv03_rtm.030324-2048) 48.50 KB (49,664 bytes) 3/25/2003 7:00 AM Microsoft Corporation 5.2.3790.0 (srv03_rtm.030324-2048) 87.50 KB (89,600 bytes) 3/25/2003 7:00 AM Microsoft Corporation 5.2.3790.0 (srv03_rtm.030324-2048) 87.50 KB (89,600 bytes) 3/25/2003 7:00 AM Microsoft Corporation 5.2.3790.0 (srv03_rtm.030324-2048) 1 1		
7/7/2003 7:35 PM		
isqlw.exe c:\program files\microsoft sql server\80\tools\binn\isqlw.exe	7/7/2003 7:35 PM 2000.080.0760.00 7.17 MB (7,520,337 bytes) regapi 5.2	.3790.0 (srv03_rtm.030324-2048) 48.50 KB (49,664
1752 8 204800 1413120 7/7/2003 7:35 PM 2000.080.0760.00 344.56 KB (352,828 bytes) 6/9/2003 1:08 PM helpctr.exe c:\windows\pchealth\helpctr\binaries\helpctr.exe c 1552 8 ws2_32 5.2.3790.0 (srv03_rtm.030324-2048) 87.50 KB (89,600 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ws2_32.dll		1
helpctr.exe c:\windows\pchealth\helpctr\binaries\helpctr.exe 1552 8 c:\windows\system32\ws2_32.dll		
		C 1
		.3790.0 (srv03_rtm.030324-2048) 87.50 KB (89,600 5/2003 7:00 AM Microsoft Corporation
(srv03_rtm.030324-2048) 764.00 KB (782,336 bytes) 6/9/2003 bytes) 3/25/2003 7:00 AM Microsoft Corporation	(srv03_rtm.030324-2048) 764.00 KB (782,336 bytes) 6/9/2003 bytes) 3/2	.3790.0 (srv03_rtm.030324-2048) 87.50 KB (89,600 5/2003 7:00 AM Microsoft Corporation stem32\ws2_32.dll
12:46 PM c:\windows\system32\ws2help.dll		.3790.0 (srv03_rtm.030324-2048) 87.50 KB (89,600 5/2003 7:00 AM Microsoft Corporation stem32\ws2_32.dll .3790.0 (srv03_rtm.030324-2048) 19.50 KB (19,968 5/2003 7:00 AM Microsoft Corporation
wmiprvse.exe Not Available 1588 8 Not psapi 5.2.3790.0 (srv03_rtm.030324-2048) 21.50 KB (22,016 Available Not Available 7/7/2003 7:41 PM Not Available bytes) 3/25/2003 7:00 AM Microsoft Corporation	•	.3790.0 (srv03_rtm.030324-2048) 87.50 KB (89,600 5/2003 7:00 AM Microsoft Corporation stem32\ws2_32.dll .3790.0 (srv03_rtm.030324-2048) 19.50 KB (19,968 5/2003 7:00 AM Microsoft Corporation stem32\ws2help.dll
	Not Available Not Available c:\windows\sy	.3790.0 (srv03_rtm.030324-2048) 87.50 KB (89,600 5/2003 7:00 AM Microsoft Corporation stem32\ws2_32.dll .3790.0 (srv03_rtm.030324-2048) 19.50 KB (19,968 5/2003 7:00 AM Microsoft Corporation stem32\ws2help.dll .3790.0 (srv03_rtm.030324-2048) 21.50 KB (22,016

5.2.3790.0 (srv03 rtm.030324-2048) comctl32 5.82 (srv03 rtm.030324-2048) 561.00 KB (574,464 bytes) version 17.00 KB (17,408 6/9/2003 8:24 AM bytes) 3/25/2003 7:00 AM Microsoft Corporation Microsoft Corporation c:\windows\system32\version.dll c:\windows\winsxs\x86 microsoft.windows.common-controls 6595b64144ccf1 setupapi 5.2.3790.0 (srv03_rtm.030324-2048) $df_5.82.0.0_x-ww_8a69\overline{b}a05\backslash comctl32.dll$ 1,014.50 KB (1.038.848 bytes) 3/25/2003 7:00 AM Microsoft Corporation uxtheme 6.00.3790.0 (srv03 rtm.030324-2048) 196.00 KB (200,704 c:\windows\system32\setupapi.dll 3/25/2003 7:00 AM Microsoft Corporation bytes) 5.2.3790.0 (srv03 rtm.030324-2048) msgina 1.14 MB (1,191,936 c:\windows\system32\uxtheme.dll 5.2.3790.0 (srv03 rtm.030324-2048) bytes) 3/25/2003 7:00 AM Microsoft Corporation samlib 49.00 KB (50,176 $c:\ \ windows\ \ system 32\ \ msgina.dll$ bytes) 3/25/2003 7:00 AM Microsoft Corporation 6.00.3790.0 (srv03 rtm.030324-2048) 121.50 KB (124,416 c:\windows\system32\samlib.dll shsvcs 5.2.3790.0 (srv03_rtm.030324-2048) bytes) 3/25/2003 7:00 AM Microsoft Corporation mprapi 81.00 KB (82,944 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\shsvcs.dll bytes) shlwapi 6.00.3790.0 (srv03 rtm.030324-2048) 281.00 KB (287,744 c:\windows\system32\mprapi.dll 3/25/2003 7:00 AM Microsoft Corporation 5.2.3790.0 (srv03_rtm.030324-2048) 189.00 KB (193,536 bytes) activeds c:\windows\system32\shlwapi.dll 3/25/2003 7:00 AM Microsoft Corporation bytes) sfc 5.2.3790.0 (srv03 rtm.030324-2048) 4.50 KB (4,608 bytes) c:\windows\system32\activeds.dll 3/25/2003 7:00 AM Microsoft Corporation adsldpc 5.2.3790.0 (srv03 rtm.030324-2048) 142.50 KB (145,920 c:\windows\system32\sfc.dll bytes) 3/25/2003 7:00 AM Microsoft Corporation 5.2.3790.0 (srv03 rtm.030324-2048) 133.00 KB (136,192 c:\windows\system32\adsldpc.dll sfc os bytes) 3/25/2003 7:00 AM Microsoft Corporation credui 5.2.3790.0 (srv03 rtm.030324-2048) 159.00 KB (162,816 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\sfc os.dll bytes) 5.131.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation 161.50 KB (165,376 c:\windows\system32\credui.dll wintrust bytes) atl 3.05.2283 83.00 KB (84,992 bytes) 3/25/2003 7:00 AM c:\windows\system32\wintrust.dll Microsoft Corporation c:\windows\system32\atl.dll ole32 5.2.3790.0 (srv03_rtm.030324-2048) 1.13 MB (1,187,328 oleaut32 5.2.3790.0 486.00 KB (497,664 bytes) 3/25/2003 7:00 AM 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\oleaut32.dll Microsoft Corporation bytes) c:\windows\system32\ole32.dll rtutils 5.2.3790.0 (srv03 rtm.030324-2048) 32.00 KB (32,768 imagehlp 5.2.3790.0 (srv03_rtm.030324-2048) 3/25/2003 7:00 AM Microsoft Corporation 142.50 KB (145,920 bytes) bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rtutils.dll 5.2.3790.0 (srv03_rtm.030324-2048) 305.00 KB (312,320 c:\windows\system32\imagehlp.dll cscui comctl32 6.0 (srv03_rtm.030324-2048) 907.00 KB (928,768 bytes) bytes) 3/25/2003 7:00 AM Microsoft Corporation 6/9/2003 8:24 AM Microsoft Corporation c:\windows\system32\cscui.dll c:\windows\winsxs\x86_microsoft.windows.common-controls_6595b64144ccfl 2001.12.4720.0 (srv03 rtm.030324-2048) 481.00 KB (492,544 clbcata df 6.0.100.0 x-ww 8417450b\comctl32.dll 6/9/2003 12:42 PM Microsoft Corporation bytes) winscard 5.2.3790.0 (srv03_rtm.030324-2048) 98.50 KB (100,864 c:\windows\system32\clbcatq.dll 3/25/2003 7:00 AM Microsoft Corporation 2001.12.4720.0 (srv03 rtm.030324-2048) 778.00 KB (796,672 bytes) comres c:\windows\system32\winscard.dll 3/25/2003 7:00 AM Microsoft Corporation bytes) wtsapi32 5.2.3790.0 (srv03 rtm.030324-2048) c:\windows\system32\comres.dll 17.50 KB (17,920 3/25/2003 7:00 AM Microsoft Corporation 5.2.3790.0 (srv03_rtm.030324-2048) bytes) ntmarta 114.00 KB (116,736 c:\windows\system32\wtsapi32.dll bytes) 3/25/2003 7:00 AM Microsoft Corporation 5.2.3790.0 (srv03 rtm.030324-2048) 166.00 KB (169,984 c:\windows\system32\ntmarta.dll 3/25/2003 7:00 AM Microsoft Corporation wbemprox 5.2.3790.0 (srv03_rtm.030324-2048) 17.50 KB (17,920 bytes) c:\windows\system32\winmm.dll 6/9/2003 12:42 PM Microsoft Corporation bytes) 5.2.3790.0 (srv03_rtm.030324-2048) SXS 733.00 KB (750,592 c:\windows\system32\wbem\wbemprox.dll 3/25/2003 7:00 AM Microsoft Corporation 5.2.3790.0 (srv03 rtm.030324-2048) bytes) wbemcomn 211.50 KB 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\sxs.dll (216,576 bytes) shell32 6.00.3790.0 (srv03 rtm.030324-2048) 7.79 MB (8,166,400 c:\windows\system32\wbem\wbemcomn.dll bytes) 3/25/2003 7:00 AM Microsoft Corporation wbemsvc 5.2.3790.0 (srv03 rtm.030324-2048) 42.50 KB (43,520 6/9/2003 12:42 PM Microsoft Corporation c:\windows\system32\shell32.dll bytes) 5.2.3790.0 (srv03 rtm.030324-2048) 158.00 KB (161,792 c:\windows\system32\wbem\wbemsvc.dll wldap32 3/25/2003 7:00 AM Microsoft Corporation fastprox 5.2.3790.0 (srv03_rtm.030324-2048) bytes) 443.00 KB (453,632 c:\windows\system32\wldap32.dll 6/9/2003 12:42 PM Microsoft Corporation bytes) 5.2.3790.0 (srv03_rtm.030324-2048) 99.00 KB (101,376 c:\windows\system32\wbem\fastprox.dll cscdll bytes) 3/25/2003 7:00 AM Microsoft Corporation msvcp60 6.05.2144.0 388.00 KB (397,312 bytes) 3/25/2003 c:\windows\system32\cscdll.dll 7:00 AM Microsoft Corporation c:\windows\system32\msvcp60.dll 5.2.3790.0 (srv03 rtm.030324-2048) 87.50 KB (89,600 5.2.3790.0 (srv03 rtm.030324-2048) wlnotify ntdsapi 76.00 KB (77,824 3/25/2003 7:00 AM Microsoft Corporation 3/25/2003 7:00 AM Microsoft Corporation bytes) bytes) c:\windows\system32\wlnotify.dll c:\windows\system32\ntdsapi.dll winspool 5.2.3790.0 (srv03 rtm.030324-2048) 5.2.3790.0 (srv03 rtm.030324-2048) 140.00 KB (143,360 dnsapi 147.50 KB (151,040 3/25/2003 7:00 AM Microsoft Corporation bytes) 3/25/2003 7:00 AM Microsoft Corporation bytes) c:\windows\system32\winspool.drv c:\windows\system32\dnsapi.dll 5.2.3790.0 (srv03 rtm.030324-2048) 56.00 KB (57,344 services 5.2.3790.0 (srv03 rtm.030324-2048) 102.00 KB (104,448 mpr bytes) 3/25/2003 7:00 AM Microsoft Corporation 3/25/2003 7:00 AM Microsoft Corporation bytes) c:\windows\system32\mpr.dll c:\windows\system32\services.exe 5.2.3790.0 (srv03 rtm.030324-2048) 5.2.3790.0 (srv03_rtm.030324-2048) 316.50 KB (324,096 rsaenh 176.83 KB (181,072 scesry bytes) 3/25/2003 7:00 AM Microsoft Corporation bytes) 3/25/2003 7:00 AM Microsoft Corporation

c:\windows\system32\scesrv.dll

c:\windows\system32\rsaenh.dll

	•
authz 5.2.3790.0 (srv03_rtm.030324-2048) 67.00 KB (68,608	scecli 5.2.3790.0 (srv03_rtm.030324-2048) 179.50 KB (183,808
bytes) 3/25/2003 7:00 AM Microsoft Corporation	bytes) 3/25/2003 7:00 AM Microsoft Corporation
c:\windows\system32\authz.dll	c:\windows\system32\scecli.dll
•	
umpnpmgr 5.2.3790.0 (srv03_rtm.030324-2048) 121.50 KB (124,416	wshtcpip 5.2.3790.0 (srv03_rtm.030324-2048) 18.00 KB (18,432
bytes) 3/25/2003 7:00 AM Microsoft Corporation	bytes) 3/25/2003 7:00 AM Microsoft Corporation
c:\windows\system32\umpnpmgr.dll	c:\windows\system32\wshtcpip.dll
ncobjapi 5.2.3790.0 (srv03 rtm.030324-2048) 34.50 KB (35,328	pstorsvc 5.2.3790.0 (srv03 rtm.030324-2048) 24.00 KB (24,576
71	1
bytes) 3/25/2003 7:00 AM Microsoft Corporation	bytes) 3/25/2003 7:00 AM Microsoft Corporation
c:\windows\system32\ncobjapi.dll	c:\windows\system32\pstorsvc.dll
eventlog 5.2.3790.0 (srv03 rtm.030324-2048) 60.50 KB (61,952	psbase 5.2.3790.0 (srv03 rtm.030324-2048) 81.00 KB (82,944
bytes) 3/25/2003 7:00 AM Microsoft Corporation	bytes) 3/25/2003 7:00 AM Microsoft Corporation
c:\windows\system32\eventlog.dll	c:\windows\system32\psbase.dll
lsass 5.2.3790.0 (srv03_rtm.030324-2048) 13.00 KB (13,312	dssenh 5.2.3790.0 (srv03_rtm.030324-2048) 131.33 KB (134,480
bytes) 3/25/2003 7:00 AM Microsoft Corporation	bytes) 3/25/2003 7:00 AM Microsoft Corporation
c:\windows\system32\lsass.exe	c:\windows\system32\dssenh.dll
lsasrv 5.2.3790.0 (srv03 rtm.030324-2048) 780.50 KB (799,232	svchost 5.2.3790.0 (srv03 rtm.030324-2048) 13.00 KB (13,312
` = ' · · · · · · · · · · · · · · · · · ·	· =
bytes) 3/25/2003 7:00 AM Microsoft Corporation	bytes) 3/25/2003 7:00 AM Microsoft Corporation
c:\windows\system32\lsasrv.dll	c:\windows\system32\svchost.exe
samsrv 5.2.3790.0 (srv03 rtm.030324-2048) 452.00 KB (462,848	rpcss 5.2.3790.0 (srv03 rtm.030324-2048) 276.50 KB (283,136
bytes) 3/25/2003 7:00 AM Microsoft Corporation	bytes) 3/25/2003 7:00 AM Microsoft Corporation
c:\windows\system32\samsrv.dll	c:\windows\system32\rpcss.dll
· · · · · · · · · · · · · · · · · · ·	
cryptdll 5.2.3790.0 (srv03_rtm.030324-2048) 34.00 KB (34,816	termsrv 5.2.3790.0 (srv03_rtm.030324-2048) 216.50 KB (221,696
bytes) 3/25/2003 7:00 AM Microsoft Corporation	bytes) 6/9/2003 12:42 PM Microsoft Corporation
c:\windows\system32\cryptdll.dll	c:\windows\system32\termsrv.dll
msprivs 5.2.3790.0 (srv03 rtm.030324-2048) 46.50 KB (47,616	icaapi 5.2.3790.0 (srv03 rtm.030324-2048) 10.50 KB (10,752
bytes) 3/25/2003 7:00 AM Microsoft Corporation	bytes) 6/9/2003 12:42 PM Microsoft Corporation
c:\windows\system32\msprivs.dll	c:\windows\system32\icaapi.dll
kerberos 5.2.3790.0 (srv03 rtm.030324-2048) 332.50 KB (340,480	mstlsapi 5.2.3790.0 (srv03_rtm.030324-2048) 104.50 KB (107,008
bytes) 3/25/2003 7:00 AM Microsoft Corporation	bytes) 3/25/2003 7:00 AM Microsoft Corporation
• •	
c:\windows\system32\kerberos.dll	c:\windows\system32\mstlsapi.dll
msv1_0 5.2.3790.0 (srv03_rtm.030324-2048) 127.00 KB (130,048	audiosrv 5.2.3790.0 (srv03_rtm.030324-2048) 38.00 KB (38,912
bytes) 3/25/2003 7:00 AM Microsoft Corporation	bytes) 3/25/2003 7:00 AM Microsoft Corporation
c:\windows\system32\msv1_0.dll	c:\windows\system32\audiosrv.dll
netlogon 5.2.3790.0 (srv03 rtm.030324-2048) 409.00 KB (418,816	wkssvc 5.2.3790.0 (srv03 rtm.030324-2048) 125.00 KB (128,000
· = /	/
bytes) 3/25/2003 7:00 AM Microsoft Corporation	bytes) 3/25/2003 7:00 AM Microsoft Corporation
c:\windows\system32\netlogon.dll	c:\windows\system32\wkssvc.dll
w32time 5.2.3790.0 (srv03_rtm.030324-2048) 216.00 KB (221,184	wiarpc 5.2.3790.0 (srv03 rtm.030324-2048) 30.00 KB (30,720
bytes) 3/25/2003 7:00 AM Microsoft Corporation	bytes) 3/25/2003 7:00 AM Microsoft Corporation
	c:\windows\system32\wiarpc.dll
as/verim days: a vertam 20/ver20 time a dll	
c:\windows\system32\w32time.dll	1
iphlpapi 5.2.3790.0 (srv03_rtm.030324-2048) 82.50 KB (84,480	dmserver 5.2.3790.0 (srv03_rtm.030324-2048) 24.00 KB (24,576
· · · · · · · · · · · · · · · · · · ·	1
iphlpapi 5.2.3790.0 (srv03_rtm.030324-2048) 82.50 KB (84,480 bytes) 3/25/2003 7:00 AM Microsoft Corporation	dmserver 5.2.3790.0 (srv03_rtm.030324-2048) 24.00 KB (24,576 bytes) 3/25/2003 7:00 AM Microsoft Corporation
iphlpapi 5.2.3790.0 (srv03_rtm.030324-2048) 82.50 KB (84,480 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\iphlpapi.dll	dmserver 5.2.3790.0 (srv03_rtm.030324-2048) 24.00 KB (24,576 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\dmserver.dll
iphlpapi 5.2.3790.0 (srv03_rtm.030324-2048) 82.50 KB (84,480 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\iphlpapi.dll schannel 5.2.3790.0 (srv03_rtm.030324-2048) 149.50 KB (153,088	dmserver 5.2.3790.0 (srv03_rtm.030324-2048) 24.00 KB (24,576 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\dmserver.dll srvsvc 5.2.3790.0 (srv03_rtm.030324-2048) 89.00 KB (91,136
iphlpapi 5.2.3790.0 (srv03_rtm.030324-2048) 82.50 KB (84,480 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\iphlpapi.dll schannel 5.2.3790.0 (srv03_rtm.030324-2048) 149.50 KB (153,088 bytes) 3/25/2003 7:00 AM Microsoft Corporation	dmserver 5.2.3790.0 (srv03_rtm.030324-2048) 24.00 KB (24,576 bytes) bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\dmserver.dll 89.00 KB (91,136 bytes) 3/25/2003 7:00 AM Microsoft Corporation
iphlpapi 5.2.3790.0 (srv03_rtm.030324-2048) 82.50 KB (84,480 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\iphlpapi.dll schannel 5.2.3790.0 (srv03_rtm.030324-2048) 149.50 KB (153,088 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\schannel.dll	dmserver 5.2.3790.0 (srv03_rtm.030324-2048) 24.00 KB (24,576 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\dmserver.dll srvsvc 5.2.3790.0 (srv03_rtm.030324-2048) 89.00 KB (91,136
iphlpapi 5.2.3790.0 (srv03_rtm.030324-2048) 82.50 KB (84,480 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\iphlpapi.dll schannel 5.2.3790.0 (srv03_rtm.030324-2048) 149.50 KB (153,088 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\schannel.dll	dmserver 5.2.3790.0 (srv03_rtm.030324-2048) 24.00 KB (24,576 bytes) bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\dmserver.dll 89.00 KB (91,136 bytes) 3/25/2003 7:00 AM Microsoft Corporation
iphlpapi 5.2.3790.0 (srv03_rtm.030324-2048) 82.50 KB (84,480 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\iphlpapi.dll schannel 5.2.3790.0 (srv03_rtm.030324-2048) 149.50 KB (153,088 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\schannel.dll wdigest 5.2.3790.0 (srv03_rtm.030324-2048) 61.00 KB (62,464	dmserver 5.2.3790.0 (srv03_rtm.030324-2048) 24.00 KB (24,576 bytes) bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\dmserver.dll 89.00 KB (91,136 bytes) 3/25/2003 7:00 AM Microsoft Corporation 89.00 KB (91,136 bytes) c:\windows\system32\srvsvc.dll 16.50 KB (16,896 bytes)
iphlpapi 5.2.3790.0 (srv03_rtm.030324-2048) 82.50 KB (84,480 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\iphlpapi.dll schannel 5.2.3790.0 (srv03_rtm.030324-2048) 149.50 KB (153,088 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\schannel.dll wdigest 5.2.3790.0 (srv03_rtm.030324-2048) 61.00 KB (62,464 bytes) 3/25/2003 7:00 AM Microsoft Corporation	dmserver 5.2.3790.0 (srv03_rtm.030324-2048) 24.00 KB (24,576 bytes) bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\dmserver.dll 89.00 KB (91,136 bytes) 3/25/2003 7:00 AM Microsoft Corporation 89.00 KB (91,136 bytes) c:\windows\system32\srvsvc.dll 16.50 KB (16,896 bytes) 3/25/2003 7:00 AM Microsoft Corporation
iphlpapi 5.2.3790.0 (srv03_rtm.030324-2048) 82.50 KB (84,480 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\iphlpapi.dll schannel 5.2.3790.0 (srv03_rtm.030324-2048) 149.50 KB (153,088 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\schannel.dll wdigest 5.2.3790.0 (srv03_rtm.030324-2048) 61.00 KB (62,464 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\schannel.dll circle windows\system32\schannel.dll circle windows\system32\schannel.dll circle windows\system32\schannel.dll circle windows\system32\schannel.dll	dmserver 5.2.3790.0 (srv03_rtm.030324-2048) 24.00 KB (24,576 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\dmserver.dll srvsvc 5.2.3790.0 (srv03_rtm.030324-2048) 89.00 KB (91,136 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\srvsvc.dll seclogon 5.2.3790.0 (srv03_rtm.030324-2048) 16.50 KB (16,896 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\seclogon.dll
iphlpapi 5.2.3790.0 (srv03_rtm.030324-2048) 82.50 KB (84,480 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\iphlpapi.dll schannel 5.2.3790.0 (srv03_rtm.030324-2048) 149.50 KB (153,088 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\schannel.dll wdigest 5.2.3790.0 (srv03_rtm.030324-2048) 61.00 KB (62,464 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\wdigest.dll rassfm 5.2.3790.0 (srv03_rtm.030324-2048) 20.50 KB (20,992	dmserver 5.2.3790.0 (srv03_rtm.030324-2048) 24.00 KB (24,576 bytes) bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\dmserver.dll 89.00 KB (91,136 bytes) 5.2.3790.0 (srv03_rtm.030324-2048) 89.00 KB (91,136 bytes) 6.2003 3/25/2003 7:00 AM 7.2003 3/25/2003 7:00 AM 89.00 KB (91,136 bytes) 16.50 KB (16,896 bytes) 89.00 KB (91,136 bytes) 3/25/2003 7:00 AM 89.00 KB (16,896 bytes)
iphlpapi 5.2.3790.0 (srv03_rtm.030324-2048) 82.50 KB (84,480 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\iphlpapi.dll schannel 5.2.3790.0 (srv03_rtm.030324-2048) 149.50 KB (153,088 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\schannel.dll wdigest 5.2.3790.0 (srv03_rtm.030324-2048) 61.00 KB (62,464 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\schannel.dll circle windows\system32\schannel.dll circle windows\system32\schannel.dll circle windows\system32\schannel.dll circle windows\system32\schannel.dll	dmserver 5.2.3790.0 (srv03_rtm.030324-2048) 24.00 KB (24,576 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\dmserver.dll srvsvc 5.2.3790.0 (srv03_rtm.030324-2048) 89.00 KB (91,136 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\srvsvc.dll seclogon 5.2.3790.0 (srv03_rtm.030324-2048) 16.50 KB (16,896 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\seclogon.dll
iphlpapi 5.2.3790.0 (srv03_rtm.030324-2048) 82.50 KB (84,480 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\iphlpapi.dll schannel 5.2.3790.0 (srv03_rtm.030324-2048) 149.50 KB (153,088 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\schannel.dll wdigest 5.2.3790.0 (srv03_rtm.030324-2048) 61.00 KB (62,464 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\wdigest.dll rassfm 5.2.3790.0 (srv03_rtm.030324-2048) 20.50 KB (20,992 bytes) 3/25/2003 7:00 AM Microsoft Corporation	dmserver 5.2.3790.0 (srv03_rtm.030324-2048) 24.00 KB (24,576 bytes) bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\dmserver.dll 89.00 KB (91,136 bytes) svsvc 5.2.3790.0 (srv03_rtm.030324-2048) 89.00 KB (91,136 bytes) c:\windows\system32\srvsvc.dll seclogon 5.2.3790.0 (srv03_rtm.030324-2048) 16.50 KB (16,896 bytes) bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\seclogon.dll wmisvc 5.2.3790.0 (srv03_rtm.030324-2048) 131.00 KB (134,144 bytes) 6/9/2003 12:42 PM Microsoft Corporation
iphlpapi 5.2.3790.0 (srv03_rtm.030324-2048) 82.50 KB (84,480 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\iphlpapi.dll schannel 5.2.3790.0 (srv03_rtm.030324-2048) 149.50 KB (153,088 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\schannel.dll wdigest 5.2.3790.0 (srv03_rtm.030324-2048) 61.00 KB (62,464 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\wdigest.dll rassfm 5.2.3790.0 (srv03_rtm.030324-2048) 20.50 KB (20,992 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ransfm.dll	dmserver 5.2.3790.0 (srv03_rtm.030324-2048) 24.00 KB (24,576 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\dmserver.dll srvsvc 5.2.3790.0 (srv03_rtm.030324-2048) 89.00 KB (91,136 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\srvsvc.dll seclogon 5.2.3790.0 (srv03_rtm.030324-2048) 16.50 KB (16,896 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\seclogon.dll wmisvc 5.2.3790.0 (srv03_rtm.030324-2048) 131.00 KB (134,144 bytes) 6/9/2003 12:42 PM Microsoft Corporation c:\windows\system32\swbem\wmisvc.dll
iphlpapi 5.2.3790.0 (srv03_rtm.030324-2048) 82.50 KB (84,480 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\iphlpapi.dll schannel 5.2.3790.0 (srv03_rtm.030324-2048) 149.50 KB (153,088 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\schannel.dll wdigest 5.2.3790.0 (srv03_rtm.030324-2048) 61.00 KB (62,464 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\wdigest.dll rassfm 5.2.3790.0 (srv03_rtm.030324-2048) 20.50 KB (20,992 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ransfm 5.2.3790.0 (srv03_rtm.030324-2048) 20.50 KB (20,992 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ransfm.dll kdcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 221.00 KB (226,304	dmserver 5.2.3790.0 (srv03_rtm.030324-2048) 24.00 KB (24,576 bytes) bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\dmserver.dll 89.00 KB (91,136 bytes) 89.00 KB (91,136 bytes) bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\svevvc.dll 16.50 KB (16,896 bytes) bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\seclogon.dll 131.00 KB (134,144 bytes) 6/9/2003 12:42 PM Microsoft Corporation c:\windows\system32\seclogon.dll 131.00 KB (134,144 bytes) 6/9/2003 12:42 PM Microsoft Corporation c:\windows\system32
iphlpapi 5.2.3790.0 (srv03_rtm.030324-2048) 82.50 KB (84,480 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\iphlpapi.dll schannel 5.2.3790.0 (srv03_rtm.030324-2048) 149.50 KB (153,088 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\schannel.dll wdigest 5.2.3790.0 (srv03_rtm.030324-2048) 61.00 KB (62,464 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\wdigest.dll rassfm 5.2.3790.0 (srv03_rtm.030324-2048) 20.50 KB (20,992 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rassfm.dll kdcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 221.00 KB (226,304 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rassfm.dll kdcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 221.00 KB (226,304 bytes) 3/25/2003 7:00 AM Microsoft Corporation	dmserver 5.2.3790.0 (srv03_rtm.030324-2048) 24.00 KB (24,576 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\dmserver.dll srvsvc 5.2.3790.0 (srv03_rtm.030324-2048) 89.00 KB (91,136 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\system
iphlpapi 5.2.3790.0 (srv03_rtm.030324-2048) 82.50 KB (84,480 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\iphlpapi.dll schannel 5.2.3790.0 (srv03_rtm.030324-2048) 149.50 KB (153,088 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\schannel.dll wdigest 5.2.3790.0 (srv03_rtm.030324-2048) 61.00 KB (62,464 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\wdigest.dll rassfm 5.2.3790.0 (srv03_rtm.030324-2048) 20.50 KB (20,992 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ransfm 5.2.3790.0 (srv03_rtm.030324-2048) 20.50 KB (20,992 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ransfm.dll kdcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 221.00 KB (226,304	dmserver 5.2.3790.0 (srv03_rtm.030324-2048) 24.00 KB (24,576 bytes) bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\dmserver.dll 89.00 KB (91,136 bytes) 89.00 KB (91,136 bytes) bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\svevvc.dll 16.50 KB (16,896 bytes) bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\seclogon.dll 131.00 KB (134,144 bytes) 6/9/2003 12:42 PM Microsoft Corporation c:\windows\system32\seclogon.dll 131.00 KB (134,144 bytes) 6/9/2003 12:42 PM Microsoft Corporation c:\windows\system32
iphlpapi 5.2.3790.0 (srv03_rtm.030324-2048) 82.50 KB (84,480 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\iphlpapi.dll schannel 5.2.3790.0 (srv03_rtm.030324-2048) 149.50 KB (153,088 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\schannel.dll wdigest 5.2.3790.0 (srv03_rtm.030324-2048) 61.00 KB (62,464 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\wdigest.dll rassfm 5.2.3790.0 (srv03_rtm.030324-2048) 20.50 KB (20,992 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rangle\tag{vto} 3/25/2003 7:00 AM Microsoft Corporati	dmserver 5.2.3790.0 (srv03_rtm.030324-2048) 24.00 KB (24,576 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\dmserver.dll srvsvc 5.2.3790.0 (srv03_rtm.030324-2048) 89.00 KB (91,136 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\srvsvc.dll seclogon 5.2.3790.0 (srv03_rtm.030324-2048) 16.50 KB (16,896 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\seclogon.dll wmisvc 5.2.3790.0 (srv03_rtm.030324-2048) 131.00 KB (134,144 bytes) 6/9/2003 12:42 PM Microsoft Corporation c:\windows\system32\svenaptheta bytes) 6/9/2003 12:42 PM Microsoft Corporation c:\windows\system32\wbem\wmisvc.dll vssapi 5.2.3790.0 (srv03_rtm.030324-2048) 528.00 KB (540,672 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\vssapi.dll
iphlpapi 5.2.3790.0 (srv03_rtm.030324-2048) 82.50 KB (84,480 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\iphlpapi.dll schannel 5.2.3790.0 (srv03_rtm.030324-2048) 149.50 KB (153,088 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\schannel.dll wdigest 5.2.3790.0 (srv03_rtm.030324-2048) 61.00 KB (62,464 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\wdigest.dll rassfm 5.2.3790.0 (srv03_rtm.030324-2048) 20.50 KB (20,992 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rangle\tagsfm.dll kdcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 221.00 KB (226,304 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rangle\tagsfm.dll kdcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 221.00 KB (226,304 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rangle\tagsfm.dll kdcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 21.00 KB (226,304 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rangle\tagsfm.dll kdcsvc dll ntdsa 5.2.3790.0 (srv03_rtm.030324-2048) 1.45 MB (1,520,640	dmserver 5.2.3790.0 (srv03_rtm.030324-2048) 24.00 KB (24,576 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\dmserver.dll srvsvc 5.2.3790.0 (srv03_rtm.030324-2048) 89.00 KB (91,136 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\srvsvc.dll seclogon 5.2.3790.0 (srv03_rtm.030324-2048) 16.50 KB (16,896 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\seclogon.dll wmisvc 5.2.3790.0 (srv03_rtm.030324-2048) 131.00 KB (134,144 bytes) 6/9/2003 12:42 PM Microsoft Corporation c:\windows\system32\swbem\wmisvc.dll vsapi 5.2.3790.0 (srv03_rtm.030324-2048) 528.00 KB (540,672 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\system332\system32\system32\system32\system32\system32\system32\system332\system32\system32\system32\system32\system32\system32\system332\system32\system332\syst
iphlpapi 5.2.3790.0 (srv03_rtm.030324-2048) 82.50 KB (84,480 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\iphlpapi.dll schannel 5.2.3790.0 (srv03_rtm.030324-2048) 149.50 KB (153,088 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\schannel.dll wdigest 5.2.3790.0 (srv03_rtm.030324-2048) 61.00 KB (62,464 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\wdigest.dll rassfm 5.2.3790.0 (srv03_rtm.030324-2048) 20.50 KB (20,992 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rangle\tagsfm.dll kdcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 221.00 KB (226,304 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rangle\tagsfm.dll kdcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 221.00 KB (226,304 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rangle\tagsfm.dll kdcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 1.45 MB (1,520,640 bytes) 3/25/2003 7:00 AM Microsoft Corporation	dmserver 5.2.3790.0 (srv03_rtm.030324-2048) 24.00 KB (24,576 bytes) bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\dmserver.dll 89.00 KB (91,136 bytes) 89.00 KB (91,136 bytes) bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\srvsvc.dll 16.50 KB (16,896 bytes) bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\seclogon.dll 131.00 KB (134,144 bytes) 6/9/2003 12:42 PM Microsoft Corporation c:\windows\system32\wbem\wmisvc.dll 52.3790.0 (srv03_rtm.030324-2048) 528.00 KB (540,672 bytes) bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\vssapi.dll 52.3790.0 (srv03_rtm.030324-2048) 15.00 KB (15,360 bytes) bytes) 3/25/2003 7:00 AM Microsoft Corporation
iphlpapi 5.2.3790.0 (srv03_rtm.030324-2048) 82.50 KB (84,480 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\iphlpapi.dll schannel 5.2.3790.0 (srv03_rtm.030324-2048) 149.50 KB (153,088 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\schannel.dll wdigest 5.2.3790.0 (srv03_rtm.030324-2048) 61.00 KB (62,464 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\wdigest.dll rassfm 5.2.3790.0 (srv03_rtm.030324-2048) 20.50 KB (20,992 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rassfm.dll kdcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 221.00 KB (226,304 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rassfm.dll kdcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 221.00 KB (226,304 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\kdcsvc.dll ntdsa 5.2.3790.0 (srv03_rtm.030324-2048) 1.45 MB (1,520,640 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\kdcsvc.dll	dmserver 5.2.3790.0 (srv03_rtm.030324-2048) 24.00 KB (24,576 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\dmserver.dll srvsvc 5.2.3790.0 (srv03_rtm.030324-2048) 89.00 KB (91,136 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\srvsvc.dll seclogon 5.2.3790.0 (srv03_rtm.030324-2048) 16.50 KB (16,896 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\seclogon.dll wmisvc 5.2.3790.0 (srv03_rtm.030324-2048) 131.00 KB (134,144 bytes) 6/9/2003 12:42 PM Microsoft Corporation c:\windows\system32\swbem\wmisvc.dll vssapi 5.2.3790.0 (srv03_rtm.030324-2048) 528.00 KB (540,672 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\vssapi.dll winrnr 5.2.3790.0 (srv03_rtm.030324-2048) 15.00 KB (15,360 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\vssapi.dll winrnr 5.2.3790.0 (srv03_rtm.030324-2048) 15.00 KB (15,360 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\vssapi.dll winrnr 5.2.3790.0 (srv03_rtm.030324-2048) 15.00 KB (15,360 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\vssapi.dll winrnr 5.2.3790.0 (srv03_rtm.030324-2048) 15.00 KB (15,360 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\vssapi.dll
iphlpapi 5.2.3790.0 (srv03_rtm.030324-2048) 82.50 KB (84,480 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\iphlpapi.dll schannel 5.2.3790.0 (srv03_rtm.030324-2048) 149.50 KB (153,088 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\schannel.dll wdigest 5.2.3790.0 (srv03_rtm.030324-2048) 61.00 KB (62,464 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\wdigest.dll rassfm 5.2.3790.0 (srv03_rtm.030324-2048) 20.50 KB (20,992 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rassfm.dll kdcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 221.00 KB (226,304 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rassfm.dll kdcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 221.00 KB (226,304 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\kdcsvc.dll ntdsa 5.2.3790.0 (srv03_rtm.030324-2048) 1.45 MB (1,520,640 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\kdcsvc.dll ntdsa 5.2.3790.0 (srv03_rtm.030324-2048) 32.00 KB (32,768 ltd)	dmserver 5.2.3790.0 (srv03_rtm.030324-2048) 24.00 KB (24,576 bytes) bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\dmserver.dll 89.00 KB (91,136 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\srvsvc.dll seclogon 5.2.3790.0 (srv03_rtm.030324-2048) 16.50 KB (16,896 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\seclogon.dll wmisvc 5.2.3790.0 (srv03_rtm.030324-2048) 131.00 KB (134,144 bytes) 6/9/2003 12:42 PM Microsoft Corporation c:\windows\system32\wbem\wmisvc.dll 528.00 KB (540,672 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\wsapi.dll 528.00 KB (15,360 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\wsapi.dll 5.00 KB (15,360 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\window
iphlpapi 5.2.3790.0 (srv03_rtm.030324-2048) 82.50 KB (84,480 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\iphlpapi.dll schannel 5.2.3790.0 (srv03_rtm.030324-2048) 149.50 KB (153,088 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\schannel.dll wdigest 5.2.3790.0 (srv03_rtm.030324-2048) 61.00 KB (62,464 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\wdigest.dll rassfm 5.2.3790.0 (srv03_rtm.030324-2048) 20.50 KB (20,992 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rassfm.dll kdcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 221.00 KB (226,304 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rassfm.dll kdcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 221.00 KB (226,304 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\kdcsvc.dll ntdsa 5.2.3790.0 (srv03_rtm.030324-2048) 1.45 MB (1,520,640 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\kdcsvc.dll	dmserver 5.2.3790.0 (srv03_rtm.030324-2048) 24.00 KB (24,576 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\dmserver.dll srvsvc 5.2.3790.0 (srv03_rtm.030324-2048) 89.00 KB (91,136 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\srvsvc.dll seclogon 5.2.3790.0 (srv03_rtm.030324-2048) 16.50 KB (16,896 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\seclogon.dll wmisvc 5.2.3790.0 (srv03_rtm.030324-2048) 131.00 KB (134,144 bytes) 6/9/2003 12:42 PM Microsoft Corporation c:\windows\system32\swbem\wmisvc.dll vssapi 5.2.3790.0 (srv03_rtm.030324-2048) 528.00 KB (540,672 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\vssapi.dll winrnr 5.2.3790.0 (srv03_rtm.030324-2048) 15.00 KB (15,360 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\vssapi.dll winrnr 5.2.3790.0 (srv03_rtm.030324-2048) 15.00 KB (15,360 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\vssapi.dll winrnr 5.2.3790.0 (srv03_rtm.030324-2048) 15.00 KB (15,360 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\vssapi.dll winrnr 5.2.3790.0 (srv03_rtm.030324-2048) 15.00 KB (15,360 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\vssapi.dll
iphlpapi 5.2.3790.0 (srv03_rtm.030324-2048) 82.50 KB (84,480 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\iphlpapi.dll schannel 5.2.3790.0 (srv03_rtm.030324-2048) 149.50 KB (153,088 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\schannel.dll wdigest 5.2.3790.0 (srv03_rtm.030324-2048) 61.00 KB (62,464 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\schannel.dll rassfm 5.2.3790.0 (srv03_rtm.030324-2048) 20.50 KB (20,992 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rassfm.dll kdcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 221.00 KB (226,304 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rassfm.dll kdcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 221.00 KB (226,304 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rdcsvc.dll ntdsa 5.2.3790.0 (srv03_rtm.030324-2048) 1.45 MB (1,520,640 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rdcsvc.dll ntdsa 5.2.3790.0 (srv03_rtm.030324-2048) 32.00 KB (32,768 bytes) 3/25/2003 7:00 AM Microsoft Corporation	dmserver 5.2.3790.0 (srv03_rtm.030324-2048) 24.00 KB (24,576 bytes) bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\dmserver.dll 89.00 KB (91,136 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\syrvsvc.dll seclogon 5.2.3790.0 (srv03_rtm.030324-2048) 16.50 KB (16,896 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\seclogon.dll wmisvc 5.2.3790.0 (srv03_rtm.030324-2048) 131.00 KB (134,144 bytes) 6/9/2003 12:42 PM Microsoft Corporation c:\windows\system32\wbem\wmisvc.dll vssapi 5.2.3790.0 (srv03_rtm.030324-2048) 528.00 KB (540,672 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\system30324-2048) 15.00 KB (15,360 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\sys
iphlpapi 5.2.3790.0 (srv03_rtm.030324-2048) 82.50 KB (84,480 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\iphlpapi.dll schannel 5.2.3790.0 (srv03_rtm.030324-2048) 149.50 KB (153,088 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\schannel.dll wdigest 5.2.3790.0 (srv03_rtm.030324-2048) 61.00 KB (62,464 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\schannel.dll rassfm 5.2.3790.0 (srv03_rtm.030324-2048) 20.50 KB (20,992 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rassfm.dll kdcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 221.00 KB (226,304 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rassfm.dll kdcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 221.00 KB (226,304 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\radsfacesvc.dll ntdsa 5.2.3790.0 (srv03_rtm.030324-2048) 1.45 MB (1,520,640 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\radsfacesvc.dll ntdsat 5.2.3790.0 (srv03_rtm.030324-2048) 32.00 KB (32,768 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\radsfacesvc.dll ntdsat 5.2.3790.0 (srv03_rtm.030324-2048) 32.00 KB (32,768 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\radsfacesvc.dll ntdsatq 5.2.3790.0 (srv03_rtm.030324-2048) 32.00 KB (32,768 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\radsfacesvc.dll ntdsatq 5.2.3790.0 (srv03_rtm.030324-2048) 32.00 KB (32,768 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\radsfacesvc.dll	dmserver 5.2.3790.0 (srv03_rtm.030324-2048) 24.00 KB (24,576 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\dmserver.dll srvsvc 5.2.3790.0 (srv03_rtm.030324-2048) 89.00 KB (91,136 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\srvsvc.dll seclogon 5.2.3790.0 (srv03_rtm.030324-2048) 16.50 KB (16,896 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\seclogon.dll wmisvc 5.2.3790.0 (srv03_rtm.030324-2048) 131.00 KB (134,144 bytes) 6/9/2003 12:42 PM Microsoft Corporation c:\windows\system32\seclogon.dll vssapi 5.2.3790.0 (srv03_rtm.030324-2048) 528.00 KB (540,672 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\svsapi.dll winrnr 5.2.3790.0 (srv03_rtm.030324-2048) 15.00 KB (15,360 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\system
iphlpapi 5.2.3790.0 (srv03_rtm.030324-2048) 82.50 KB (84,480 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\iphlpapi.dll schannel 5.2.3790.0 (srv03_rtm.030324-2048) 149.50 KB (153,088 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\schannel.dll wdigest 5.2.3790.0 (srv03_rtm.030324-2048) 61.00 KB (62,464 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\wdigest.dll rassfm 5.2.3790.0 (srv03_rtm.030324-2048) 20.50 KB (20,992 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\wdigest.dll rassfm 5.2.3790.0 (srv03_rtm.030324-2048) 221.00 KB (226,304 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rassfm.dll kdcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 221.00 KB (226,304 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\kdcsvc.dll ntdsa 5.2.3790.0 (srv03_rtm.030324-2048) 1.45 MB (1,520,640 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntdsa.dll ntdsatq 5.2.3790.0 (srv03_rtm.030324-2048) 32.00 KB (32,768 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntdsatq.dll mswsock 5.2.3790.0 (srv03_rtm.030324-2048) 32.00 KB (32,768 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntdsatq.dll mswsock 5.2.3790.0 (srv03_rtm.030324-2048) 254.00 KB (260,096	dmserver 5.2.3790.0 (srv03_rtm.030324-2048) 24.00 KB (24,576 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\dmserver.dll srvsvc 5.2.3790.0 (srv03_rtm.030324-2048) 89.00 KB (91,136 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\srvsvc.dll seclogon 5.2.3790.0 (srv03_rtm.030324-2048) 16.50 KB (16,896 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\seclogon.dll wmisvc 5.2.3790.0 (srv03_rtm.030324-2048) 131.00 KB (134,144 bytes) 6/9/2003 12:42 PM Microsoft Corporation c:\windows\system32\seclogon.dll vssapi 5.2.3790.0 (srv03_rtm.030324-2048) 528.00 KB (540,672 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\system332\system3
iphlpapi 5.2.3790.0 (srv03_rtm.030324-2048) 82.50 KB (84,480 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\iphlpapi.dll schannel 5.2.3790.0 (srv03_rtm.030324-2048) 149.50 KB (153,088 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\schannel.dll wdigest 5.2.3790.0 (srv03_rtm.030324-2048) 61.00 KB (62,464 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\wdigest.dll rassfm 5.2.3790.0 (srv03_rtm.030324-2048) 20.50 KB (20,992 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rassfm.dll kdcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 221.00 KB (226,304 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rassfm.dll kdcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 221.00 KB (226,304 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rassfm.dll ntdsa 5.2.3790.0 (srv03_rtm.030324-2048) 1.45 MB (1,520,640 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rassfm.dll ntdsatq 5.2.3790.0 (srv03_rtm.030324-2048) 32.00 KB (32,768 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ransland strundsatq.dll mswsock 5.2.3790.0 (srv03_rtm.030324-2048) 254.00 KB (260,096 bytes) 3/25/2003 7:00 AM Microsoft Corporation	dmserver 5.2.3790.0 (srv03_rtm.030324-2048) 24.00 KB (24,576 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\dmserver.dll srvsvc 5.2.3790.0 (srv03_rtm.030324-2048) 89.00 KB (91,136 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\sysvc.dll seclogon 5.2.3790.0 (srv03_rtm.030324-2048) 16.50 KB (16,896 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\seclogon.dll wmisvc 5.2.3790.0 (srv03_rtm.030324-2048) 131.00 KB (134,144 bytes) 6/9/2003 12:42 PM Microsoft Corporation c:\windows\system32\system32\wbem\wmisvc.dll vssapi 5.2.3790.0 (srv03_rtm.030324-2048) 528.00 KB (540,672 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\system
iphlpapi 5.2.3790.0 (srv03_rtm.030324-2048) 82.50 KB (84,480 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\iphlpapi.dll schannel 5.2.3790.0 (srv03_rtm.030324-2048) 149.50 KB (153,088 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\schannel.dll wdigest 5.2.3790.0 (srv03_rtm.030324-2048) 61.00 KB (62,464 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\wdigest.dll rassfin 5.2.3790.0 (srv03_rtm.030324-2048) 20.50 KB (20,992 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rassfin.dll kdcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 221.00 KB (226,304 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rassfin.dll kdcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 221.00 KB (226,304 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\kdcsvc.dll ntdsa 5.2.3790.0 (srv03_rtm.030324-2048) 1.45 MB (1,520,640 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntdsat.dll ntdsatq 5.2.3790.0 (srv03_rtm.030324-2048) 32.00 KB (32,768 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntdsatq.dll mswsock 5.2.3790.0 (srv03_rtm.030324-2048) 254.00 KB (260,096 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntdsatq.dll mswsock 5.2.3790.0 (srv03_rtm.030324-2048) 254.00 KB (260,096 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntdsatq.dll mswsock 5.2.3790.0 (srv03_rtm.030324-2048) 254.00 KB (260,096 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntdsatq.dll	dmserver 5.2.3790.0 (srv03_rtm.030324-2048) 24.00 KB (24,576 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\dmserver.dll srvsvc 5.2.3790.0 (srv03_rtm.030324-2048) 89.00 KB (91,136 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\srvsvc.dll seclogon 5.2.3790.0 (srv03_rtm.030324-2048) 16.50 KB (16,896 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\seclogon.dll wmisvc 5.2.3790.0 (srv03_rtm.030324-2048) 131.00 KB (134,144 bytes) 6/9/2003 12:42 PM Microsoft Corporation c:\windows\system32\seclogon.dll vssapi 5.2.3790.0 (srv03_rtm.030324-2048) 528.00 KB (540,672 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\system332\system3
iphlpapi 5.2.3790.0 (srv03_rtm.030324-2048) 82.50 KB (84,480 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\iphlpapi.dll schannel 5.2.3790.0 (srv03_rtm.030324-2048) 149.50 KB (153,088 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\schannel.dll wdigest 5.2.3790.0 (srv03_rtm.030324-2048) 61.00 KB (62,464 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\wdigest.dll rassfm 5.2.3790.0 (srv03_rtm.030324-2048) 20.50 KB (20,992 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rassfm.dll kdcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 221.00 KB (226,304 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rassfm.dll kdcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 221.00 KB (226,304 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rassfm.dll ntdsa 5.2.3790.0 (srv03_rtm.030324-2048) 1.45 MB (1,520,640 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rassfm.dll ntdsatq 5.2.3790.0 (srv03_rtm.030324-2048) 32.00 KB (32,768 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ransland strundsatq.dll mswsock 5.2.3790.0 (srv03_rtm.030324-2048) 254.00 KB (260,096 bytes) 3/25/2003 7:00 AM Microsoft Corporation	dmserver 5.2.3790.0 (srv03_rtm.030324-2048) 24.00 KB (24,576 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\dmserver.dll srvsvc 5.2.3790.0 (srv03_rtm.030324-2048) 89.00 KB (91,136 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\sysvc.dll seclogon 5.2.3790.0 (srv03_rtm.030324-2048) 16.50 KB (16,896 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\seclogon.dll wmisvc 5.2.3790.0 (srv03_rtm.030324-2048) 131.00 KB (134,144 bytes) 6/9/2003 12:42 PM Microsoft Corporation c:\windows\system32\system32\wbem\wmisvc.dll vssapi 5.2.3790.0 (srv03_rtm.030324-2048) 528.00 KB (540,672 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\system
iphlpapi 5.2.3790.0 (srv03_rtm.030324-2048) 82.50 KB (84,480 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\iphlpapi.dll schannel 5.2.3790.0 (srv03_rtm.030324-2048) 149.50 KB (153,088 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\schannel.dll wdigest 5.2.3790.0 (srv03_rtm.030324-2048) 61.00 KB (62,464 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\wdigest.dll rassfin 5.2.3790.0 (srv03_rtm.030324-2048) 20.50 KB (20,992 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rassfin.dll kdcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 221.00 KB (226,304 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rassfin.dll kdcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 221.00 KB (226,304 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\kdcsvc.dll ntdsa 5.2.3790.0 (srv03_rtm.030324-2048) 1.45 MB (1,520,640 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntdsat.dll ntdsatq 5.2.3790.0 (srv03_rtm.030324-2048) 32.00 KB (32,768 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntdsatq.dll mswsock 5.2.3790.0 (srv03_rtm.030324-2048) 254.00 KB (260,096 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntdsatq.dll mswsock 5.2.3790.0 (srv03_rtm.030324-2048) 254.00 KB (260,096 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntdsatq.dll mswsock 5.2.3790.0 (srv03_rtm.030324-2048) 254.00 KB (260,096 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\mswsock.dll esent 5.2.3790.0 (srv03_rtm.030324-2048) 1.01 MB (1,056,256	dmserver 5.2.3790.0 (srv03_rtm.030324-2048) 24.00 KB (24,576 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\dmserver.dll srvsvc 5.2.3790.0 (srv03_rtm.030324-2048) 89.00 KB (91,136 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\svrsvc.dll seclogon 5.2.3790.0 (srv03_rtm.030324-2048) 16.50 KB (16,896 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\seclogon.dll wmisvc 5.2.3790.0 (srv03_rtm.030324-2048) 131.00 KB (134,144 bytes) 6/9/2003 12:42 PM Microsoft Corporation c:\windows\system32\swbem\wmisvc.dll vssapi 5.2.3790.0 (srv03_rtm.030324-2048) 528.00 KB (540,672 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\svsapi.dll winrnr 5.2.3790.0 (srv03_rtm.030324-2048) 15.00 KB (15,360 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\svsapi.dll winrnr 5.2.3790.0 (srv03_rtm.030324-2048) 209.00 KB (214,016 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\system332\system
iphlpapi 5.2.3790.0 (srv03_rtm.030324-2048) 82.50 KB (84,480 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\iphlpapi.dll schannel 5.2.3790.0 (srv03_rtm.030324-2048) 149.50 KB (153,088 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\schannel.dll wdigest 5.2.3790.0 (srv03_rtm.030324-2048) 61.00 KB (62,464 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\wdigest.dll rassfin 5.2.3790.0 (srv03_rtm.030324-2048) 20.50 KB (20,992 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rassfin.dll kdcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 221.00 KB (226,304 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\rassfin.dll kdcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 221.00 KB (226,304 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\kdcsvc.dll ntdsa 5.2.3790.0 (srv03_rtm.030324-2048) 1.45 MB (1,520,640 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntdsat.dll ntdsatq 5.2.3790.0 (srv03_rtm.030324-2048) 32.00 KB (32,768 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntdsatq.dll mswsock 5.2.3790.0 (srv03_rtm.030324-2048) 254.00 KB (260,096 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntdsatq.dll mswsock 5.2.3790.0 (srv03_rtm.030324-2048) 254.00 KB (260,096 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntdsatq.dll mswsock 5.2.3790.0 (srv03_rtm.030324-2048) 254.00 KB (260,096 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntdsatq.dll	dmserver 5.2.3790.0 (srv03_rtm.030324-2048) 24.00 KB (24,576 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\dmserver.dll srvsvc 5.2.3790.0 (srv03_rtm.030324-2048) 89.00 KB (91,136 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\svsvc.dll seclogon 5.2.3790.0 (srv03_rtm.030324-2048) 16.50 KB (16,896 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\seclogon.dll wmisvc 5.2.3790.0 (srv03_rtm.030324-2048) 131.00 KB (134,144 bytes) 6/9/2003 12:42 PM Microsoft Corporation c:\windows\system32\seclogon.dll vssapi 5.2.3790.0 (srv03_rtm.030324-2048) 528.00 KB (540,672 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\svssapi.dll winrnr 5.2.3790.0 (srv03_rtm.030324-2048) 15.00 KB (15,360 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\system

tapi32 5.2.3790.0 (srv03_rtm.030324-2048) 175.00 KB (179,200 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\tapi32.dll wzcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 272.50 KB (279,040 bytes) 3/25/2003 7:15 AM Microsoft Corporation c:\windows\system32\wzcsvc.dll wmi 5.2.3790.0 (srv03_rtm.030324-2048) 6.50 KB (6,656 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\wmi.dll dhcpcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 101.50 KB (103,936 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\wmi.dll dhcpcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 101.50 KB (103,936 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\dhcpcsvc.dll wzcsapi 5.2.3790.0 (srv03_rtm.030324-2048) 24.50 KB (25,088 bytes) 3/25/2003 7:15 AM Microsoft Corporation c:\windows\system32\wzcsapi.dll es 2001.12.4720.0 (srv03_rtm.030324-2048) 221.50 KB (226,816 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\es.dll comsvcs 2001.12.4720.0 (srv03_rtm.030324-2048) 1.14 MB (1,199,616 bytes) 6/9/2003 12:42 PM Microsoft Corporation	pchsvc 5.2.3790.0 (srv03_rtm.030324-2048) 31.50 KB (32,256 bytes) 6/9/2003 12:46 PM Microsoft Corporation c:\windows\pchealth\helpctr\binaries\pchsvc.dll ersvc 5.2.3790.0 (srv03_rtm.030324-2048) 22.00 KB (22,528 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ersvc.dll ibmhpasv 5.1.1.1 14.50 KB (14,848 bytes) 6/9/2003 1:38 PM IBM Corporation c:\windows\system32\ersvc.dll 1,008.50 KB (1,032,704 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\explorer.exe browseui 6.00.3790.0 (srv03_rtm.030324-2048) 1.01 MB (1,057,280 bytes) bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\browseui.dll shdocvw 6.00.3790.0 (srv03_rtm.030324-2048) 1.33 MB (1,393,664 bytes) bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\shdocvw.dll apphelp 5.2.3790.0 (srv03_rtm.030324-2048) 122.00 KB (124,928 bytes) bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\shdocvw.dll
c:\windows\system32\comsvcs.dll sens 5.2.3790.0 (srv03_rtm.030324-2048) 35.50 KB (36,352	themeui 6.00.3790.0 (srv03_rtm.030324-2048) 360.50 KB (369,152 bytes) 3/25/2003 7:00 AM Microsoft Corporation
bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\sens.dll rasadhlp 5.2.3790.0 (srv03_rtm.030324-2048) 6.50 KB (6,656 bytes) 3/25/2003 7:00 AM Microsoft Corporation	c:\windows\system32\themeui.dll msimg32 5.2.3790.0 (srv03_rtm.030324-2048) 4.50 KB (4,608 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\msimg32.dll
c:\windows\system32\rasadhlp.dll wbemcore 5.2.3790.0 (srv03_rtm.030324-2048) 457.00 KB (467,968 bytes) 6/9/2003 12:42 PM Microsoft Corporation c:\windows\system32\wbem\wbemcore.dll	actxprxy 6.00.3790.0 (srv03_rtm.030324-2048) 95.00 KB (97,280 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\actxprxy.dll linkinfo 5.2.3790.0 (srv03_rtm.030324-2048) 16.50 KB (16,896
esscli 5.2.3790.0 (srv03_rtm.030324-2048) 235.50 KB (241,152 bytes) 6/9/2003 12:42 PM Microsoft Corporation c:\windows\system32\wbem\esscli.dll	bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\linkinfo.dll ntshrui 6.00.3790.0 (srv03 rtm.030324-2048) 136.00 KB (139,264
wmiutils 5.2.3790.0 (srv03 rtm.030324-2048) 90.50 KB (92,672 bytes) 6/9/2003 12:42 PM Microsoft Corporation c:\windows\system32\wbem\wmiutils.dll	bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntshrui.dll urlmon 6.00.3790.0 (srv03_rtm.030324-2048) 501.50 KB (513,536
repdrvfs 5.2.3790.0 (srv03_rtm.030324-2048) 165.00 KB (168,960 bytes) 6/9/2003 12:42 PM Microsoft Corporation c:\windows\system32\wbem\repdrvfs.dll	bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\urlmon.dll webcheck 6.00.3790.0 (srv03 rtm.030324-2048) 261.50 KB (267,776
wmiprvsd 5.2.3790.0 (srv03_rtm.030324-2048) 405.50 KB (415,232 bytes) 6/9/2003 12:42 PM Microsoft Corporation c:\windows\system32\wbem\wmiprvsd.dll	bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\webcheck.dll wsock32 5.2.3790.0 (srv03_rtm.030324-2048) 22.00 KB (22,528
wbemess 5.2.3790.0 (srv03_rtm.030324-2048) 256.50 KB (262,656 bytes) 6/9/2003 12:42 PM Microsoft Corporation c:\windows\system32\wbem\wbemess.dll	bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\wsock32.dll stobject 5.2.3790.0 (srv03_rtm.030324-2048) 117.50 KB (120,320
netshell 5.2.3790.0 (srv03_rtm.030324-2048) 1.67 MB (1,747,456 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\netshell.dll	bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\stobject.dll batmeter 6.00.3790.0 (srv03 rtm.030324-2048) 28.50 KB (29,184
clusapi 5.2.3790.0 (srv03_rtm.030324-2048) 56.00 KB (57,344 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\clusapi.dll hnetcfg 5.2.3790.0 (srv03_rtm.030324-2048) 243.50 KB (249,344	bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\batmeter.dll powrprof 6.00.3790.0 (srv03_rtm.030324-2048) 14.50 KB (14,848 bytes) 3/25/2003 7:00 AM Microsoft Corporation
bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\hnetcfg.dll wininet 6.00.3790.0 (srv03_rtm.030324-2048) 609.00 KB (623,616 bytes) 3/25/2003 7:00 AM Microsoft Corporation	c:\windows\system32\powrprof.dll printui 5.2.3790.0 (srv03_rtm.030324-2048) 536.50 KB (549,376 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\printui.dll
c:\windows\system32\wininet.dll rasdlg 5.2.3790.0 (srv03_rtm.030324-2048) 642.00 KB (657,408 bytes) 3/25/2003 7:00 AM Microsoft Corporation	cfgmgr32 5.2.3790.0 (srv03_rtm.030324-2048) 17.50 KB (17,920 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\cfgmgr32.dll
c:\windows\system32\rasdlg.dll ncprov 5.2.3790.0 (srv03_rtm.030324-2048) 43.00 KB (44,032 bytes) 6/9/2003 12:42 PM Microsoft Corporation	drprov 5.2.3790.0 (srv03_rtm.030324-2048) 12.50 KB (12,800 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\drprov.dll
c:\windows\system32\wbem\ncprov.dll wbemcons 5.2.3790.0 (srv03_rtm.030324-2048) 69.00 KB (70,656 bytes) 6/9/2003 12:42 PM Microsoft Corporation	ntlanman 5.2.3790.0 (srv03_rtm.030324-2048) 41.00 KB (41,984 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\ntlanman.dll
c:\windows\system32\wbem\wbemcons.dll	netui0 5.2.3790.0 (srv03_rtm.030324-2048) 75.50 KB (77,312 bytes) 3/25/2003 7:00 AM Microsoft Corporation c:\windows\system32\netui0.dll

netui 1 5.2.3790.0 (srv03_rtm.030324-2048) 184.00 KB (188,416	mtxclu 2001.12.4720.0 (srv03_rtm.030324-2048) 74.50 KB (76,288
bytes) 3/25/2003 7:00 AM Microsoft Corporation	bytes) 3/25/2003 7:00 AM Microsoft Corporation
c:\windows\system32\netui1.dll	c:\windows\system32\mtxclu.dll
davclnt 5.2.3790.0 (srv03 rtm.030324-2048) 23.50 KB (24,064	resutils 5.2.3790.0 (srv03 rtm.030324-2048) 59.00 KB (60,416
bytes) 3/25/2003 7:00 AM Microsoft Corporation	bytes) 3/25/2003 7:00 AM Microsoft Corporation
c:\windows\system32\davcInt.dll	c:\windows\system32\resutils.dll
browselc 6.00.3790.0 (srv03 rtm.030324-2048) 62.00 KB (63,488	mfc42u 6.05.3014.0 960.00 KB (983,040 bytes) 3/25/2003
` = /	
bytes) 3/25/2003 7:00 AM Microsoft Corporation	
c:\windows\system32\browselc.dll	ssnetlib 2000.080.0766.00 80.56 KB (82,492 bytes) 6/9/2003
shdoclc 6.00.3790.0 (srv03_rtm.030324-2048) 588.50 KB (602,624	1:08 PM Microsoft Corporation c:\program files\microsoft sql
bytes) 3/25/2003 7:00 AM Microsoft Corporation	server\mssql\binn\ssnetlib.dll
c:\windows\system32\shdoclc.dll	ssnmpn70 2000.080.0534.00 24.56 KB (25,148 bytes) 6/9/2003
mprui 5.2.3790.0 (srv03 rtm.030324-2048) 49.00 KB (50,176	1:08 PM Microsoft Corporation c:\program files\microsoft sql
bytes) 3/25/2003 7:00 AM Microsoft Corporation	server\mssql\binn\ssnmpn70.dll
c:\windows\system32\mprui.dll	security 5.2.3790.0 (srv03 rtm.030324-2048) 5.50 KB (5,632 bytes)
netui2 5.2.3790.0 (srv03 rtm.030324-2048) 309.50 KB (316,928	3/25/2003 7:00 AM Microsoft Corporation
bytes) 3/25/2003 7:00 AM Microsoft Corporation	c:\windows\system32\security.dll
• *	
c:\windows\system32\netui2.dll	ssmslpcn 2000.080.0760.00 28.56 KB (29,244 bytes) 6/9/2003
comdlg32 6.00.3790.0 (srv03_rtm.030324-2048) 261.00 KB (267,264	1:08 PM Microsoft Corporation c:\program files\microsoft sql
bytes) 3/25/2003 7:00 AM Microsoft Corporation	server\mssql\binn\ssmslpcn.dll
c:\windows\system32\comdlg32.dll	isqlw 2000.080.0760.00 344.56 KB (352,828 bytes) 6/9/2003
netmsg 5.2.3790.0 (srv03_rtm.030324-2048) 178.00 KB (182,272	1:08 PM Microsoft Corporation c:\program files\microsoft sql
bytes) 3/25/2003 7:00 AM Microsoft Corporation	server\80\tools\binn\isqlw.exe
c:\windows\system32\netmsg.dll	sqlunirl 2000.080.0728.00 176.56 KB (180,800 bytes) 3/25/2003
netplwiz 5.2.3790.0 (srv03 rtm.030324-2048) 843.00 KB (863,232	7:00 AM Microsoft Corporation c:\windows\system32\sqlunirl.dll
bytes) 3/25/2003 7:00 AM Microsoft Corporation	sqlresld 2000.080.0382.00 28.56 KB (29,248 bytes) 6/9/2003
c:\windows\system32\netplwiz.dll	1
J 1	1 1 5
zipfldr 6.00.3790.0 (srv03_rtm.030324-2048) 316.00 KB (323,584	server\80\tools\binn\sqlresld.dll
bytes) 3/25/2003 7:00 AM Microsoft Corporation	sqlgui 2000.080.0760.00 444.56 KB (455,232 bytes) 6/9/2003
c:\windows\system32\zipfldr.dll	1:09 PM Microsoft Corporation c:\program files\microsoft sql
pronomgr 6.2.35.0 84.00 KB (86,016 bytes) 3/11/2003 3:24 PM	server\80\tools\binn\sqlgui.dll
Intel(R) Corporation c:\program files\intel\ncs\proset\pronomgr.exe	w95scm 2000.080.0760.00 48.56 KB (49,728 bytes) 6/9/2003
enupguir 6.2.35.0 340.00 KB (348,160 bytes) 3/11/2003 3:15 PM	1:09 PM Microsoft Corporation c:\program files\microsoft sql
Intel(R) Corporation c:\program files\intel\ncs\proset\enupguir.dll	server\80\tools\binn\w95scm.dll
pnc802 3 6.2.35.0 84.00 KB (86,016 bytes) 3/11/2003 3:33 PM	odbc32 3.525.1022.0 (srv03 rtm.030324-2048) 232.00 KB (237,568
Intel(R) Corporation c:\program files\intel\ncs\proset\8023\pnc802 3.dll	bytes) 3/25/2003 7:00 AM Microsoft Corporation
enupcmrs 6.2.35.0 344.07 KB (352,325 bytes) 3/11/2003 3:20 PM	c:\windows\system32\odbc32.dll
1	
Intel(R) Corporation c:\program files\intel\ncs\proset\8023\enupcmrs.dll	sqlsvc 2000.080.0760.00 92.56 KB (94,784 bytes) 6/9/2003
cmd 5.2.3790.0 (srv03_rtm.030324-2048) 374.00 KB (382,976	1:09 PM Microsoft Corporation c:\program files\microsoft sql
bytes) 3/25/2003 7:00 AM Microsoft Corporation	server\80\tools\binn\sqlsvc.dll
c:\windows\system32\cmd.exe	odbcbcp 2000.085.1022.00 (srv03_rtm.030324-2048) 24.00 KB (24,576
wpabaln 5.2.3790.0 (srv03_rtm.030324-2048) 31.00 KB (31,744	bytes) 3/25/2003 7:00 AM Microsoft Corporation
bytes) 3/25/2003 7:00 AM Microsoft Corporation	c:\windows\system32\odbcbcp.dll
c:\windows\system32\wpabaln.exe	imm32 5.2.3790.0 (srv03 rtm.030324-2048) 105.50 KB (108,032
sqlservr 2000.080.0760.00 7.17 MB (7,520,337 bytes) 6/9/2003	bytes) 3/25/2003 7:00 AM Microsoft Corporation
1:08 PM Microsoft Corporation c:\program files\microsoft sql	c:\windows\system32\imm32.dll
server\mssql\binn\sqlservr.exe	sqlqry 2000.080.0760.00 392.56 KB (401,984 bytes) 6/9/2003
opends60 2000.080.0194.00 24.06 KB (24,639 bytes) 6/9/2003	1:08 PM Microsoft Corporation c:\program files\microsoft sql
1	server\80\tools\binn\sqlgry.dll
1:08 PM Microsoft Corporation c:\program files\microsoft sql	1 1 2
server\mssql\binn\opends60.dll	pfutil80 2000.080.0760.00 272.56 KB (279,104 bytes) 6/9/2003
ums 2000.080.0760.00 52.55 KB (53,808 bytes) 6/9/2003	1:09 PM Microsoft Corporation c:\program files\microsoft sql
1:08 PM Microsoft Corporation c:\program files\microsoft sql	server\80\tools\binn\pfutil80.dll
server\mssql\binn\ums.dll	semsfc 2000.080.0760.00 228.56 KB (234,048 bytes) 6/9/2003
sqlsort 2000.080.0760.00 576.56 KB (590,396 bytes) 6/9/2003	1:09 PM Microsoft Corporation c:\program files\microsoft sql
1:08 PM Microsoft Corporation c:\program files\microsoft sql	server\80\tools\binn\semsfc.dll
server\mssql\binn\sqlsort.dll	pfclnt80 2000.080.0760.00 404.56 KB (414,272 bytes) 6/9/2003
msvcirt 7.0.3790.0 (srv03 rtm.030324-2048) 50.00 KB (51,200	1:09 PM Microsoft Corporation c:\program files\microsoft sql
bytes) 3/25/2003 7:00 AM Microsoft Corporation	server\80\tools\binn\pfcInt80.dll
c:\windows\system32\msvcirt.dll sqlevn70 2000.080.0760.00 28.00 KB (28.672 bytes) 6/9/2003	odbcint 3.525.1022.0 (srv03_rtm.030324-2048) 92.00 KB (94,208 bytes) 2/25/2003 7:00 AM Microsoft Corporation
1	bytes) 3/25/2003 7:00 AM Microsoft Corporation
1:08 PM Microsoft Corporation c:\program files\microsoft sql	c:\windows\system32\odbcint.dll
server\mssql\binn\resources\1033\sqlevn70.rll	sqlsvc 2000.080.0194.00 24.00 KB (24,576 bytes) 6/9/2003
xolehlp 2001.12.4720.0 (srv03_rtm.030324-2048) 8.50 KB (8,704 bytes)	1:09 PM Microsoft Corporation c:\program files\microsoft sql
6/9/2003 12:42 PM Microsoft Corporation	server\80\tools\binn\resources\1033\sqlsvc.rll
c:\windows\system32\xolehlp.dll	
	sqlgui 2000.080.0194.00 56.00 KB (57,344 bytes) 6/9/2003
msdtcprx 2001.12.4720.0 (srv03 rtm.030324-2048) 427.50 KB (437,760	•
	sqlgui 2000.080.0194.00 56.00 KB (57,344 bytes) 6/9/2003 1:09 PM Microsoft Corporation c:\program files\microsoft sql
msdtcprx 2001.12.4720.0 (srv03_rtm.030324-2048) 427.50 KB (437,760 bytes) 6/9/2003 12:42 PM Microsoft Corporation c:\windows\system32\msdtcprx.dll	sqlgui 2000.080.0194.00 56.00 KB (57,344 bytes) 6/9/2003

C 2000 000 0104 00 24 00 V/D (24 57(1 +)) (10/2002	1 1 1 1 (00 2700 0 (02) 020224 2040) 442 50 Y/D (454 144
semsfc 2000.080.0194.00 24.00 KB (24,576 bytes) 6/9/2003	mshtmled 6.00.3790.0 (srv03_rtm.030324-2048) 443.50 KB (454,144
1:09 PM Microsoft Corporation c:\program files\microsoft sql	bytes) 3/25/2003 7:00 AM Microsoft Corporation
server\80\tools\binn\resources\1033\semsfc.rll	c:\windows\system32\mshtmled.dll
pfclnt80 2000.080.0194.00 28.00 KB (28,672 bytes) 6/9/2003	vbscript 5.6.0.8515 404.00 KB (413,696 bytes) 3/25/2003 7:00 AM
1:09 PM Microsoft Corporation c:\program files\microsoft sql	Microsoft Corporation c:\windows\system32\vbscript.dll
server\80\tools\binn\resources\1033\pfclnt80.rll	mfc42 6.05.3014.0 960.00 KB (983,040 bytes) 3/25/2003
pfutil80 2000.080.0382.00 144.00 KB (147,456 bytes) 6/9/2003	7:00 AM Microsoft Corporation c:\windows\system32\mfc42.dll
1:09 PM Microsoft Corporation c:\program files\microsoft sql	msinfo 5.2.3790.0 (srv03_rtm.030324-2048) 358.50 KB (367,104
server\80\tools\binn\resources\1033\pfutil80.rll	bytes) 6/9/2003 12:46 PM Microsoft Corporation
sqlqry 2000.080.0194.00 180.00 KB (184,320 bytes) 6/9/2003	c:\windows\pchealth\helpctr\binaries\msinfo.dll
1:08 PM Microsoft Corporation c:\program files\microsoft sql	riched32 5.2.3790.0 (srv03_rtm.030324-2048) 3.50 KB (3,584 bytes)
server\80\tools\binn\resources\1033\sqlqry.rll	3/25/2003 7:00 AM Microsoft Corporation
isqlw 2000.080.0382.00 240.00 KB (245,760 bytes) 6/9/2003	c:\windows\system32\riched32.dll
1:08 PM Microsoft Corporation c:\program files\microsoft sql	riched20 5.31.23.1218 406.00 KB (415,744 bytes) 3/25/2003
server\80\tools\binn\resources\1033\isqlw.rll	7:00 AM Microsoft Corporation c:\windows\system32\riched20.dll
odbccp32 3.525.1022.0 (srv03_rtm.030324-2048) 100.00 KB (102,400	helpsvc 5.2.3790.0 (srv03_rtm.030324-2048) 720.00 KB (737,280
bytes) 3/25/2003 7:00 AM Microsoft Corporation	bytes) 6/9/2003 12:46 PM Microsoft Corporation
c:\windows\system32\odbccp32.dll	c:\windows\pchealth\helpctr\binaries\helpsvc.exe
objmgr 2000.080.0760.00 308.56 KB (315,968 bytes) 6/9/2003	
1:08 PM Microsoft Corporation c:\program files\microsoft sql	[Services]
server\80\tools\binn\objmgr.dll	
objmgr 2000.080.0760.00 56.00 KB (57,344 bytes) 6/9/2003	Display Name Name State Start ModeService Type
1:08 PM Microsoft Corporation c:\program files\microsoft sql	Path Error Control Start NameTag ID
server\80\tools\binn\resources\1033\objmgr.rll	Alerter Alerter Stopped Disabled Share Process
sqllex 2000.080.0194.00 148.06 KB (151,616 bytes) 6/9/2003	c:\windows\system32\svchost.exe -k localservice Normal NT
1:09 PM Microsoft Corporation c:\program files\microsoft sql	AUTHORITY\LocalService 0
server\80\tools\binn\sqllex.dll	Application Layer Gateway Service ALG Stopped Manual
sqlsrv32 2000.085.1022.00 (srv03_rtm.030324-2048) 404.00 KB (413,696	Own Process c:\windows\system32\alg.exe Normal NT
bytes) 3/25/2003 7:00 AM Microsoft Corporation	AUTHORITY\LocalService 0
c:\windows\system32\sqlsrv32.dll	Application Management AppMgmt Stopped Manual Share
sqlsrv32 2000.085.1022.00 (srv03 rtm.030324-2048) 88.00 KB (90,112	Process c:\windows\system32\svchost.exe -k netsvcs Normal
bytes) 3/25/2003 7:00 AM Microsoft Corporation	LocalSystem 0
c:\windows\system32\sqlsrv32.rll	Windows Audio AudioSrv Running Auto Share Process
dbnetlib 2000.085.1022 (srv03 rtm.030324-2048) 76.00 KB (77,824	c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0
bytes) 3/25/2003 7:00 AM Microsoft Corporation	Background Intelligent Transfer Service BITS Stopped Manual
c:\windows\system32\dbnetlib.dll	Share Process c:\windows\system32\svchost.exe -k netsvcs Normal
dbmslpcn 2000.080.0760.00 28.56 KB (29,244 bytes) 6/9/2003	LocalSystem 0
1:08 PM Microsoft Corporation c:\windows\system32\dbmslpcn.dll	Computer Browser Browser Stopped Manual Share Process
helpctr 5.2.3790.0 (srv03 rtm.030324-2048) 764.00 KB (782,336	c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0
bytes) 6/9/2003 12:46 PM Microsoft Corporation	Indexing Service CiSvc Stopped Disabled Share Process
c:\windows\pchealth\helpctr\binaries\helpctr.exe	c:\windows\system32\cisvc.exe Normal LocalSystem 0
heappres 5.2.3790.0 (srv03 rtm.030324-2048) 6.50 KB (6,656 bytes)	ClipBook ClipSrv Stopped Disabled Own Process
6/9/2003 12:46 PM Microsoft Corporation	c:\windows\system32\clipsrv.exe Normal LocalSystem 0
c:\windows\pchealth\helpctr\binaries\hcappres.dll	COM+ System Application COMSysApp Stopped Manual
itss 5.2.3790.0 (srv03_rtm.030324-2048) 119.50 KB (122,368	Own Process c:\windows\system32\dllhost.exe
bytes) 3/25/2003 7:00 AM Microsoft Corporation	/processid: {02d4b3f1-fd88-11d1-960d-00805fc79235} Normal
c:\windows\system32\itss.dll	LocalSystem 0
msxml3 8.40.9419.0 1.28 MB (1,337,344 bytes) 3/25/2003	Cryptographic Services CryptSvc Stopped Manual Share
7:00 AM Microsoft Corporation c:\windows\system32\msxml3.dll	Process c:\windows\system32\svchost.exe -k netsvcs Normal
pchshell 5.2.3790.0 (srv03 rtm.030324-2048) 100.50 KB (102,912	LocalSystem 0
bytes) 6/9/2003 12:46 PM Microsoft Corporation	Distributed File System Dfs Stopped Manual Own
c:\windows\pchealth\helpctr\binaries\pchshell.dll	Process c:\windows\system32\dfssvc.exe Normal LocalSystem 0
mlang 6.00.3790.0 (srv03 rtm.030324-2048) 570.00 KB (583,680	DHCP Client Dhcp Stopped Manual Share Process
bytes) 3/25/2003 7:00 AM Microsoft Corporation	c:\windows\system32\svchost.exe -k networkservice Normal NT
c:\windows\system32\mlang.dll	AUTHORITY\NetworkService 0
mshtml 6.00.3790.0 (srv03_rtm.030324-2048) 2.78 MB (2,916,352	Logical Disk Manager Administrative Service dmadmin Stopped
bytes) 3/25/2003 7:00 AM Microsoft Corporation	Manual Share Process c:\windows\system32\dmadmin.exe /com
c:\windows\system32\mshtml.dll	Normal LocalSystem 0
msimtf 5.2.3790.0 (srv03 rtm.030324-2048) 149.00 KB (152,576	Logical Disk Managerdmserver Running Auto Share Process
bytes) 3/25/2003 7:00 AM Microsoft Corporation	c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0
c:\windows\system32\msimtf.dll	DNS Client Dnscache Stopped Manual Share Process
msctf 5.2.3790.0 (srv03_rtm.030324-2048) 287.00 KB (293,888	c:\windows\system32\svchost.exe -k networkservice Normal NT
bytes) 3/25/2003 7:00 AM Microsoft Corporation	AUTHORITY\NetworkService 0
c:\windows\system32\msctf.dll	Error Reporting Service ERSvc Running Auto Share
jscript 5.6.0.8515 436.00 KB (446,464 bytes) 3/25/2003 7:00 AM	Process c:\windows\system32\svchost.exe -k winerr Ignore
Microsoft Corporation c:\windows\system32\jscript.dll	LocalSystem 0
msls31 3.10.349.0 147.00 KB (150,528 bytes) 3/25/2003 7:00 AM	Event Log Eventlog Running Auto Share Process
Microsoft Corporation c:\windows\system32\msls31.dll	c:\windows\system32\services.exe Normal LocalSystem 0
	Troinidi Localbystein 0

COM+ Event System	EventSyst	em	Running	Manual	Share	
Process c:\window	vs\system32	\svchost.ex	e -k netsvc	Normal		
LocalSystem Mylex Global Array	() Manager Se	er or	comsem	Stopped	Manual	
Own Process		gamserv\ga	gamscm mscm.exe	Stopped Normal	Manuai	
LocalSystem	0	O				
	helpsvc			Share Pro		2
c:\windows\system32				LocalSyst		0
Human Interface Dev Process c:\window	vs\system32			Disabled Normal	Share	
LocalSystem	0	Svenost.ea	C -K HOLSVO	NUIIII		
HTTP SSLHTTPFilt		Stopped		Share Pro	cess	
c:\windows\system32					0	
IBM Active PCI Aler Process c:\window				Auto Normal	Own	
LocalSystem	vs\system32 0	\lomnpasv.	exe	Nomai		
IMAPI CD-Burning		ce	ImapiServ	ice	Stopped	
Disabled Own Proc	ess		s\system32			
LocalSystem	0	G: 1	D: 11 1	^ D		
Intersite Messaging c:\windows\system32		Stopped	Disabled Normal	Own Proc LocalSyst		0
Kerberos Key Distrib			kdc	Stopped		U
Share Process		vs\system32		Normal	D.Du.	
LocalSystem	0	-				
Server lanmanser		Running	Auto	Share Prod		^
c:\windows\system32 Workstation			Normal Running	LocalSyst Auto	em Share	0
	vs\system32				Share	
LocalSystem	0			,		
License Logging	LicenseSe		Stopped	Disabled	Own	
	vs\system32		Normal	NT		
AUTHORITY\Netwo		0 I mHosts	Running	Auto	Share	
		LIIII IUsis	Kummg	Auto	Silaic	
Process c:\windov	vs\svstem32	\svchost.ex	e -k localse	rvice	Normal	
Process c:\windov NT AUTHORITY\La		\svchost.ex 0	e -k localse	rvice	Normal	
NT AUTHORITY\Lo Messenger Messenge	ocalService r Stopped	0 Disabled	Share Prod	cess		
NT AUTHORITY\Ld Messenger Messenge c:\windows\system32	ocalService or Stopped 2\svchost.ex	0 Disabled e -k netsvcs	Share Proc Normal	cess LocalSyst	em	0
NT AUTHORITY\Ld Messenger Messenge c:\windows\system32 NetMeeting Remote	ocalService or Stopped Nsvchost.ex Desktop Sha	0 Disabled e -k netsves aring	Share Proc Normal mnmsrvc	cess LocalSyst Stopped	em Disabled	0
NT AUTHORITY\Ld Messenger Messenge c:\windows\system32	ocalService or Stopped Nsvchost.ex Desktop Sha	0 Disabled e -k netsvcs	Share Proc Normal mnmsrvc	cess LocalSyst Stopped	em	0
NT AUTHORITY\Lo Messenger Messenge c:\windows\system32 NetMeeting Remote Own Process	ocalService or Stopped 2\svchost.ex Desktop Sha c:\window 0 on Coordina	0 Disabled e -k netsves aring vs\system32 ator	Share Proc s Normal mnmsrvc mnmsrvc.e	LocalSyst Stopped exe	em Disabled	0
NT AUTHORITY\Ld Messenger Messenge c:\windows\system32 NetMeeting Remote Own Process LocalSystem Distributed Transacti Own Process	ocalService or Stopped 2\svchost.ex Desktop Sha c:\window 0 on Coordina c:\window	O Disabled e -k netsves aring vs\system32 ator vs\system32	Share Proc s Normal mnmsrvc mnmsrvc.e	LocalSyst Stopped exe	em Disabled Normal	0
NT AUTHORITY\Ld Messenger Messenge c:\windows\system32 NetMeeting Remote Own Process LocalSystem Distributed Transacti Own Process AUTHORITY\Netwo	ocalService or Stopped 2\svchost.ex Desktop Sha c:\window 0 on Coordina c:\window orkService	0 Disabled e -k netsves aring vs\system32 ator vs\system32 0	Share Proces Normal mnmsrvc. mnmsrvc.e MSDTC	LocalSyst Stopped exe Running Normal	em Disabled Normal Auto NT	0
NT AUTHORITY\Le Messenger Messenge c:\windows\system32 NetMeeting Remote Own Process LocalSystem Distributed Transacti Own Process AUTHORITY\Netwo	ocalService or Stopped 2\svchost.ex Desktop Sha c:\window 0 on Coordina c:\window orkService MSIServe	0 Disabled e -k netsves aring vs\system32 attor vs\system32 0 r Stopped	Share Proc S Normal mnmsrvc. mnmsrvc.e MSDTC \msdtc.exe	LocalSyst Stopped exe Running Normal Share Proc	em Disabled Normal Auto NT	
NT AUTHORITY\Ld Messenger Messenge c:\windows\system32 NetMeeting Remote Own Process LocalSystem Distributed Transacti Own Process AUTHORITY\Netwo	ocalService or Stopped 2\svchost.ex Desktop Sh: c:\window 0 on Coordina c:\window orkService MSIServe: 2\msiexec.ex	0 Disabled e -k netsves aring //s\system32 attor //s\system32 0 r Stopped xe /v	Share Proces Normal mnmsrvc. mnmsrvc.e MSDTC	LocalSyst Stopped exe Running Normal	em Disabled Normal Auto NT	0
NT AUTHORITY\L Messenger Messenge c:\windows\system32 NetMeeting Remote Own Process LocalSystem Distributed Transacti Own Process AUTHORITY\Netw Windows Installer c:\windows\system32 MSSQLSERVER	ocalService or Stopped 2\svchost.ex Desktop Sh: c:\window 0 on Coordina c:\window orkService MSIServe: 2\msiexec.ex	0 Disabled e -k netsves aring vs\system32 ator vs\system32 0 r Stopped xe /v ERVER	Share Process Normal mnmsrvc. mnmsrvc.c. MSDTC msdtc.exe Manual Normal Stopped	LocalSyst Stopped exe Running Normal Share Pro LocalSyst Manual	em Disabled Normal Auto NT cess em	
NT AUTHORITY\Le Messenger Messenge c:\windows\system32 NetMeeting Remote Own Process LocalSystem Distributed Transacti Own Process AUTHORITY\Netw Windows Installer c:\windows\system32 MSSQLSERVER Process C:\progra- LocalSystem	ocalService or Stopped 2\svchost.ex Desktop Sha c:\window 0 on Coordina c:\window orkService MSIServee 2\msiexec.ex MSSQLSI -1\micros~1	0 Disabled e -k netsves arring ys\system32 ator ys\system32 0 r Stopped Ke /v ERVER \mssql\binr	Share Process Normal mnmsrvc. MSDTC msdtc.exe Manual Normal Stopped a sqlservr.ex	LocalSyst Stopped exe Running Normal Share Pro LocalSyst Manual	em Disabled Normal Auto NT cess em Own Normal	
NT AUTHORITY\Le Messenger Messenger c:\windows\system32 NetMeeting Remote Own Process LocalSystem Distributed Transacti Own Process AUTHORITY\Netw Windows Installer c:\windows\system32 MSSQLSERVER Process c:\progra- LocalSystem MSSQLServerADHe	ocalService or Stopped 2\svchost.ex Desktop Sha c:\window 0 on Coordina c:\window orkService MSIServee 2\msiexec.ex MSSQLSI -1\micros~1 0 elper	0 Disabled e -k netsves aring ss\system32 ator ss\system32 o r Stopped ke /v ERVER \mssql\binr MSSQLSe	Share Process Normal mnmsrvc. MSDTC msdtc.exe Manual Normal Stopped absglservr.es	LocalSyst Stopped exe Running Normal Share Prod LocalSyst Manual Ke	em Disabled Normal Auto NT cess em Own	
NT AUTHORITY\Ld Messenger Messenger c:\windows\system32 NetMeeting Remote Own Process LocalSystem Distributed Transacti Own Process AUTHORITY\Netw Windows Installer c:\windows\system32 MSSQLSERVER Process C:\progra- LocalSystem MSSQLServerADHe Manual Own Proces	ocalService or Stopped 2\svchost.ex Desktop Sha c:\window 0 on Coordina c:\window orkService MSIServea 2\msiexec.ex MSSQLSI -1\micros~1 0 elper eess	0 Disabled e -k netsves aring rs\system32 ator rs\system32 0 r Stopped &c /v ERVER \mssql\binr MSSQLSe c:\program	Share Process Normal mnmsrvc. MSDTC msdtc.exe Manual Normal Stopped alsqlservr.es	LocalSyst Stopped exe Running Normal Share Prod LocalSyst Manual Ke	em Disabled Normal Auto NT cess em Own Normal Stopped	
NT AUTHORITY\Le Messenger Messenger c:\windows\system32 NetMeeting Remote Own Process LocalSystem Distributed Transacti Own Process AUTHORITY\Netw Windows Installer c:\windows\system32 MSSQLSERVER Process c:\progra- LocalSystem MSSQLServerADHe	ocalService or Stopped 2\svchost.ex Desktop Sha c:\window 0 on Coordina c:\window orkService MSIServea 2\msiexec.ex MSSQLSI -1\micros~1 0 elper eess	0 Disabled e -k netsves aring rs\system32 ator rs\system32 0 r Stopped &c /v ERVER \mssql\binr MSSQLSe c:\program	Share Process Normal mnmsrvc. MSDTC msdtc.exe Manual Normal Stopped absglservr.es	LocalSyst Stopped exe Running Normal Share Prod LocalSyst Manual Ke	em Disabled Normal Auto NT cess em Own Normal Stopped	
NT AUTHORITY\Le Messenger Messenger c:\windows\system32 NetMeeting Remote Own Process LocalSystem Distributed Transacti Own Process AUTHORITY\Netw Windows Installer c:\windows\system32 MSSQLSERVER Process C:\progra- LocalSystem MSSQLSERVER MSSQLServerADHe Manual Own Proc server\80\tools\binn\system32 Network DDE c:\windows\system32	ocalService or Stopped 2\svchost.ex Desktop Sha c:\window 0 on Coordina c:\window orkService MSIServee 2\msisqLSi -1\micros~1 0 elper eess sqladhlp.exe NetDDE 2\netdde.exe	O Disabled e -k netsves aring ss/system32 ator ss/system32 O r Stopped ke /v ERVER /mssql/binr MSSQLSe c:\program e Normal Stopped Normal	Share Process Normal mnmsrve. MSDTC msdtc.exe Manual Normal Stopped alsqlservr.exerverADHean files/micro LocalSysto Disabled LocalSysto LocalSysto Systomatics and the stopped	LocalSyst Stopped exe Running Normal Share Prod LocalSyst Manual see lper ssoft sql em Share Prod em	em Disabled Normal Auto NT cess em Own Normal Stopped 0 cess 0	
NT AUTHORITY\L Messenger Messenge c:\windows\system32 NetMeeting Remote Own Process LocalSystem Distributed Transacti Own Process AUTHORITY\Netw Windows Installer c:\windows\system32 MSSQLSERVER Process c:\progra- LocalSystem MSSQLSERVEA Manual Own Proc server\80\tools\binn\system32 Network DDE c:\windows\system32 Network DDE DSDM	ocalService or Stopped 2\svchost.ex Desktop Sh: c:\window 0 on Coordina c:\window orkService MSIServe: 2\msiexec.ex MSSQLSI -1\micros~1 0 elper eess sqladhlp.exe NetDDE 2\netdde.exe MNetDDEds	O Disabled e -k netsves aring gs\system32 ator rs\system32 0 r Stopped ke /v ERVER \mssql\binr MSSQLSe c:\program e Normal Stopped b Normal stopped sdm	Share Process Normal mnmsrve. MSDTC msdtc.exe Manual Normal Stopped a sqlservr.e: ErverADHe in files micro Local Syste Disabled Local Syste Stopped	LocalSyst Stopped exe Running Normal Share Prod LocalSyst Manual see lper ssoft sql em Share Prod em Disabled	em Disabled Normal Auto NT cess em Own Normal Stopped 0 cess 0 Share	0
NT AUTHORITY\Le Messenger Messenger c:\windows\system32 NetMeeting Remote Own Process LocalSystem Distributed Transacti Own Process AUTHORITY\Netwo Windows Installer c:\windows\system32 MSSQLSERVER Process c:\progra- LocalSystem MSSQLSERVEA Manual Own Procest Manual Own Procest C:\windows\system32 Network DDE c:\windows\system32 Network DDE DSDM Process c:\windows C:\windows\system32 Network DDE DSDM Process c:\windows\system32 Network DDE DSDM Process c:\windows\system32	ocalService or Stopped 2\svchost.ex Desktop Sh: c:\window 0 on Coordina c:\window orkService MSIServe: 2\msiexec.ex MSSQLSI -1\micros~1 0 elper eses sqladhlp.exe NetDDE 2\netdde.exe MNetDDEds vs\system32	O Disabled e -k netsves aring system32 ator system32 0 r Stopped ke /v ERVER \mssql\binr MSSQLSec:\program e Normal Stopped e Normal sdm \mediane \	Share Process Normal mnmsrve. MSDTC msdtc.exe Manual Normal Stopped ansqlservr.est price of the process of the	LocalSyst Stopped exe Running Normal Share Proc LocalSyst Manual se Iper Ssoft sql em Share Proc em Disabled LocalSyst	em Disabled Normal Auto NT cess em Own Normal Stopped 0 cess 0 Share	
NT AUTHORITY\L Messenger Messenger c:\windows\system32 NetMeeting Remote Own Process LocalSystem Distributed Transacti Own Process AUTHORITY\Netwo Windows Installer c:\windows\system32 MSSQLSERVER Process c:\progra- LocalSystem MSSQLServerADHe Manual Own Proc server\80\tools\binn\struct Network DDE c:\windows\system32 Network DDE DSDM Process c:\window Net Logon Netlogon	ocalService or Stopped 2\svchost.ex Desktop Sh. c:\window 0 on Coordina c:\window orkService MSIServe: 2\msiexec.ex MSSQLSI -1\micros~1 0 elper eses sqladhlp.exe NetDDE 2\netdde.exe MNetDDEds vs\system32 Stopped	O Disabled e -k netsves aring system32 ator system32 O r Stopped ke /v ERVER \mssql\binz C:\program e Normal Stopped b Normal sdm c\netde.exe Manual	Share Process Normal mnmsrve. MSDTC msdtc.exe Manual Normal Stopped ansqlservr.est process of the process of th	LocalSyst Stopped exe Running Normal Share Production LocalSyst Manual exe Iper Share Production Share Production Share Production Share Production Disabled LocalSyst exess	em Disabled Normal Auto NT cess em Own Normal Stopped 0 cess 0 Share em	0
NT AUTHORITY\Le Messenger Messenger c:\windows\system32 NetMeeting Remote Own Process LocalSystem Distributed Transacti Own Process AUTHORITY\Netwo Windows Installer c:\windows\system32 MSSQLSERVER Process c:\progra- LocalSystem MSSQLSERVEA Manual Own Procest Manual Own Procest C:\windows\system32 Network DDE c:\windows\system32 Network DDE DSDM Process c:\windows C:\windows\system32 Network DDE DSDM Process c:\windows\system32 Network DDE DSDM Process c:\windows\system32	ocalService or Stopped 2\svchost.ex Desktop Sh. c:\window 0 on Coordina c:\window orkService MSIServe: 2\msiexec.ex MSSQLSI -1\micros~1 0 elper sess NetDDE 2\netdde.exe MNetDDEds vs\system32 Stopped 2\lsass.exe	O Disabled e -k netsves aring system32 ator system32 0 r Stopped ke /v ERVER \mssql\binr MSSQLSec:\program e Normal Stopped e Normal sdm \mediane \	Share Process Normal mnmsrve. MSDTC msdtc.exe Manual Normal Stopped ansqlservr.est price of the process of the	LocalSyst Stopped exe Running Normal Share Production LocalSyst Manual exe Iper Share Production Share Production Share Production Share Production Disabled LocalSyst exess	em Disabled Normal Auto NT cess em Own Normal Stopped 0 cess 0 Share em 0	0
NT AUTHORITY\L Messenger Messenge c:\windows\system32 NetMeeting Remote Own Process LocalSystem Distributed Transacti Own Process AUTHORITY\Netwo Windows Installer c:\windows\system32 MSSQLSERVER Process c:\progra- LocalSystem MSSQLServerADHe Manual Own Proc server\80\tools\binn\system32 Network DDE DSDN Process c:\windows\system32 Network DDE DSDN Process c:\window Net Logon Netlogon c:\windows\system32 Network Connection c:\windows\system32	ocalService or Stopped 2\svchost.ex Desktop Sh. c:\window 0 on Coordina c:\window orkService MSIServee 2\msiexec.ex MSSQLSI -1\micros~1 0 elper eess sqladhlp.exe MNetDDE 2\netdde.exe MNetDDEd: vs\system32 Stopped 2\lsass.exe s Netman 2\svchost.ex	O Disabled e -k netsves aring vs\system32 ator vs\system32 0 r Stopped xe /v ERVER \mssql\binz e \cdot\normal Stopped xe \cdot\normal Stopped xe \normal	Share Process Normal mnmsrve. MSDTC MSDTC Msdtc.exe Manual Normal Stopped Asqlservr.exe ErverADHe In flest micro LocalSysto Disabled LocalSysto Stopped Normal Share Process LocalSysto LocalSysto Manual Manual LocalSysto Manu	Running Normal Share Proc LocalSyst Manual se Iper Share Proc em Classification Share Proc LocalSyst	em Disabled Normal Auto NT cess em Own Normal Stopped 0 cess 0 Share em 0 cess em	0
NT AUTHORITY\L Messenger Messenge c:\windows\system32 NetMeeting Remote Own Process LocalSystem Distributed Transacti Own Process AUTHORITY\Netwo Windows\system32 MSSQLSERVER Process c:\progra- LocalSystem MSSQLSERVER Process c:\progra- LocalSystem MSSQLServerADHe Manual Own Proc server\80\tools\binn\system32 Network DDE c:\windows\system32 Network DDE DSDM Process c:\window Net Logon Netlogon c:\windows\system32 Network Connection c:\windows\system32 Intel NCS NetService	ocalService or Stopped 2\svchost.ex Desktop Sha c:\window 0 on Coordina c:\window orkService MSIServee 2\msiexec.ev MSSQLSI -1\micros~1 0 elper ness sqladhlp.exe MNetDDE devs\system32 Stopped 2\lsass.exe s Netman 2\svchost.ex e NetSvc	O Disabled e -k netsves aring vs\system32 ator vs\system32 0 r Stopped &e /v ERVER \mssql\binz Stopped e \centre Normal Stopped e Normal sdm \centre Normal Normal Running e -k netsves Stopped	Share Process Normal mnmsrvc. MSDTC msdtc.exe Manual Normal Stopped a sqlservr.exerverADHea files/micro LocalSyste Disabled LocalSyste Stopped Normal Share Processes Manual Share Processes Manual Manual	Running Normal Share Proc LocalSyst Manual se Iper Share Proc Em Share Proc LocalSyst Own Proc	em Disabled Normal Auto NT cess em Own Normal Stopped 0 cess 0 Share em 0 cess em ess	0 0
NT AUTHORITY\L Messenger Messenger c:\windows\system32 NetMeeting Remote Own Process LocalSystem Distributed Transacti Own Process AUTHORITY\Netwo Windows\system32 MSSQLSERVER Process c:\progra- LocalSystem MSSQLSERVER Process c:\progra- LocalSystem MSSQLServerADHe Manual Own Proc server\80\tools\binn\system32 Network DDE c:\windows\system32 Network DDE DSDM Process c:\window Net Logon Netlogon c:\windows\system32 Network Connection c:\windows\system32 Intel NCS NetService c:\program files\intel	ocalService or Stopped 2\svchost.ex Desktop Sha c:\window 0 on Coordina c:\window orkService MSIServee 2\msiexec.es MSSQLSI -1\micros~1 0 elper sess sqladhlp.exe MNetDDE 2\netdde.exe MNetDDE 2\stopped 2\lsass.exe s Netman 2\svchost.ex e NetSvc \ncs\sync\nes\sync\nes\sync\nes \text{NetSyc}\nes\sync\nes sync\nes\sync\n	O Disabled e -k netsves aring salvaystem32 attor vs/ssystem32 O r Stopped &c /v ERVER \mssql\binz MSSQLSe c:\program e Normal Stopped e Normal sdm \manning e -k netsves Stopped etsvc.exe	Share Process Normal mnmsrvc. MSDTC msdtc.exe Manual Normal Stopped a sqlservr.e: erverADHe a files/micro LocalSyste Disabled LocalSyste Stopped Normal Share Process Manual Share Process Manual Manual Normal	Running Normal Share Proc LocalSyst Manual se Iper Disabled LocalSyst em Share Proc Em Disabled LocalSyst ess Em Share Proc LocalSyst County Town LocalSys	em Disabled Normal Auto NT cess em Own Normal Stopped 0 cess 0 Share em 0 cess em ess em ess em	0
NT AUTHORITY\L Messenger Messenger c:\windows\system32 NetMeeting Remote Own Process LocalSystem Distributed Transacti Own Process AUTHORITY\Netwo Windows\system32 MSSQLSERVER Process c:\progra- LocalSystem MSSQLSERVER Process c:\progra- LocalSystem MSSQLServerADHe Manual Own Proc server\80\tools\binn\system32 Network DDE c:\windows\system32 Network DDE DSDM Process c:\window Net Logon Netlogon c:\windows\system32 Network Connection c:\windows\system32 Intel NCS NetService c:\program files\intel Network Location As	ocalService or Stopped 2\svchost.ex Desktop Sha c:\window 0 on Coordina c:\window 0 on Coordina c:\window 0 Stopped 2\ssexe.ex MSSQLSI -1\micros~1 0 Elper Sess sqladhlp.exe NetDDE 2\netdde.exe MNetDDEds vs\system32 Stopped 2\lsass.exe s Netman 2\svchost.ex e NetSvc \ncs\sync\ncs\sync\ncs wareness (N	O Disabled e -k netsves aring salvaystem32 attor vs/ssystem32 or Stopped ke /v ERVER \mssql\binz MSSQLSe c:\program e Normal Stopped e Normal sdm \normal Running e -k netsves Stopped etsvc.exe [ILA)	Share Process Normal mnmsrvc. MSDTC msdtc.exe Manual Normal Stopped a sqlservr.e: erverADHe a files/micro LocalSyste Stopped Normal Share Process Manual Share Process Manual Manual Normal Normal Nla	Running Normal Share Proc LocalSyst Manual Re Iper Disabled LocalSyst em Share Proc Em Disabled LocalSyst ess Em Share Proc LocalSyst County Town LocalSyst County Town LocalSyst Running	em Disabled Normal Auto NT cess em Own Normal Stopped 0 cess 0 Share em 0 cess em Manual	0 0
NT AUTHORITY\L Messenger Messenger c:\windows\system32 NetMeeting Remote Own Process LocalSystem Distributed Transacti Own Process AUTHORITY\Netwo Windows\system32 MSSQLSERVER Process c:\progra- LocalSystem MSSQLSERVER Process c:\progra- LocalSystem MSSQLServerADHe Manual Own Proc server\80\tools\binn\system32 Network DDE c:\windows\system32 Network DDE DSDM Process c:\window Net Logon Netlogon c:\windows\system32 Network Connection c:\windows\system32 Intel NCS NetService c:\program files\intel	ocalService or Stopped 2\svchost.ex Desktop Sha c:\window 0 on Coordina c:\window 0 on Coordina c:\window 0 Stopped 2\ssexe.ex MSSQLSI -1\micros~1 0 Elper Sess sqladhlp.exe NetDDE 2\netdde.exe MNetDDEds vs\system32 Stopped 2\lsass.exe s Netman 2\svchost.ex e NetSvc \ncs\sync\ncs\sync\ncs wareness (N	O Disabled e -k netsves aring salvaystem32 attor vs/ssystem32 O r Stopped &c /v ERVER \mssql\binz MSSQLSe c:\program e Normal Stopped e Normal sdm \manning e -k netsves Stopped etsvc.exe	Share Process Normal mnmsrvc. MSDTC msdtc.exe Manual Normal Stopped a sqlservr.e: erverADHe a files/micro LocalSyste Stopped Normal Share Process Manual Share Process Manual Manual Normal Normal Nla	Running Normal Share Proc LocalSyst Manual Re Iper Disabled LocalSyst em Share Proc Em Disabled LocalSyst ess Em Share Proc LocalSyst County Town LocalSyst County Town LocalSyst Running	em Disabled Normal Auto NT cess em Own Normal Stopped 0 cess 0 Share em 0 cess em Manual	0 0
NT AUTHORITY\L Messenger Messenger c:\windows\system32 NetMeeting Remote Own Process LocalSystem Distributed Transacti Own Process AUTHORITY\Netwo Windows Installer c:\windows\system32 MSSQLSERVER Process c:\progra- LocalSystem MSSQLSERVER Manual Own Process c:\windows\system32 Network DDE c:\windows\system32 Network DDE DSDN Process c:\window Net Logon Netlogon c:\windows\system32 Network Connection c:\windows\system32 Intel NCS NetService c:\program files\intel Network Location Av Share Process LocalSystem File Replication	ocalService or Stopped 2\svchost.ex Desktop Sh: c:\window 0 on Coordina c:\window orkService MSIServe: 2\msiexec.ex MSSQLSI -1\micros~1 0 elper eses sqladhlp.exe NetDDE 2\netdde.exe MNetDDEds vs\system32 Stopped 2\lsass.exe s Netman 2\svchost.ex e NetSvc hnes\sync\ne wareness (N c:\window 0 NtFrs	O Disabled e -k netsves aring system32 ator system32 or Stopped ke /v ERVER hmssql\binr MSSQLSec:\program e Normal Stopped e Normal Stopped shormal Normal Running e -k netsves Stopped etsve.exe LLA)	Share Process Normal mnmsrve. MSDTC msdtc.exe Manual Normal Stopped anselservr.est process of the process of th	Running Normal Share Proc LocalSyst Manual Share Proc LocalSyst Manual Re Iper Share Proc Em Disabled LocalSyst Sess Em Share Proc LocalSyst Coun Proc LocalSyst Running E-k netsvo: Own Proc County State County	em Disabled Normal Auto NT cess em Own Normal Stopped 0 cess 0 Share em 0 cess em em Stopped cess output share em em share em Manual shormal	0 0
NT AUTHORITY\Le Messenger Messenger c:\windows\system32 NetMeeting Remote Own Process LocalSystem Distributed Transacti Own Process AUTHORITY\Netwo Windows Installer c:\windows\system32 MSSQLSERVER Process c:\progra- LocalSystem MSSQLSERVER Manual Own Proc server\80\tools\binn\system32 Network DDE c:\windows\system32 Network DDE DSDM Process c:\window Net Logon Netlogon c:\windows\system32 Network Connection c:\windows\system32 Intel NCS NetService c:\program files\tintel Network Location As Share Process LocalSystem	ocalService or Stopped 2\svchost.ex Desktop Sh: c:\window 0 on Coordina c:\window orkService MSIServe: 2\msiexec.ex MSSQLSI -1\micros~1 0 elper eses sqladhlp.exe NetDDE 2\netdde.exe MNetDDEds vs\system32 Stopped 2\lsass.exe s Netman 2\svchost.ex e NetSvc hnes\sync\ne wareness (N c:\window 0 NtFrs	O Disabled e -k netsves aring salvaystem32 attor salvaystem32 or Stopped ke /v ERVER hmssql\binr MSSQLSe c:\program e Normal Stopped e Normal sdm hormal Running e -k netsves Stopped etsvc.exe ILA)	Share Process Normal mnmsrvc. MSDTC msdtc.exe Manual Normal Stopped a sqlservr.es: erverADHe in files/micro LocalSyste Disabled LocalSyste Stopped Normal Share Process Manual Share Process Manual Mormal Normal Nla syschost.ex	Running Normal Share Proc LocalSyst Manual Share Proc LocalSyst Manual Re Iper Share Proc Em Disabled LocalSyst Sess Em Share Proc LocalSyst Coun Proc LocalSyst Running E-k netsvo: Own Proc County State County	em Disabled Normal Auto NT cess em Own Normal Stopped 0 cess 0 Share em 0 cess em disabled normal	0 0

1	NT LM Security Support Provide	der	NtLmSsp	Stopped	Manual	
		ws\system32		Normal		
	LocalSystem 0	-				
	Removable Storage NtmsSvc	Stopped	Manual	Share Pro	cess	
	c:\windows\system32\svchost.e	xe -k netsvc	s Normal	LocalSyst	em	0
		Running	Auto	Share Pro	cess	
	c:\windows\system32\services.e	exe	Normal	LocalSyst	em	0
1	IPSEC Services PolicyAg		Stopped	Manual	Share	
	Process c:\windows\system3	2\lsass.exe	Normal	LocalSyst	em	0
	Protected Storage Protected		Running	Auto		
	Process c:\windows\system3		Normal	LocalSyst		0
	Remote Access Auto Connection					
		ws\system32	2\svchost.ex	e -k netsvc	s Normal	
	LocalSystem 0					
	Remote Access Connection Ma		RasMan	Stopped		
		ws\system32	2\svchost.ex	e -k netsvc	s Normal	
	LocalSystem 0					
	Remote Desktop Help Session I				Stopped	
	Manual Own Process		vs\system32	\sessmgr.ex	xe	
	Normal LocalSystem	0		Q: 1	D: 11 1	
'	Routing and Remote Access			Stopped	Disabled	
		ws\system32	2\svchost.ex	e -k netsvc	s Normal	
	LocalSystem 0		a. I		CI	
	Remote Registry RemoteR				Share	
	Process c:\windows\system3		ke -k regsvc	Normal	NT	
'	AUTHORITY\LocalService	0	D I .		C 1	
	Remote Procedure Call (RPC) I		RpcLocate		Stopped	
	Manual Own Process		vs\system32	\locator.ex	e Normai	
	NT AUTHORITY\NetworkSer		0		CI	
	Remote Procedure Call (RPC)			Auto	Share	
	Process c:\windows\system3	2\svchost -k	rpess	Normal		
	LocalSystem 0	DC DD	Cr 1	M 1	CI	
	Resultant Set of Policy Provider			Manual	Share	
	Process c:\windows\system3	2\rsopprov.e	exe	Normal		
	LocalSystem 0	Halman	00.00vm	Ctannad	Manual	
	Special Administration Console Share Process c:\window	e neipei ws\system32	Sacsvr	Stopped		
'	LocalSystem 0	ws\system32	2 SVCHOSLEX	.e -k netsve	Sinoiiliai	
	Security Accounts Manager	SamSs	Running	Auto	Share	
	Process c:\windows\system3		Normal	LocalSyst		0
		r Stopped	Manual	Share Pro		U
	c:\windows\system32\scardsvr.	1.1	Ignore	NT	CCSS	
	AUTHORITY\LocalService	0	ignore	111		
		Stopped	Manual	Share Pro	cess	
	c:\windows\system32\svchost.e			LocalSyst		0
	Secondary Logon seclogon		Auto	Share Pro		-
	c:\windows\system32\svchost.e	xe -k netsvo		LocalSyst		0
	System Event Notification	SENS	Running	-	Share	_
	Process c:\windows\system3				.	
	LocalSystem 0		/			
	ServeRAID Manager Agent	ServeRAI	DManager A	Agent	Stopped	
	Manual Own Process		n files\raidn			
	Normal LocalSystem	0				
	Internet Connection Firewall (Id	CF) / Interne	et Connectio	on Sharing ((ICS)	
	SharedAccess Stopped	Disabled	Share Prod	cess		
	c:\windows\system32\svchost.e			LocalSyst	em	0
	Shell Hardware Detection	ShellHW		Running		
	Share Process c:\window	ws\system32		e -k netsvc	s Ignore	
	LocalSystem 0	•			-	
	Print Spooler Spooler	Stopped	Manual	Own Proc	ess	
	c:\windows\system32\spoolsv.e	xe	Normal	LocalSyst	em	0
	SQLSERVERAGENT	SQLSER	VERAGEN'	T	Stopped	
	Manual Own Process	c:\progra~	~1\micros~1	\mssql\bini	n\sqlagent.	exe
	Normal LocalSystem	0		-	-	
	Windows Image Acquisition (W	VIA)	stisvc	Stopped	Disabled	
		ws\system32	2\svchost.ex	e -k imgsvo	e Normal	
	NT AUTHORITY\LocalService	e 0				

Minney & C. Anney Chadan Camp Devides around Manual	A 11 Y7 A 11 Y7
Microsoft Software Shadow Copy Provider swprv Stopped Manual Own Process c:\windows\system32\svchost.exe -k swprv Normal	Accessories All Users: Accessories All Users Accessories\Accessibility All Users: Accessories\Accessibility
LocalSystem 0	Accessories\Accessibility All Users:Accessories\Accessibility All Users
Performance Logs and Alerts SysmonLog Stopped Manual	Accessories\Communications All Users: Accessories\Communications
Own Process c:\windows\system32\smlogsvc.exe Normal	All Users
NT Authority\NetworkService 0	Accessories\Entertainment All Users: Accessories\Entertainment
	All Users All Users
1 2 1 11	
	Accessories\System Tools All Users:Accessories\System Tools
Terminal Services TermService Running Manual Share	All Users
Process c:\windows\system32\svchost.exe -k termsvcs Normal	Administrative Tools All Users: Administrative Tools All Users
LocalSystem 0	Intel Network Adapters All Users: Intel Network Adapters All Users
Themes Themes Stopped Disabled Share Process	Microsoft SQL Server All Users: Microsoft SQL Server All Users
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0	ServeRAID Manager All Users:ServeRAID Manager All Users
Telnet TlntSvr Stopped Disabled Own Process	Startup All Users:Startup All Users
c:\windows\system32\tlntsvr.exe Normal NT AUTHORITY\LocalService 0	Accessories NT AUTHORITY\SYSTEM:Accessories NT
Distributed Link Tracking Server TrkSvr Stopped Disabled Share	AUTHORITY\SYSTEM
Process c:\windows\system32\svchost.exe -k netsvcs Normal	Accessories\Accessibility NT
LocalSystem 0	AUTHORITY\SYSTEM:Accessories\Accessibility NT
Distributed Link Tracking Client TrkWks Stopped Manual Share	AUTHORITY\SYSTEM
Process c:\windows\system32\svchost.exe -k netsvcs Normal	Accessories\Entertainment NT
LocalSystem 0	AUTHORITY\SYSTEM:Accessories\Entertainment NT
Terminal Services Session Directory Tssdis Stopped Disabled	AUTHORITY\SYSTEM
Own Process c:\windows\system32\tssdis.exe Normal	Startup NT AUTHORITY\SYSTEM:Startup NT
LocalSystem 0	AUTHORITY\SYSTEM
Upload Manager uploadmgr Stopped Manual Share Process	Accessories IBMSERVER3\Administrator:Accessories
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0	IBMSERVER3\Administrator
Uninterruptible Power Supply UPS Stopped Manual Own	Accessories\Accessibility
Process c:\windows\system32\ups.exe Normal NT	IBMSERVER3\Administrator:Accessories\Accessibility
AUTHORITY\LocalService 0	IBMSERVER3\Administrator
Virtual Disk Service vds Stopped Manual Own Process	Accessories\Entertainment
c:\windows\system32\vds.exe Normal LocalSystem 0	IBMSERVER3\Administrator:Accessories\Entertainment
Volume Shadow Copy VSS Stopped Manual Own	IBMSERVER3\Administrator
Process c:\windows\system32\vssvc.exe Normal LocalSystem 0	Startup IBMSERVER3\Administrator:Startup
Windows Time W32Time Running Auto Share Process	IBMSERVER3\Administrator
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0	IDINOEK VERS VIGINIIISUUOI
WebClient WebClient Stopped Disabled Share Process	[Startup Programs]
c:\windows\system32\svchost.exe -k localservice Normal NT	[Startup 170grams]
AUTHORITY\LocalService 0	Program Command User NameLocation
WinHTTP Web Proxy Auto-Discovery Service WinHttpAutoProxySvc	desktop desktop.iniNT AUTHORITY\SYSTEM Startup
Stopped Manual Share Process c:\windows\system32\svchost.exe	desktop desktop.iniIBMSERVER3\Administrator Startup
-k localservice Normal NT AUTHORITY\LocalService 0	desktop desktop.ini.DEFAULT Startup
Windows Management Instrumentation winning Running Auto	desktop desktop.iniAll Users Common Startup
Share Process c:\windows\system32\svchost.exe -k netsvcs Ignore	PRONoMgr.exe c:\program files\inte\\ncs\proset\pronomgr.exe
	All Users HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run
	All Osels TIKEW/SOFT WAKE/WICIOSOR/WINDOWS/CUITEREVEISION/KUII
Portable Media Serial Number Service WmdmPmSN Stopped Manual Share Process c:\windows\system32\svchost.exe -k netsvcs	[OLE Registration]
· · · · · · · · · · · · · · · · · · ·	[OLE Registration]
,	Object Local Server
Windows Management Instrumentation Driver Extensions Wmi Stopped Manual Share Process c:\windows\system32\sychost.exe	Object Local Server Sound (OLE2) sndrec32.exe
Stopped Manual Share Process c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0	Media Clip mplay32.exe
WMI Performance Adapter WmiApSrv Stopped Manual	Video Clipmplay32.exe /avi
Own Process c:\windows\system32\wbem\wmiapsrv.exe Normal	MIDI Sequence mplay32.exe /mid
LocalSystem 0	Sound Not Available
Automatic Updates wuauserv Stopped Manual Share Process	Media Clip Not Available
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0	WordPad Document "%programfiles%\windows nt\accessories\wordpad.exe"
Wireless Configuration WZCSVC Stopped Manual Share	Windows Media Services DRM Storage object Not Available
Process c:\windows\system32\svchost.exe -k netsvcs Normal	Bitmap Image mspaint.exe
LocalSystem 0	
	[Windows Error Reporting]
[Program Groups]	T' T D / 'I
Communication Name of N	Time Type Details
Group Name	
Accessories Default User: Accessories Default User	[Internet Settings]
Accessories\Accessibility Default User: Accessories\Accessibility	
Default User	
Accessories\Entertainment Default User:Accessories\Entertainment	
Default User	[Internet Explorer]
Startup Default User:Startup Default User	

[Following are sub-ca [Summary]	ategories of	this main o	category]	mshtmled.dll 6 C:\WINDOWS\system.	5.0.3790.0	444 KB 3/25/2003 8:00:00 AM Microsoft Corporation
[Summary]					5.0.3790.0	
Item Value				C:\WINDOWS\system.		Microsoft Corporation
Version 6.0.3790.0				msident.dll6.0.3790.0 4		3/25/2003 8:00:00 AM
Build 63790				C:\WINDOWS\system3		Microsoft Corporation
Application Path	C:\Program	Files\Inte	rnet Explorer	msidntld.dll 6	5.0.3790.0	15 KB 3/25/2003 8:00:00 AM
Language English (U	nited States)		C:\WINDOWS\system3	32	Microsoft Corporation
Active Printer	Not Availa	ble		msieftp.dll 6.0.3790.0 2	230 KB	3/25/2003 8:00:00 AM
				C:\WINDOWS\system3		Microsoft Corporation
Cipher Strength	128-bit				5.0.3790.0	
Content Advisor	Disabled			C:\WINDOWS\system.		Microsoft Corporation
IEAK Install	No			mstime.dll 6.0.3790.0 4		3/25/2003 8:00:00 AM
rent vv. i a				C:\WINDOWS\system3		Microsoft Corporation
[File Versions]					5.0.3790.0	
Eile Vessies	C:	D-4-	D-4h C	C:\WINDOWS\system.		Microsoft Corporation
File Version	Size 6.0.3790.0	Date 05 VD	Path Company 3/25/2003 8:00:00 AM		5.3.3790.0	
actxprxy.dll C:\WINDOWS\syster			Corporation	C:\WINDOWS\system3 sendmail.dll 6	5.0.3790.0	Intel Corporation 52 KB 3/25/2003 8:00:00 AM
advpack.dll	6.0.3790.0		3/25/2003 8:00:00 AM	C:\WINDOWS\system3		Microsoft Corporation
C:\WINDOWS\syster			Corporation	shdoclc.dll 6.0.3790.0 5		3/25/2003 8:00:00 AM
asctrls.ocx 6.0.3790.0			8:00:00 AM	C:\WINDOWS\system.		Microsoft Corporation
C:\WINDOWS\syster			Corporation	•		1,361 KB 3/25/2003 8:00:00 AM
browselc.dll	6.0.3790.0		3/25/2003 8:00:00 AM	C:\WINDOWS\system.		Microsoft Corporation
C:\WINDOWS\syster			Corporation	•	5.0.3790.0	<u> </u>
browseui.dll			3/25/2003 8:00:00 AM	C:\WINDOWS\system3	32	Microsoft Corporation
C:\WINDOWS\syster	m32	Microsoft	Corporation	shlwapi.dl16.0.3790.0 2	281 KB	3/25/2003 8:00:00 AM
cdfview.dll	6.0.3790.0	144 KB	3/25/2003 8:00:00 AM	C:\WINDOWS\system.	32	Microsoft Corporation
C:\WINDOWS\syster	n32	Microsoft	Corporation	tdc.ocx 1.3.0.3130 5	58 KB	3/25/2003 8:00:00 AM
comctl32.dll	5.82.3790.0)	561 KB 3/25/2003 8:00:00 AM	C:\WINDOWS\system3	32	Microsoft Corporation
C:\WINDOWS\syster	n32	Microsoft	Corporation	url.dll 6.0.3790.0 3		3/25/2003 8:00:00 AM
dxtrans.dll 6.3.3790.0			8:00:00 AM	C:\WINDOWS\system.		Microsoft Corporation
C:\WINDOWS\syster			Corporation	urlmon.dll 6.0.3790.0 5		3/25/2003 8:00:00 AM
dxtmsft.dll 6.3.3790.0			8:00:00 AM	C:\WINDOWS\system.		Microsoft Corporation
C:\WINDOWS\syster			Corporation		5.0.3790.0	
iecont.dll <file miss<="" td=""><td>_</td><td>Not Availa</td><td>able Not Available</td><td>C:\WINDOWS\system3</td><td></td><td>Microsoft Corporation</td></file>	_	Not Availa	able Not Available	C:\WINDOWS\system3		Microsoft Corporation
Not Available iecontlc.dll <file miss<="" td=""><td>Not Availa</td><td>oie Not Availa</td><td>able Not Available</td><td>wininet.dll 6.0.3790.0 6 C:\WINDOWS\system.</td><td></td><td>3/25/2003 8:00:00 AM Microsoft Corporation</td></file>	Not Availa	oie Not Availa	able Not Available	wininet.dll 6.0.3790.0 6 C:\WINDOWS\system.		3/25/2003 8:00:00 AM Microsoft Corporation
Not Available	Not Availa		able Not Available	C.\WINDOWS\system.	32	Microsoft Corporation
iedkcs32.dll	16.0.3790.0		300 KB 3/25/2003 8:00:00 AM	[Connectivity]		
C:\WINDOWS\syster			Corporation	[Connectivity]		
iepeers.dll 6.0.3790.0			8:00:00 AM	Item Value		
C:\WINDOWS\syster			Corporation	Connection Preference		Never dial
iesetup.dll 6.0.3790.0			8:00:00 AM			
C:\WINDOWS\syster		Microsoft	Corporation	LAN Settings		
ieuinit.inf Not Availa	ıble	20 KB	3/25/2003 8:00:00 AM	•		
C:\WINDOWS\syster		Not Availa	able	AutoConfigProxy N	Not Availa	ble
iexplore.exe	6.0.3790.0	90 KB	3/25/2003 8:00:00 AM	AutoProxyDetectMode		Disabled
C:\Program Files\Inte			Microsoft Corporation	AutoConfigURL		
imgutil.dll 5.2.3790.0			8:00:00 AM	Proxy Disabled		
C:\WINDOWS\syster			Corporation	ProxyServer		
inetcpl.cpl 6.0.3790.0			8:00:00 AM	ProxyOverride		
C:\WINDOWS\syster			Corporation	[C1-1		
inetcplc.dl16.0.3790.0 C:\WINDOWS\system			8:00:00 AM Corporation	[Cache]		
inseng.dll 6.0.3790.0			8:00:00 AM	[Following are sub-cate	agories of	this main category 1
C:\WINDOWS\syster			Corporation	[Summary]	egories or	tins main category j
mlang.dll 6.0.3790.0			8:00:00 AM	[Summary]		
C:\WINDOWS\syster			Corporation	Item Value		
msencode.dll	2002.10.4.0		112 KB 3/25/2003 8:00:00 AM		Automatic	
C:\WINDOWS\syster		Not Availa		Temporary Internet File		
mshta.exe 6.0.3790.0			8:00:00 AM			ettings\Temporary Internet Files
C:\WINDOWS\syster			Corporation	Total Disk Space N	Not Availa	ble
mshtml.dll 6.0.3790.0				Available Disk Space N		
C:\WINDOWS\syster			Corporation	Maximum Cache Size N		
mshtml.tlb 6.0.3790.0				Available Cache Size N	Not Availa	ble
C:\WINDOWS\syster	n32	Microsoft	Corporation			
			1	[List of Objects]		

Program File Status CodeBase Array A No cached object information available Array identifier.....A Array size.....140012 MB [Content] Free space......0 MB Number of logical drives......1 [Following are sub-categories of this main category] Number of physical drives.....2 [Summary] Item Value Logical drives in array A Content Advisor Disabled Logical drive.....1 [Personal Certificates] Array letter.....A State.....Okay Issued To Issued By Validity Signature Algorithm RAID level.....1 Data space......69878 MB No personal certificate information available Parity space.....69878 MB [Other People Certificates] Issued To Issued By Validity Signature Algorithm Physical drives in array A No other people certificate information available -----Type.....Hard disk drive [Publishers] Vendor.....IBM-ESXS Product or model number......ST373307LC Name FRU part number.....32P0730 No publisher information available Serial number.....3HZ03BRC Firmware level.....B258 Channel.....1 [Security] SCSI ID.....2 Size.....70006 MB Zone Security Level My Computer Custom State.....Online Local intranet Medium-low Array letter.....A Trusted sites Medium Mirror role.....Primary Internet High PFA error.....No Partitioned.....No Restricted sites High Write-cache mode......Write through Type.....Hard disk drive **Disk Controller Configuration Parameters** Vendor.....IBM-ESXS Product or model number......ST373307LC Integrated Ultra320 Interface FRU part number......06P5760 Serial number.....3HZ038PZ Firmware level.....B258 Channel.....1 July 8, 2003 7:23:35 AM EDT SCSI ID.....4 Size.....70006 MB Configuration summary State.....Online Array letter.....A Mirror role.....Secondary Server name.....ibmserver3 PFA error.....No ServeRAID Manager agent........6.10.13 (1195) Partitioned.....No ServeRAID Manager console......6.10.13 (1195) Write-cache mode......Write through Number of controllers.....2 Operating system......Windows 2003 SCSI channel 1 Configuration information for controller 1 Number of drives.....3 Controller type.....LSI 1030 SCSI initiator ID......7 BIOS version.....5.03.09.00 Firmware version......1.00.14.00 Type.....Hard disk drive Device driver version......1.08.18.00 Vendor.....IBM-ESXS Hot-swap rebuild.....Enabled Product or model number......ST318305LC Data scrubbing.....Enabled FRU part number......06P5369 Auto-synchronization.....Enabled Serial number.....3JK001V1 Unattended mode.....Enabled Firmware level.....B244 Number of arrays.....1 Channel.....1 Number of logical drives.....1 SCSI ID.....0 Number of hot-spare drives.....0 Size.....17357 MB Number of ready drives.....1

StateReady	
PFA errorNo	
PartitionedYes Write-cache modeWrite back	End of the configuration information for controller 2
WITTE-Cacife Hode	
TypeHard disk drive	Mylex AcceleRAID A352 Adapter 1
VendorIBM-ESXS	
Product or model numberST373307LC	GCFVERSION=2.00;
FRU part number32P0730	Begin
Serial number3HZ03BRC	BeginGroup
Firmware levelB258	PhysicalDevice0 = Channel=0, Target=0, Size=17340MB,
Channel1	State=Online, TransferSneed=90MHz, TransferWidth=16Dit
SCSI ID2 Size70006 MB	TransferSpeed=80MHz, TransferWidth=16Bit, MaxTag=16;
StateOnline	PhysicalDevice1 = Channel=0, Target=1, Size=17340MB,
Array letterA	State=Online,
Mirror rolePrimary	TransferSpeed=80MHz, TransferWidth=16Bit,
PFA errorNo	MaxTag=16;
PartitionedNo	PhysicalDevice2 = Channel=0, Target=2, Size=17340MB,
Write-cache modeWrite through	State=Online,
	TransferSpeed=80MHz, TransferWidth=16Bit,
TypeHard disk drive	MaxTag=16;
VendorIBM-ESXS	PhysicalDevice3 = Channel=0, Target=3, Size=17340MB, State=Online,
Product or model numberST373307LC	TransferSpeed=80MHz, TransferWidth=16Bit,
FRU part number06P5760 Serial number3HZ038PZ	MaxTag=16;
Firmware levelB258	PhysicalDevice4 = Channel=0, Target=4, Size=17340MB,
Channel1	State=Online,
SCSI ID4	TransferSpeed=80MHz, TransferWidth=16Bit,
Size70006 MB	MaxTag=16;
StateOnline	PhysicalDevice5 = Channel=0, Target=5, Size=17340MB,
Array letterA	State=Online,
Mirror roleSecondary	TransferSpeed=80MHz, TransferWidth=16Bit,
PFA errorNo	MaxTag=16; PhysicalDevice6 = Channel=0, Target=6, Size=17340MB,
PartitionedNo Write-cache modeWrite through	State=Online.
write-cache modewrite unough	TransferSpeed=80MHz, TransferWidth=16Bit,
TypeEnclosure	MaxTag=16;
VendorIBM	PhysicalDevice7 = Channel=1, Target=8, Size=17340MB,
Product or model number02R0962a S320	State=Online,
Firmware level1	TransferSpeed=80MHz, TransferWidth=16Bit,
Channel1	MaxTag=16;
SCSI ID8	PhysicalDevice8 = Channel=1, Target=9, Size=17340MB, State=Online.
	TransferSpeed=80MHz, TransferWidth=16Bit,
End of the configuration information for controller 1	MaxTag=16;
	PhysicalDevice9 = Channel=1, Target=10, Size=17340MB,
	State=Online,
Configuration information for controller 2	TransferSpeed=80MHz, TransferWidth=16Bit,
	MaxTag=16;
Controller typeLSI 1030	PhysicalDevice10 = Channel=1, Target=11, Size=17340MB,
BIOS version5.03.09.00	State=Online,
Firmware version	TransferSpeed=80MHz, TransferWidth=16Bit, MaxTag=16;
Device driver version	PhysicalDevice11 = Channel=1, Target=12, Size=17340MB,
Data scrubbingEnabled	State=Online,
Auto-synchronizationEnabled	TransferSpeed=80MHz, TransferWidth=16Bit,
Unattended modeEnabled	MaxTag=16;
Number of arrays0	PhysicalDevice12 = Channel=1, Target=13, Size=17340MB,
Number of logical drives0	State=Online,
Number of hot-spare drives0	TransferSpeed=80MHz, TransferWidth=16Bit,
Number of ready drives0	MaxTag=16;
	PhysicalDevice13 = Channel=1, Target=14, Size=17340MB,
CCCL 1 11	State=Online, TransferSpeed=90MHz, TransferWidth=16Dit
SCSI channel 1	TransferSpeed=80MHz, TransferWidth=16Bit, MaxTag=16;
Number of drives0	LogicalDevice0 = StripeSize=64KB, Raid=0, WriteThrough=1,
SCSI initiator ID7	Size=242704MB, BIOSGeometry=8GB,
	, , , , , , , , , , , , , , , , , , , ,

```
(PhysicalDevice0, StartAddress=0MB/0Blocks,
                                                                                   PhysicalDevice0 = Channel=0, Target=0, Size=17340MB,
Size=17336MB/35504128Blocks),
                                                                         State=Online,
                   (PhysicalDevice1, StartAddress=0MB/0Blocks,
                                                                                             TransferSpeed=80MHz, TransferWidth=16Bit,
Size=17336MB/35504128Blocks),
                                                                         MaxTag=16;
                    (PhysicalDevice2, StartAddress=0MB/0Blocks,
                                                                                   PhysicalDevice1 = Channel=0, Target=1, Size=17340MB,
Size=17336MB/35504128Blocks),
                                                                         State=Online,
                    (PhysicalDevice3, StartAddress=0MB/0Blocks,
                                                                                             TransferSpeed=80MHz, TransferWidth=16Bit,
Size=17336MB/35504128Blocks),
                                                                         MaxTag=16;
                    (PhysicalDevice4, StartAddress=0MB/0Blocks,
                                                                                   PhysicalDevice2 = Channel=0, Target=2, Size=17340MB,
Size=17336MB/35504128Blocks),
                                                                         State=Online,
                    (PhysicalDevice5, StartAddress=0MB/0Blocks,
                                                                                             TransferSpeed=80MHz, TransferWidth=16Bit,
Size=17336MB/35504128Blocks),
                                                                         MaxTag=16;
                   (PhysicalDevice6, StartAddress=0MB/0Blocks,
                                                                                   PhysicalDevice3 = Channel=0, Target=3, Size=17340MB,
Size=17336MB/35504128Blocks),
                                                                         State=Online,
                    (PhysicalDevice7, StartAddress=0MB/0Blocks,
                                                                                             TransferSpeed=80MHz, TransferWidth=16Bit,
Size=17336MB/35504128Blocks),
                                                                         MaxTag=16;
                    (PhysicalDevice8, StartAddress=0MB/0Blocks,
                                                                                   PhysicalDevice4 = Channel=0, Target=4, Size=17340MB,
Size=17336MB/35504128Blocks),
                                                                         State=Online,
                                                                                             TransferSpeed=80MHz, TransferWidth=16Bit,
                    (PhysicalDevice9, StartAddress=0MB/0Blocks,
Size=17336MB/35504128Blocks),
                                                                         MaxTag=16;
                    (PhysicalDevice10, StartAddress=0MB/0Blocks,
                                                                                   PhysicalDevice5 = Channel=0, Target=5, Size=17340MB,
Size=17336MB/35504128Blocks),
                                                                         State=Online,
                                                                                             TransferSpeed=80MHz, TransferWidth=16Bit,
                    (PhysicalDevice11, StartAddress=0MB/0Blocks,
Size=17336MB/35504128Blocks),
                                                                         MaxTag=16;
                   (PhysicalDevice12, StartAddress=0MB/0Blocks,
                                                                                   PhysicalDevice6 = Channel=0, Target=6, Size=17340MB,
Size=17336MB/35504128Blocks),
                                                                         State=Online,
                    (PhysicalDevice13, StartAddress=0MB/0Blocks,
                                                                                             TransferSpeed=80MHz, TransferWidth=16Bit,
Size=17336MB/35504128Blocks);
                                                                         MaxTag=16;
EndGroup
                                                                                   PhysicalDevice7 = Channel=1, Target=8, Size=17340MB,
BeginControllerParameter
                                                                         State=Online,
          ControllerName = AcceleRAID 352;
                                                                                             TransferSpeed=80MHz, TransferWidth=16Bit,
          ControllerType = 30;
                                                                         MaxTag=16;
          FirmwareVersion = 7.01:
                                                                                   PhysicalDevice8 = Channel=1, Target=9, Size=17340MB,
                                                                         State=Online,
         CacheLineSize = 8KB;
          AutomaticRebuildRate = 50;
                                                                                             TransferSpeed=80MHz, TransferWidth=16Bit,
         BackgroundInitializeRate = 50;
                                                                         MaxTag=16;
          ConsistencyCheckRate = 50;
                                                                                   PhysicalDevice9 = Channel=1, Target=10, Size=17340MB,
          MORERate = 50;
                                                                         State=Online,
         InitiatorID = 7;
                                                                                             TransferSpeed=80MHz, TransferWidth=16Bit,
          DevicesPerSpin = 2;
                                                                         MaxTag=16;
          SequentialDelay = 6S;
                                                                                   PhysicalDevice10 = Channel=1, Target=11, Size=17340MB,
         EnableDriveSizing = 0;
                                                                         State=Online,
          EnableClustering = 0;
                                                                                             TransferSpeed=80MHz, TransferWidth=16Bit,
          EnableBGInit = 1;
                                                                         MaxTag=16;
          EnableBiosLoadDelay = 0;
                                                                                   PhysicalDevice11 = Channel=1, Target=12, Size=17340MB,
          EnableForcedUnitAccess = 0;
                                                                         State=Online,
          DisableBios = 0;
                                                                                             TransferSpeed=80MHz, TransferWidth=16Bit,
          EnableCDROMBoot = 0;
                                                                         MaxTag=16;
          EnableStorageWorks = 0;
                                                                                   PhysicalDevice12 = Channel=1, Target=13, Size=17340MB,
          EnableSAFTE = 0;
                                                                         State=Online,
          EnableSES = 0;
                                                                                             TransferSpeed=80MHz, TransferWidth=16Bit,
         EnableARM = 1;
                                                                         MaxTag=16;
          EnableOFM = 1;
                                                                                   PhysicalDevice13 = Channel=1, Target=14, Size=17340MB,
          OEMCode = 0;
                                                                         State=Online,
          StartupOption = 0;
                                                                                             TransferSpeed=80MHz, TransferWidth=16Bit,
                                                                         MaxTag=16;
          EnableTempOffline = 0;
          EnablePatrolRead = 0;
                                                                                   LogicalDevice0 = StripeSize=64KB, Raid=0, WriteThrough=1,
          EnableSmartMode = 0:
                                                                         Size=242704MB, BIOSGeometry=8GB,
          DlyBtwnIterations = 0;
                                                                                             (PhysicalDevice0, StartAddress=0MB/0Blocks,
         SmartScanInterval = 0:
                                                                         Size=17336MB/35504128Blocks),
EndControllerParameter
                                                                                             (PhysicalDevice1, StartAddress=0MB/0Blocks,
End
                                                                         Size=17336MB/35504128Blocks),
                                                                                             (PhysicalDevice2, StartAddress=0MB/0Blocks,
                                                                         Size=17336MB/35504128Blocks),
Mylex AcceleRAID A352 Adapter 2
                                                                                             (PhysicalDevice3, StartAddress=0MB/0Blocks,
                                                                         Size=17336MB/35504128Blocks),
GCFVERSION=2.00;
                                                                                             (PhysicalDevice4, StartAddress=0MB/0Blocks,
Begin
                                                                         Size=17336MB/35504128Blocks),
```

BeginGroup

```
PhysicalDevice2 = Channel=0, Target=2, Size=17340MB,
                   (PhysicalDevice5, StartAddress=0MB/0Blocks,
Size=17336MB/35504128Blocks),
                                                                         State=Online,
                    (PhysicalDevice6, StartAddress=0MB/0Blocks,
                                                                                             TransferSpeed=80MHz, TransferWidth=16Bit,
Size=17336MB/35504128Blocks),
                                                                         MaxTag=16;
                    (PhysicalDevice7, StartAddress=0MB/0Blocks,
                                                                                   PhysicalDevice3 = Channel=0, Target=3, Size=17340MB,
Size=17336MB/35504128Blocks),
                                                                         State=Online,
                    (PhysicalDevice8, StartAddress=0MB/0Blocks,
                                                                                             TransferSpeed=80MHz, TransferWidth=16Bit,
Size=17336MB/35504128Blocks),
                                                                         MaxTag=16;
                    (PhysicalDevice9, StartAddress=0MB/0Blocks,
                                                                                   PhysicalDevice4 = Channel=0, Target=4, Size=17340MB,
Size=17336MB/35504128Blocks),
                                                                         State=Online,
                    (PhysicalDevice10, StartAddress=0MB/0Blocks,
                                                                                             TransferSpeed=80MHz, TransferWidth=16Bit,
Size=17336MB/35504128Blocks),
                                                                         MaxTag=16;
                   (PhysicalDevice11, StartAddress=0MB/0Blocks,
                                                                                   PhysicalDevice5 = Channel=0, Target=5, Size=17340MB,
Size=17336MB/35504128Blocks),
                                                                         State=Online,
                   (PhysicalDevice12, StartAddress=0MB/0Blocks,
                                                                                             TransferSpeed=80MHz, TransferWidth=16Bit,
Size=17336MB/35504128Blocks),
                                                                         MaxTag=16;
                    (PhysicalDevice13, StartAddress=0MB/0Blocks,
                                                                                   PhysicalDevice6 = Channel=0, Target=6, Size=17340MB,
Size=17336MB/35504128Blocks);
                                                                         State=Online,
EndGroup
                                                                                             TransferSpeed=80MHz, TransferWidth=16Bit,
BeginControllerParameter
                                                                         MaxTag=16;
          ControllerName = AcceleRAID 352;
                                                                                   PhysicalDevice7 = Channel=1, Target=8, Size=17340MB,
          ControllerType = 30;
                                                                         State=Online,
          Firmware Version = 7.01;
                                                                                             TransferSpeed=80MHz, TransferWidth=16Bit,
         CacheLineSize = 8KB;
                                                                         MaxTag=16;
          AutomaticRebuildRate = 50;
                                                                                   PhysicalDevice8 = Channel=1, Target=9, Size=17340MB,
          BackgroundInitializeRate = 50;
                                                                         State=Online,
          ConsistencyCheckRate = 50;
                                                                                             TransferSpeed=80MHz, TransferWidth=16Bit,
         MORERate = 50:
                                                                         MaxTag=16;
          InitiatorID = 7;
                                                                                   PhysicalDevice9 = Channel=1, Target=10, Size=17340MB,
          DevicesPerSpin = 2;
                                                                         State=Online,
          Sequential Delay = 6S;
                                                                                             TransferSpeed=80MHz, TransferWidth=16Bit,
          EnableDriveSizing = 0;
                                                                         MaxTag=16;
          EnableClustering = 0;
                                                                                   PhysicalDevice10 = Channel=1, Target=11, Size=17340MB,
          EnableBGInit = 1;
                                                                         State=Online,
          EnableBiosLoadDelay = 0;
                                                                                             TransferSpeed=80MHz, TransferWidth=16Bit,
         EnableForcedUnitAccess = 0;
                                                                         MaxTag=16;
          DisableBios = 0
                                                                                   PhysicalDevice11 = Channel=1, Target=12, Size=17340MB,
          EnableCDROMBoot = 0;
                                                                         State=Online,
          EnableStorageWorks = 0;
                                                                                             TransferSpeed=80MHz, TransferWidth=16Bit,
         EnableSAFTE = 0;
                                                                         MaxTag=16;
          EnableSES = 0;
                                                                                   PhysicalDevice12 = Channel=1, Target=13, Size=17340MB,
          EnableARM = 1;
                                                                         State=Online,
          EnableOFM = 1;
                                                                                             TransferSpeed=80MHz, TransferWidth=16Bit,
          OEMCode = 0;
                                                                         MaxTag=16;
          StartupOption = 0;
                                                                                   PhysicalDevice13 = Channel=1, Target=14, Size=17340MB,
          EnableTempOffline = 0;
                                                                         State=Online,
         EnablePatrolRead = 0;
                                                                                             TransferSpeed=80MHz, TransferWidth=16Bit,
          EnableSmartMode = 0;
                                                                         MaxTag=16;
          DlyBtwnIterations = 0;
                                                                                   LogicalDevice0 = StripeSize=64KB, Raid=0, WriteThrough=1,
          SmartScanInterval = 0;
                                                                         Size=242704MB, BIOSGeometry=8GB,
EndControllerParameter
                                                                                             (PhysicalDevice0, StartAddress=0MB/0Blocks,
                                                                         Size=17336MB/35504128Blocks),
                                                                                             (PhysicalDevice1, StartAddress=0MB/0Blocks,
                                                                         Size=17336MB/35504128Blocks),
                                                                                             (PhysicalDevice2, StartAddress=0MB/0Blocks,
Mylex AcceleRAID A352 Adapter 3
                                                                         Size=17336MB/35504128Blocks),
                                                                                             (PhysicalDevice3, StartAddress=0MB/0Blocks,
GCFVERSION=2.00;
                                                                         Size=17336MB/35504128Blocks),
Begin
                                                                                             (PhysicalDevice4, StartAddress=0MB/0Blocks,
BeginGroup
                                                                         Size=17336MB/35504128Blocks),
         PhysicalDevice0 = Channel=0, Target=0, Size=17340MB,
                                                                                             (PhysicalDevice5, StartAddress=0MB/0Blocks,
State=Online,
                                                                         Size=17336MB/35504128Blocks),
                    TransferSpeed=80MHz, TransferWidth=16Bit,
                                                                                             (PhysicalDevice6, StartAddress=0MB/0Blocks,
MaxTag=16;
                                                                         Size=17336MB/35504128Blocks),
          PhysicalDevice1 = Channel=0, Target=1, Size=17340MB,
                                                                                             (PhysicalDevice7, StartAddress=0MB/0Blocks,
State=Online,
                                                                         Size=17336MB/35504128Blocks),
                    TransferSpeed=80MHz, TransferWidth=16Bit,
                                                                                             (PhysicalDevice8, StartAddress=0MB/0Blocks,
MaxTag=16;
```

Size=17336MB/35504128Blocks),

```
[HKEY LOCAL MACHINE\SOFTWARE\Microsoft\TPCC]
                               (PhysicalDevice9, StartAddress=0MB/0Blocks,
Size=17336MB/35504128Blocks),
                                                                                                                     "Path"="c:\\inetpub\\wwwroot\\"
                                                                                                                     "NumberOfDeliveryThreads"=dword:0x20
                               (PhysicalDevice10, StartAddress=0MB/0Blocks,
                                                                                                                     "MaxConnections"=dword:0x3c8c
Size=17336MB/35504128Blocks),
                                                                                                                    "MaxPendingDeliveries"=dword:0x6a4
                               (PhysicalDevice11, StartAddress=0MB/0Blocks,
                                                                                                                     "DB_Protocol"="ODBC"
Size=17336MB/35504128Blocks),
                                                                                                                     "TxnMonitor"="COM'
                               (PhysicalDevice12, StartAddress=0MB/0Blocks,
                                                                                                                     "DbServer"="ibmserver3"
Size=17336MB/35504128Blocks),
                                                                                                                    "DbName"="tpcc"
"DbUser"="sa"
                               (PhysicalDevice13, StartAddress=0MB/0Blocks,
Size=17336MB/35504128Blocks);
                                                                                                                     "DbPassword"=""
EndGroup
BeginControllerParameter
                                                                                                                     "COM SinglePool"="YES"
                ControllerName = AcceleRAID 352;
               ControllerType = 30;
                                                                                                                    Microsoft Internet Information Service Registry Parameters
                Firmware Version = 7.01;
                CacheLineSize = 8KB;
                                                                                                                    [HKEY LOCAL MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo]
                AutomaticRebuildRate = 50;
                BackgroundInitializeRate = 50;
                                                                                                                    [HKEY\_LOCAL\_MACHINE \ SYSTEM \ Current Control Set \ Services \ In et Info \ P
                ConsistencyCheckRate = 50;
                                                                                                                    arameters
                MORERate = 50;
                                                                                                                    "ListenBackLog"=dword:0x19
                                                                                                                     "DispatchEntries"=hex(7):4c,00,44,00,41,00,50,00,53,00,56,00,43,00,00,00,00,
               InitiatorID = 7;
                DevicesPerSpin = 2;
               Sequential Delay = 6S:
                                                                                                                    "PoolThreadLimit"=dword:0xbe
               EnableDriveSizing = 0;
                                                                                                                    "ThreadTimeout"=dword:0x15180
                EnableClustering = 0;
               EnableBGInit = 1;
                                                                                                                    [HKEY LOCAL MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo\P
               EnableBiosLoadDelay = 0;
                                                                                                                    erformance]
               EnableForcedUnitAccess = 0;
                                                                                                                    "Library"="infoctrs.dll"
                                                                                                                     "Open"="OpenINFOPerformanceData"
               DisableBios = 0;
                EnableCDROMBoot = 0;
                                                                                                                     "Close"="CloseINFOPerformanceData"
                EnableStorageWorks = 0;
                                                                                                                     "Collect"="CollectINFOPerformanceData"
                EnableSAFTE = 0;
                                                                                                                     "Last Counter"=dword:0x842
               EnableSES = 0:
                                                                                                                    "Last Help"=dword:0x843
                EnableARM = 1;
                                                                                                                     "First Counter"=dword:0x802
               EnableOFM = 1;
                OEMCode = 0;
                                                                                                                    World Wide Web Service Registry Parameters
                StartupOption = 0;
               EnableTempOffline = 0;
                                                                                                                    [HKEY LOCAL MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC]
                EnablePatrolRead = 0;
                                                                                                                     "Type"=dword:0x20
                EnableSmartMode = 0:
                                                                                                                     "Start"=dword:0x2
                DlvBtwnIterations = 0;
                                                                                                                     "ErrorControl"=dword:0x1
                                                                                                                     "ImagePath"=hex(2):43,00,3a,00,5c,00,57,00,49,00,4e,00,4e,00,54,00,5c,00,53,
               SmartScanInterval = 0;
EndControllerParameter
                                                                                                                      79,00,73,00,74,00,65,00,6d,00,33,00,32,00,5c,00,69,00,6e,00,65,00,74,00,73,\\
End
                                                                                                                      00,72,00,76,00,5c,00,69,00,6e,00,65,00,74,00,69,00,6e,00,66,00,6f,00,2e,00
                                                                                                                      65,00,78,00,65,00,00,00
Microsoft Windows 2000 Server with COM+
                                                                                                                     "DisplayName"="World Wide Web Publishing Service"
                                                                                                                     "DependOnService"=hex(7):49,00,49,00,53,00,41,00,44,00,4d,00,49,00,4e,00,0
                                                                                                                    0,00,\
Client Configuration Parameters
                                                                                                                     00,00
                                                                                                                     "DependOnGroup"=hex(7):00,00
                                                                                                                     "ObjectName"="LocalSystem"
Client Configuration Parameters
                                                                                                                     "Description"="Provides Web connectivity and administration through the
                                                                                                                    Internet Information Services snap-in."
COM+ Settings
                                                                                                                    [HKEY\_LOCAL\_MACHINE \SYSTEM \Current Control Set \Services \W3SVC \CARROW \C
TPCC.AllTxns:
     Activation:
         Enable Object Pooling selected
                                                                                                                    "NOTE"="This is for backward compatibility only."
         Minimum Pool Size: 250
                                                                                                                    [HKEY\_LOCAL\_MACHINE \SYSTEM \Current Control Set \Services \W3SVC \]
         Maximum Pool Size: 250
                                                                                                                    ASP\Parameters]
         Creating Timeout: 60,000
         Enable Object Construction
                                                                                                                    [HKEY LOCAL MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\P
         Enable Just in Time Activation
                                                                                                                    arameters]
     Concurrency:
                                                                                                                    "MajorVersion"=dword:00000005
         Concurrency Required
```

TPCC Application Registry Parameters

"MinorVersion"=dword:00000000

"InstallPath"="C:\\WINNT\\System32\\inetsrv"

'CertMapList"="C:\\WINNT\\System32\\inetsrv\\iiscrmap.dll"

"AccessDeniedMessage"="Error: Access is Denied."

"Filter DLLs"="

"LogFileDirectory"="C:\\WINNT\\System32\\LogFiles"

"AcceptExOutstanding"=dword:00000028

[HKEY LOCAL MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\P arameters\ADCLaunch]

[HKEY LOCAL MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\P arameters\ADCLaunch\AdvancedDataFactory]

[HKEY LOCAL MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\P arameters\ADCLaunch\RDSServer.DataFactory]

[HKEY LOCAL MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\P arameters\Script Map]

[HKEY LOCAL MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\P arameters\Virtual Roots1

"/"="c:\\inetpub\\wwwroot,,205"

"/Scripts"="c:\\inetpub\\scripts,,1"

"/IISAdmin"="C:\\WINNT\\System32\\inetsrv\\iisadmin,,1"

"/IISSamples"="c:\\inetpub\\iissamples,,1"

"/MSADC"="c:\\program files\\common files\\system\\msadc,,1"

"/IISHelp"="c:\\winnt\\help\\iishelp,,1"

"/ vti bin"="C:\\Program Files\\Common Files\\Microsoft Shared\\Web Server Extensions\\40\\isapi,,1"

"/Rpc"="C:\\WINNT\\System32\\RpcProxy,,1"

"/Printers"="C:\\WINNT\\web\\printers,,201"

[HKEY LOCAL MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\P erformance]

"Library"="w3ctrs.dll"

"Open"="OpenW3PerformanceData"

"Close"="CloseW3PerformanceData"

"Collect"="CollectW3PerformanceData"

"Last Counter"=dword:000008f2

"Last Help"=dword:000008f3

"First Counter"=dword:00000850

"First Help"=dword:00000851

 $[HKEY_LOCAL_MACHINE \SYSTEM \Current Control Set \Services \W3SVC \SYSTEM \Current Control Set \SYSTEM \Current \Curre$

"Security"=hex:01,00,14,80,a0,00,00,00,ac,00,00,00,14,00,00,00,30,00,00,00

00,1c,00,01,00,00,00,02,80,14,00,ff,01,0f,00,01,01,00,00,00,00,00,00,01,00,00,\ 00,00,02,00,70,00,04,00,00,00,00,00,18,00,fd,01,02,00,01,01,00,00,00,00,00,\ 05.12.00.00,00,74.00,6f,00,00,00,1c,00,ff,01,0f,00,01,02,00,00,00,00,00,05,20,00,00,00,20,02,00,00,72,00,73,00,00,00,18,00,8d,01,02,00,01,01,00,00,00,00,05,20,00,00,00,23,02,00,00,72,00,73,00,01,01,00,00,00,00,00,05,12,00,00,00,01,01,00,00,00,00,00,05,12,00,00,00

[HKEY LOCAL MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\E

"0"="Root\\LEGACY W3SVC\\0000"

"Count"=dword:00000001

"NextInstance"=dword:00000001

Client System Information Report

System Information report written at: 07/07/2003 07:41:00 PM [System Information]

[Following are sub-categories of this main category]

[System Summary]

Item Value

OS Name Microsoft Windows 2000 Server Version 5.0.2195 Service Pack 2 Build 2195 OS Manufacturer Microsoft Corporation

System Name CLIENT10 System Manufacturer IBM System Model -[8647xxx]-X86-based PC System Type

Processor x86 Family 15 Model 2 Stepping 7 GenuineIntel ~37419 Mhz Processor x86 Family 15 Model 2 Stepping 7 GenuineIntel ~37419 Mhz **BIOS Version**)Phoenix - Award WorkstationBIOS v6.00PG

Windows Directory C:\WINNT

C:\WINNT\System32 System Directory \Device\Harddisk0\Partition1 **Boot Device**

Locale United States

User NameCLIENT10\Administrator Time Zone Eastern Daylight Time

Total Physical Memory 1.048.044 KB Available Physical Memory 881,364 KB

Total Virtual Memory 4,092,700 KB

Available Virtual Memory 3,851,972 KB

Page File Space 3,044,656 KB

Page File C:\pagefile.sys

[Hardware Resources]

[Following are sub-categories of this main category]

[Conflicts/Sharing]

Resource Device

No conflicted/shared resources

[DMA]

Channel Device Status

Direct memory access controller OK 2 Standard floppy disk controller OK

[Forced Hardware]

PNP Device ID No Forced Hardware

[I/O]

Address Range Device Status 0x0000-0x0CF7 PCI bus OK

0x0000-0x0CF7 Direct memory access controller OK

0x0D00-0x40B7 PCI bus OK 0x40B9-0xFFFF PCI bus OK

0x9000-0xBFFF PCI standard PCI-to-PCI bridge OK 0x9000-0xBFFF PCI standard PCI-to-PCI bridge OK 0x9000-0xBFFF PCI standard PCI-to-PCI bridge OK

0x9000-0xBFFF Intel(R) PRO/100 S Dual Port Server Adapter #5

0xB000-0xBFFF PCI standard PCI-to-PCI bridge OK

0xB000-0xBFFF PCI standard PCI-to-PCI bridge OK 0xB000-0xBFFF

OK

Intel(R) PRO/100 S Dual Port Server Adapter #3

0xB400-0xB43F

Intel(R) PRO/100 S Dual Port Server Adapter #4

OK

0x9400-0x943F

Intel(R) PRO/100 S Dual Port Server Adapter #6

OK

0xA000-0xA0FF LSI Logic 1020/1030 Ultra320 SCSI Adapter

OK

0xA400-0xA4FF	LSI Logic 1020/1030 Ultra320 SCSI Adapt	er	[Memory]	
OK			[:::::::::::	
0xC000-0xC0FF	ATI Technologies Inc. RAGE XL PCI	OK	Range Device Status	
0x03B0-0x03BB	ATI Technologies Inc. RAGE XL PCI	OK	0xC8000-0xCBFFF System bo	
0x03C0-0x03DF	ATI Technologies Inc. RAGE XL PCI	OK	0xF0000-0xF7FFF System be	
0x0A79-0x0A79	ISAPNP Read Data Port OK ISAPNP Read Data Port OK		0xF8000-0xFBFFF System by	
0x0279-0x0279 0x0274-0x0277	ISAPNP Read Data Port OK ISAPNP Read Data Port OK		0xFC000-0xFFFFF System bo 0x3FFF0000-0x3FFFFFFF	System board OK
0xF000-0xF00F	Standard Dual Channel PCI IDE Controller	OK	0x0000-0x9FFFF System bo	
0x01F0-0x01F7	Primary IDE Channel OK	OK	0x100000-0x3FFEFFFF	System board OK
0x03F6-0x03F6	Primary IDE Channel OK		0xFEC00000-0xFECFFFFF	System board OK
0x0170-0x0177	Secondary IDE Channel OK		0xFEE00000-0xFEEFFFFF	System board OK
0x0376-0x0376	Secondary IDE Channel OK		0xFFB00000-0xFFB7FFFF	System board OK
0x5000-0x501F	PCI Device OK		0xFFF00000-0xFFFFFFF	System board OK
0x4000-0x40BF	Motherboard resources OK		0xE0000-0xEFFFF System be	
0x0010-0x001F	Motherboard resources OK		0xFFB80000-0xFFBFFFFF	Intel(r) 82802 Firmware Hub Device
0x0022-0x003F	Motherboard resources OK		OK	OV
0x0044-0x005F 0x0062-0x0063	Motherboard resources OK Motherboard resources OK		0xA0000-0xBFFFF PCI bus 0xA0000-0xBFFFF ATI Tech	OK nologies Inc. RAGE XL PCI OK
0x0065-0x006F	Motherboard resources OK Motherboard resources		0xC0000-0xDFFFF PCI bus	OK
0x0074-0x007F	Motherboard resources OK		0x40000000-0xFEBFFFFF	PCI bus OK
0x0091-0x0093	Motherboard resources OK		0xEC000000-0xEFFFFFF	PCI standard PCI-to-PCI bridge OK
0x00A2-0x00BF	Motherboard resources OK		0xF0000000-0xF2FFFFFF	PCI standard PCI-to-PCI bridge OK
0x00E0-0x00EF	Motherboard resources OK		0xF0000000-0xF2FFFFFF	PCI standard PCI-to-PCI bridge OK
0x04D0-0x04D1	Motherboard resources OK		0xF2201000-0xF2201FFF	System Interrupt Controller OK
0x0800-0x087F	Motherboard resources OK		0xF2000000-0xF21FFFFF	PCI standard PCI-to-PCI bridge OK
0x0020-0x0021	Programmable interrupt controller	OK	0xF2000000-0xF21FFFFF	PCI standard PCI-to-PCI bridge OK
0x00A0-0x00A1	Programmable interrupt controller	OK	0xF2000000-0xF21FFFFF	Intel(R) PRO/100 S Dual Port Server Adapter
0x0080-0x0090 0x0094-0x009F	Direct memory access controller OK Direct memory access controller OK		#3 OK 0xF2100000-0xF210FFFF	Broadcom NetXtreme Gigabit Ethernet
0x00C0-0x00DF	Direct memory access controller OK		OK	Broadcom NetAttenie Gigaott Ethernet
0x0040-0x0043	System timer OK		0xF2041000-0xF2041FFF	Intel(R) PRO/100 S Dual Port Server Adapter
0x0070-0x0073	System CMOS/real time clock OK		#3 OK	mod(rt) r res, roo s s am r ore server rrampter
0x0061-0x0061	System speaker OK		0xF2040000-0xF2040FFF	Intel(R) PRO/100 S Dual Port Server Adapter
0x00F0-0x00FF	Numeric data processor OK		#4 OK	
0x03F2-0x03F5	Standard floppy disk controller OK		0xF2020000-0xF203FFFF	Intel(R) PRO/100 S Dual Port Server Adapter
0x03F7-0x03F7	Standard floppy disk controller OK		#4 OK	
0x03F8-0x03FF	Communications Port (COM1) OK		0xF2200000-0xF2200FFF	System Interrupt Controller OK
0x02F8-0x02FF 0x0378-0x037F	Communications Port (COM2) OK Printer Port (LPT1) OK		0xF1000000-0xF10FFFFF 0xF1000000-0xF10FFFFF	PCI standard PCI-to-PCI bridge OK
0x0060-0x0060	Standard 101/102-Key or Microsoft Natural	PS/2	#5 OK	Intel(R) PRO/100 S Dual Port Server Adapter
Keyboard OK	Standard 101/102-Rey of Wilcrosoft Natural	13/2	0xF1041000-0xF1041FFF	Intel(R) PRO/100 S Dual Port Server Adapter
0x0064-0x0064	Standard 101/102-Key or Microsoft Natural	PS/2	#5 OK	medity i not rought reaches
Keyboard OK			0xF1040000-0xF1040FFF	Intel(R) PRO/100 S Dual Port Server Adapter
0x40B8-0x40B8	Not Available OK		#6 OK	•
			0xF1020000-0xF103FFFF	Intel(R) PRO/100 S Dual Port Server Adapter
[IRQs]			#6 OK	
TDO N. I	ъ.		0xF1110000-0xF111FFFF	LSI Logic 1020/1030 Ultra320 SCSI Adapter
IRQ Number 9 Microso	Device		OK	I CL I:- 1020/1020 I II 220 CCCI A I
	ft ACPI-Compliant System m NetXtreme Gigabit Ethernet		0xF1100000-0xF110FFFF OK	LSI Logic 1020/1030 Ultra320 SCSI Adapter
	PRO/100 S Dual Port Server Adapter #3		0xF1120000-0xF112FFFF	LSI Logic 1020/1030 Ultra320 SCSI Adapter
	PRO/100 S Dual Port Server Adapter #4		OK	Est Esgle 1020/1030 Chias20 Sest Haupter
· /	PRO/100 S Dual Port Server Adapter #5		0xF1130000-0xF113FFFF	LSI Logic 1020/1030 Ultra320 SCSI Adapter
	PRO/100 S Dual Port Server Adapter #6		OK	
	ic 1020/1030 Ultra320 SCSI Adapter		0xF3000000-0xF3FFFFFF	ATI Technologies Inc. RAGE XL PCI
	ic 1020/1030 Ultra320 SCSI Adapter		OK	
	hnologies Inc. RAGE XL PCI		0xF5000000-0xF5000FFF	ATI Technologies Inc. RAGE XL PCI
•	IDE Channel		OK	Standard Dual Channel PCI IDE Controller
10 PCI Dev 8 System 0	CMOS/real time clock		0xFEBFFC00-0xFEBFFFFF OK	Standard Duar Chainier PCI IDE Controller
	e data processor			
	I floppy disk controller		[Components]	
	nications Port (COM1)			
3 Commun	nications Port (COM2)		[Following are sub-categories o	f this main category]
	mpatible Mouse			
1 Standard	l 101/102-Key or Microsoft Natural PS/2 Keyb	oard	[Multimedia]	

Following are sub-	categories of this main category]			PNP Device ID	0	
[Audio Codecs]					T-ST CD-ROM GCR-8480B	1.02
[5&74A80B&0&0.0		
Codec Manufact	urer Description	Status	File			
Version Size	Creation Date					
-	sg723.acm Microsoft Corporatio			[Sound Device]		
	T\System32\MSG723.ACM	4.4.3385	106.77 KB			
(109,328 bytes)	2/10/2003 5:01:31 PM			Item Value		
c:\winnt\system32\lh			22 27 VD	No sound devices		
OK C:\WINN (34,064 bytes)	T\System32\LHACM.ACM 2/10/2003 5:01:32 PM	4.4.3363	33.27 KB	[Dieplay]		
	c25 32.ax Intel Corporation	Indeo® at	ıdio software	[Display]		
	T\System32\IAC25 32.AX	2.05.53	195.00 KB	Item Value		
(199,680 bytes)	12/7/1999 7:00:00 AM	2.03.33	175.00 KB		hnologies Inc. RAGE XL PCI	
	soft32.acm DSP GROUP, INC.		OK	PNP Device ID	iniciogies inc. la laz 12 12 1	
C:\WINNT\System3		9.27 KB (9,488 bytes)		EV_4752&SUBSYS_02401014&REV_2	7\4&1A671D0C8
12/7/1999 7:00:00 A		`	, ,	0&08F0		
c:\winnt\system32\m	sg711.acm Microsoft Corporatio	n		Adapter Type	ATI RAGE XL PCI, ATI Technologie	s Inc. compatible
OK C:\WINN	T\System32\MSG711.ACM	5.00.2134	.1	Adapter Description	ATI Technologies Inc. RAGE XL PCI	
10.27 KB (10,512 by	rtes) 12/7/1999 7:00:00 A	M		Adapter RAM	8.00 MB (8,388,608 bytes)	
	naadp32.acmMicrosoft Corporatio	n		Installed Drivers	atidrab.dll	
OK C:\WINN	T\System32\IMAADP32.ACM	5.00.2134	.1	Driver Version	5.00.2179.1	
16.27 KB (16,656 by				INF File display.i	, ,	
	sgsm32.acm Microsoft Corporatio			Color Planes	1	
	T\System32\MSGSM32.ACM	5.00.2134	.1	Color Table Entries		
22.27 KB (22,800 by				Resolution 1024 x 7	'68 x 60 hertz	
	sadp32.acm Microsoft Corporatio		1	Bits/Pixel 16		
	T\System32\MSADP32.ACM	5.00.2134	.1			
14.77 KB (15,120 by	tes) 12/7/1999 7:00:00 A	NI .		[Infrared]		
[Video Codesa]				[mnared]		
[Video Codecs]				Item Value		
Codec Manufact	urer Description	Status	File	No infrared devices		
Version Size	Creation Date	Status	THE	No initiated devices		
c:\winnt\system32\ir:		Indeo® vi	deo 5 10	[Input]		
	T\System32\IR50 32.DLL	R.5.10.15		[mpar]		
737.50 KB (755,200				[Following are sub-	-categories of this main category]	
	sh261.drv Microsoft Corporatio			[· · · · · · · · · · · · · · · · · · ·		
	T\System32\MSH261.DRV		163.77 KB	[Keyboard]		
(167,696 bytes)	2/10/2003 5:01:31 PM			. , ,		
c:\winnt\system32\m	sh263.drv Microsoft Corporatio	n		Item Value		
OK C:\WINN	T\System32\MSH263.DRV	4.4.3385	252.27 KB	Description	Standard 101/102-Key or Microsoft Na	atural PS/2
(258,320 bytes)	2/10/2003 5:01:10 PM			Keyboard		
c:\winnt\system32\ic			OK		d (101- or 102-key)	
C:\WINNT\System3		108.00 KI	3 (110,592	Layout 0000040		
,	7:00:00 AM			PNP Device ID	ACPI\PNP0303\3&13C0B0C5&0	
	svidc32.dll Microsoft Corporatio			NumberOfFunction	Keys 12	
	T\System32\MSVIDC32.DLL	5.00.2134	.1			
27.27 KB (27,920 by		M	OV	[Daintin - Danis -1		
c:\winnt\system32\ir. C:\WINNT\System3.	_ \ \ / 1	ahla	OK 194.50 KB	[Pointing Device]		
(199,168 bytes)	2\IR32_32.DLL Not Avail 12/7/1999 7:00:00 AM	able	194.30 KD	Item Value		
c:\winnt\system32\m		n		Hardware Type	PS/2 Compatible Mouse	
•	T\System32\MSRLE32.DLL	5.00.2134	.1	Number of Buttons		
10.77 KB (11,024 by	-		.1	Status OK	3	
10.77 125 (11,021.0)	12///1999 /100100111			PNP Device ID	ACPI\PNP0F13\3&13C0B0C5&0	
[CD-ROM]				Power Management		
[Double Click Thres	11	
Item Value				Handedness	Right Handed Operation	
Drive D:					S	
Description	CD-ROM Drive					
Media Loaded	False			[Modem]		
Media Type	CD-ROM					
Name HL-DT-S	T CD-ROM GCR-8480B			Item Value		
Manufacturer	(Standard CD-ROM drives)			No modems		
Status OK						
Transfer Rate	Not Available			[Network]		

Index [Following are sub-categories of this main category] Service Name Raspti IP Address Not Available [Adapter] IP Subnet Not Available Default IP Gateway Not Available Value DHCP Enabled Item False [00000000] RAS Async Adapter Name DHCP Server Not Available DHCP Lease Expires Not Available Adapter Type Not Available Product Name RAS Async Adapter DHCP Lease Obtained Not Available Installed True MAC Address Not Available PNP Device ID Not Available Service Name Raspti Last Reset 7/7/2003 3:36:55 PM Driver c:\winnt\system32\drivers\raspti.sys (16880, 5.00.2146.1) Index [00000004] WAN Miniport (IP) Service Name AsyncMac Name IP Address Not Available Adapter Type Not Available WAN Miniport (IP) IP Subnet Not Available Product Name Default IP Gateway Not Available Installed True ROOT\MS NDISWANIP\0000 **DHCP** Enabled False PNP Device ID DHCP Server Not Available Last Reset 7/7/2003 3:36:55 PM DHCP Lease Expires Not Available Index 4 DHCP Lease Obtained Not Available Service Name NdisWan MAC Address Not Available IP Address Not Available Service Name Not Available IP Subnet Not Available Default IP Gateway Not Available Name [00000001] WAN Miniport (L2TP) **DHCP** Enabled False Adapter Type Not Available DHCP Server Not Available Product Name WAN Miniport (L2TP) DHCP Lease Expires Not Available DHCP Lease Obtained Not Available Installed True PNP Device ID ROOT\MS L2TPMINIPORT\0000 MAC Address Not Available Last Reset 7/7/2003 3:36:55 PM NdisWan Service Name Index Driver c:\winnt\system32\drivers\ndiswan.sys (90096, 5.00.2195.2779) Service Name Rasl2tp IP Address Not Available Name [00000005] Broadcom NetXtreme Gigabit Ethernet IP Subnet Not Available Adapter Type Ethernet 802.3 Default IP Gateway Not Available Product Name Broadcom NetXtreme Gigabit Ethernet **DHCP** Enabled Installed True DHCP Server PNP Device ID Not Available DHCP Lease Expires Not Available PCI\VEN 14E4&DEV 16A7&SUBSYS 026F1014&REV 02\5&121CC7C2 DHCP Lease Obtained Not Available &0&08E810 MAC Address Not Available Last Reset 7/7/2003 3:36:55 PM Service Name Rasl2tp Index c:\winnt\system32\drivers\rasl2tp.sys (50800, 5.00.2179.1) b57w2k Driver Service Name IP Address 192.168.132.251 IP Subnet 255.255.255.0 [00000002] WAN Miniport (PPTP) Name Wide Area Network (WAN) Default IP Gateway Adapter Type Not Available WAN Miniport (PPTP) DHCP Enabled Product Name False Installed True DHCP Server Not Available ROOT\MS PPTPMINIPORT\0000 PNP Device ID DHCP Lease Expires Not Available Last Reset 7/7/2003 3:36:55 PM DHCP Lease Obtained Not Available MAC Address 00:10:DC:76:E2:B3 Index PptpMiniport Service Name Service Name b57w2k IRQ Number IP Address Not Available c:\winnt\system32\drivers\b57w2k.sys (78352, 2.78.0.0) IP Subnet Not Available Driver Default IP Gateway Not Available **DHCP** Enabled False Name [00000008] Intel(R) PRO/100 S Dual Port Server Adapter DHCP Server Not Available Adapter Type Ethernet 802.3 DHCP Lease Expires Not Available Product Name Intel(R) PRO/100 S Dual Port Server Adapter DHCP Lease Obtained Not Available Installed True MAC Address 50:50:54:50:30:30 PNP Device ID PCI\VEN_8086&DEV_1229&SUBSYS_10158086&REV_0D\6&27CA6BFE Service Name PptpMiniport Driver c:\winnt\system32\drivers\raspptp.sys (47856, 5.00.2160.1) &0&2010E810 Last Reset 7/7/2003 3:36:55 PM [00000003] Direct Parallel Index Name 8 Adapter Type Not Available Service Name Direct Parallel IP Address 192.168.120.10 Product Name Installed True IP Subnet 255.255.255.0 ROOT\MS_PTIMINIPORT\0000 PNP Device ID Default IP Gateway Not Available Last Reset 7/7/2003 3:36:55 PM **DHCP** Enabled

DHCP Server **DHCP** Enabled Not Available False DHCP Lease Expires Not Available **DHCP Server** Not Available DHCP Lease Expires Not Available DHCP Lease Obtained Not Available MAC Address 00:02:B3:A8:B1:DB DHCP Lease Obtained Not Available Service Name E100B MAC Address 00:02:B3:B8:F9:AE IRO Number Service Name E100B 52 0xB000-0xBFFF I/O Port IRQ Number 29 c:\winnt\system32\drivers\e100bnt5.sys (141584, 6.04.14.0000) 0x9400-0x943F Driver I/O Port Driver c:\winnt\system32\drivers\e100bnt5.sys (141584, 6.04.14.0000) Name [00000009] Intel(R) PRO/100 S Dual Port Server Adapter Adapter Type Ethernet 802.3 Product Name Intel(R) PRO/100 S Dual Port Server Adapter [Protocol] Installed True Value PNP Device ID Item PCI\VEN_8086&DEV_1229&SUBSYS_10158086&REV_0D\6&27CA6BFE MSAFD Tcpip [TCP/IP] Name &0&2810E810 ConnectionlessService False Last Reset 7/7/2003 3:36:55 PM GuaranteesDelivery True Index GuaranteesSequencing True Service Name E100B MaximumAddressSize 16 bytes IP Address 192.168.110.10 MaximumMessageSize 0 bytes IP Subnet 255.255.255.0 MessageOriented False Default IP Gateway Not Available MinimumAddressSize16 bytes DHCP Enabled False PseudoStreamOriented False DHCP Server Not Available SupportsBroadcasting False DHCP Lease Expires Not Available SupportsConnectData False DHCP Lease Obtained Not Available SupportsDisconnectData False MAC Address 00:02:B3:A8:B1:DC SupportsEncryption False E100B SupportsExpeditedData Service Name True IRQ Number 53 SupportsGracefulClosing True 0xB400-0xB43F I/O Port SupportsGuaranteedBandwidth False Driver c:\winnt\system32\drivers\e100bnt5.sys (141584, 6.04.14.0000) SupportsMulticasting False Name [00000010] Intel(R) PRO/100 S Dual Port Server Adapter Name MSAFD Tcpip [UDP/IP] Adapter Type Ethernet 802.3 ConnectionlessService True GuaranteesDelivery False Product Name Intel(R) PRO/100 S Dual Port Server Adapter Installed True GuaranteesSequencing False PNP Device ID MaximumAddressSize 16 bytes PCI\VEN 8086&DEV 1229&SUBSYS 10158086&REV 0D\6&34E7E7F4& MaximumMessageSize 65467 bytes MessageOriented 0&2010F810 True Last Reset 7/7/2003 3:36:55 PM MinimumAddressSize16 bytes Index **PseudoStreamOriented** False SupportsBroadcasting True Service Name E100B IP Address 192.168.140.10 SupportsConnectData False SupportsDisconnectData IP Subnet 255.255.255.0 False Default IP Gateway Not Available SupportsEncryption False **DHCP** Enabled SupportsExpeditedData False False **DHCP Server** Not Available SupportsGracefulClosing False DHCP Lease Expires Not Available SupportsGuaranteedBandwidth False DHCP Lease Obtained SupportsMulticasting True Not Available MAC Address 00:02:B3:B8:F9:AD E100B RSVP UDP Service Provider Service Name Name IRQ Number ConnectionlessService 28 True 0x9000-0xBFFF I/O Port GuaranteesDelivery False Driver c:\winnt\system32\drivers\e100bnt5.sys (141584, 6.04.14.0000) GuaranteesSequencing False MaximumAddressSize 16 bytes [00000011] Intel(R) PRO/100 S Dual Port Server Adapter MaximumMessageSize 65467 bytes Name Adapter Type Ethernet 802.3 MessageOriented Product Name Intel(R) PRO/100 S Dual Port Server Adapter MinimumAddressSize16 bytes PseudoStreamOriented Installed True False PNP Device ID SupportsBroadcasting True PCI\VEN 8086&DEV 1229&SUBSYS 10158086&REV 0D\6&34E7E7F4& SupportsConnectData False 0&2810F810 SupportsDisconnectData False Last Reset 7/7/2003 3:36:55 PM SupportsEncryption True Index 11 SupportsExpeditedData False E100B SupportsGracefulClosing Service Name False IP Address 192.168.130.10 SupportsGuaranteedBandwidth False IP Subnet 255.255.255.0 SupportsMulticasting True Default IP Gateway Not Available

N DCVD TCD C T		D	F-1
Name RSVP TCP Service F		PseudoStreamOriented	False
ConnectionlessService	False	SupportsBroadcasting False	
GuaranteesDelivery True		SupportsConnectData False	
GuaranteesSequencing	True	SupportsDisconnectData	False
MaximumAddressSize	16 bytes	SupportsEncryption False	
MaximumMessageSize	0 bytes	SupportsExpeditedData	False
MessageOriented False		SupportsGracefulClosing	False
MinimumAddressSize16 bytes		SupportsGuaranteedBandwidth	False
PseudoStreamOriented	False	SupportsMulticasting False	
SupportsBroadcasting False		supportantiousing ruise	
SupportsConnectData False		Name MSAFD NetBIOS	
SupportsDisconnectData SupportsDisconnectData	False		005E 00D1 4000 D001 227D4220500C)]
	raise	_ 11_\	995F-88D1-4909-B081-227DA339588C}]
SupportsEncryption True	_	DATAGRAM 8	
SupportsExpeditedData	True	ConnectionlessService	True
SupportsGracefulClosing	True	GuaranteesDelivery False	
SupportsGuaranteedBandwidth	False	GuaranteesSequencing	False
SupportsMulticasting False		MaximumAddressSize	20 bytes
		MaximumMessageSize	64000 bytes
Name MSAFD NetBIOS		MessageOriented True	, and the second se
	.0D6-E62D-4DBF-958D-3332D7E2724D}]	MinimumAddressSize20 bytes	
SEQPACKET 0	10D0 E02D 4DB1 730D 3332D7E2724D}]	PseudoStreamOriented	False
ConnectionlessService	False		raise
	raise	SupportsBroadcasting True	
GuaranteesDelivery True	_	SupportsConnectData False	
GuaranteesSequencing	True	SupportsDisconnectData	False
MaximumAddressSize	20 bytes	SupportsEncryption False	
MaximumMessageSize	64000 bytes	SupportsExpeditedData	False
MessageOriented True		SupportsGracefulClosing	False
MinimumAddressSize20 bytes		SupportsGuaranteedBandwidth	False
PseudoStreamOriented	False	SupportsMulticasting False	
SupportsBroadcasting False		a upp or more more grown	
SupportsConnectData False		Name MSAFD NetBIOS	
SupportsDisconnectData SupportsDisconnectData	False		AEA 651D 45D2 DD0C 2D8ADDED0697)]
	raise		AF4-651D-45B3-BB0C-2B8ADDFB0687}]
SupportsEncryption False	7.1	SEQPACKET 7	7.1
SupportsExpeditedData	False	ConnectionlessService	False
SupportsGracefulClosing	False	GuaranteesDelivery True	
SupportsGuaranteedBandwidth	False	GuaranteesSequencing	True
SupportsMulticasting False		MaximumAddressSize	20 bytes
		MaximumMessageSize	64000 bytes
Name MSAFD NetBIOS		MessageOriented True	, and the second se
	.0D6-E62D-4DBF-958D-3332D7E2724D}]	MinimumAddressSize20 bytes	
DATAGRAM 0	1000 1000 1000 333207127210,]	PseudoStreamOriented	False
ConnectionlessService	True	SupportsBroadcasting False	T disc
	Truc	SupportsConnectData False	
GuaranteesDelivery False	F 1		F 1
GuaranteesSequencing	False	SupportsDisconnectData	False
MaximumAddressSize	20 bytes	SupportsEncryption False	
MaximumMessageSize	64000 bytes	SupportsExpeditedData	False
MessageOriented True		SupportsGracefulClosing	False
MinimumAddressSize20 bytes		SupportsGuaranteedBandwidth	False
PseudoStreamOriented	False	SupportsMulticasting False	
SupportsBroadcasting True			
SupportsConnectData False		Name MSAFD NetBIOS	
SupportsDisconnectData	False		AF4-651D-45B3-BB0C-2B8ADDFB0687}]
SupportsEncryption False	1 disc	DATAGRAM 7	AI 4-031D-43D3-DB0C-2D0ADDI B000/}]
	F-1	ConnectionlessService	T
SupportsExpeditedData	False		True
SupportsGracefulClosing	False	GuaranteesDelivery False	
SupportsGuaranteedBandwidth	False	GuaranteesSequencing	False
SupportsMulticasting False		MaximumAddressSize	20 bytes
		MaximumMessageSize	64000 bytes
Name MSAFD NetBIOS		MessageOriented True	
[\Device\NetBT Tcpip {CCEF9	995F-88D1-4909-B081-227DA339588C}]	MinimumAddressSize20 bytes	
SEQPACKET 8	7,1	PseudoStreamOriented	False
ConnectionlessService	False	SupportsBroadcasting True	
GuaranteesDelivery True	1 11100	SupportsConnectData False	
	True		Folgo
GuaranteesSequencing	True	SupportsDisconnectData	False
MaximumAddressSize	20 bytes	SupportsEncryption False	
MaximumMessageSize	64000 bytes	SupportsExpeditedData	False
Manage Contract of Tours	010000 0 1000		
MessageOriented True	0.000 0,000	SupportsGracefulClosing	False
MinimumAddressSize20 bytes			

SupportsMulticasting False MaximumAddressSize 20 bytes MaximumMessageSize 64000 bytes Name MSAFD NetBIOS MessageOriented [\Device\NetBT_Tcpip_{0AE55295-FD97-4F03-8CE0-2962253F5166}] MinimumAddressSize20 bytes SEOPACKET 6 **PseudoStreamOriented** False ConnectionlessService False SupportsBroadcasting True SupportsConnectData False GuaranteesDelivery True SupportsDisconnectData GuaranteesSequencing True False MaximumAddressSize 20 bytes SupportsEncryption False MaximumMessageSize 64000 bytes SupportsExpeditedData False MessageOriented True SupportsGracefulClosing False MinimumAddressSize20 bytes SupportsGuaranteedBandwidth False PseudoStreamOriented False SupportsMulticasting False SupportsBroadcasting False SupportsConnectData False MSAFD NetBIOS SupportsDisconnectData False [\Device\NetBT_Tcpip_{BAD679FF-C376-4B6F-ABCE-3FA6DEA10679}] SupportsEncryption False SEQPACKET 1 Supports Expedited DataFalse ConnectionlessService False SupportsGracefulClosing False GuaranteesDelivery True SupportsGuaranteedBandwidth False GuaranteesSequencing True SupportsMulticasting False MaximumAddressSize 20 bytes MaximumMessageSize 64000 bytes Name MSAFD NetBIOS MessageOriented True $[\Device\NetBT_Tcpip_\{0AE55295\text{-}FD97\text{-}4F03\text{-}8CE0\text{-}2962253F5166}\}]$ MinimumAddressSize20 bytes DATAGRAM 6 PseudoStreamOriented False ConnectionlessService SupportsBroadcasting False True GuaranteesDelivery False SupportsConnectData False SupportsDisconnectData GuaranteesSequencing False False MaximumAddressSize 20 bytes SupportsEncryption False MaximumMessageSize SupportsExpeditedData 64000 bytes False MessageOriented SupportsGracefulClosing False SupportsGuaranteedBandwidth MinimumAddressSize20 bytes False **PseudoStreamOriented** False SupportsMulticasting False SupportsBroadcasting True SupportsConnectData False Name MSAFD NetBIOS SupportsDisconnectData False [\Device\NetBT_Tcpip_{BAD679FF-C376-4B6F-ABCE-3FA6DEA10679}] SupportsEncryption False DATAGRAM 1 SupportsExpeditedData False ConnectionlessService True SupportsGracefulClosing GuaranteesDelivery False False SupportsGuaranteedBandwidth False GuaranteesSequencing False SupportsMulticasting False MaximumAddressSize 20 bytes 64000 bytes MaximumMessageSize MSAFD NetBIOS MessageOriented Name $[\Device\NetBT_Tcpip_\{863ED3FC-3C10-4E07-A1A7-C4300734A1C1\}]$ MinimumAddressSize20 bytes SEQPACKET 5 PseudoStreamOriented False ConnectionlessService False SupportsBroadcasting True GuaranteesDelivery True SupportsConnectData False GuaranteesSequencing True SupportsDisconnectData False MaximumAddressSize SupportsEncryption False 20 bytes MaximumMessageSize 64000 bytes SupportsExpeditedData False MessageOriented SupportsGracefulClosing True False MinimumAddressSize20 bytes SupportsGuaranteedBandwidth False PseudoStreamOriented False SupportsMulticasting False SupportsBroadcasting False MSAFD NetBIOS SupportsConnectData False Name SupportsDisconnectData False [\Device\NetBT_Tcpip_{746F2941-9397-47E7-8C77-C78A77DFC894}] SupportsEncryption False SEQPACKET 2 Supports Expedited DataFalse ConnectionlessService False SupportsGracefulClosing GuaranteesDelivery True False Supports Guaranteed BandwidthFalse GuaranteesSequencing True SupportsMulticasting False MaximumAddressSize 20 bytes MaximumMessageSize 64000 bytes MSAFD NetBIOS MessageOriented Name [\Device\NetBT_Tcpip_{863ED3FC-3C10-4E07-A1A7-C4300734A1C1}] MinimumAddressSize20 bytes DATAGRAM 5 PseudoStreamOriented False ConnectionlessService True SupportsBroadcasting False GuaranteesDelivery False SupportsConnectData False GuaranteesSequencing False SupportsDisconnectData False

	ncryption False			XMit on X		0
	xpeditedData	False		low Contro		
	racefulClosing	False		ULL Byte		
	uaranteedBandwidth	False		low Contro		
SupportsM	Iulticasting False		DSR Sens	-	0	T. 11
	Maren M. Drog			v Control T	- 1	Enable
Name	MSAFD NetBIOS	A 44 0405 4575 0655 650 45577600 () 1	EOF Char		0	
		941-9397-47E7-8C77-C78A77DFC894}]		lace Chara		0
DATAGR.		T		lacement E		0
	nlessService	True	Event Cha Parity Cha		0	
	sDelivery False sSequencing	False	RTS Flow			Enable
	AddressSize	20 bytes	XOff Cha		уре 19	Enable
	MessageSize	64000 bytes		t Threshold		
MessageO		04000 bytes	XOn Chai		17	
	AddressSize20 bytes		XOnXMit			
	eamOriented	False		InFlow Co		0
	roadcasting True			OutFlow		0
	onnectData False		IRQ Num		4	
	isconnectData	False	I/O Port	0x03F8-0	0x03FF	
	ncryption False		Driver	c:\winnt\s	system32\dri	vers\serial.sys (62416, 5.00.2195.2780)
	xpeditedData	False				
	racefulClosing	False	Name	COM2		
SupportsG	uaranteedBandwidth	False	Status	OK		
SupportsM	Iulticasting False		PNP Devi	ce ID	ACPI\PNF	P0501\2
			Maximum	Input Buf	fer Size	0
			Maximum	Output B	uffer Size	False
[WinSock]			Settable B	aud Rate	True	
			Settable D		True	
Item	Value			low Contro		
File	c:\winnt\system32\wi	nsock.dll	Settable P	-	True	
Version	3.10			arity Chec		
Size	2.80 KB (2,864 bytes)	Settable S		True	
P.1) : () (22)	122 111	Settable R		True	
File	c:\winnt\system32\ws	sock32.dll	Supports		True	
Version	5.00.2195.2871	·)		16 Bit Mod		F-1
Size	21.27 KB (21,776 byt	les)	Baud Rate	Special Ch	aracters	False
			Bits/Byte			
[Ports]			Stop Bits			
[1 Orts]			Parity	None		
[Followin	g are sub-categories of	f this main category l	Busy	0		
[1 Ollowing	g are sub categories of	tins main category j	,	id/Write or	Error	0
[Serial]				ode Enable		
[~*****]				XMit on X		0
Item	Value			low Contro		
Name	COM1			ULL Byte		
Status	OK		DSR Outf	low Contro	ol 0	
PNP Device	ce ID ACPI\PNI	20501\1	DSR Sens	itivity	0	
Maximum	Input Buffer Size	0	DTR Flov	v Control T	Гуре	Enable
Maximum	Output Buffer Size	False	EOF Char	acter	0	
Settable Ba	aud Rate True			lace Chara		0
Settable Da			Error Rep		Enabled	0
	low Control True		Event Cha		0	
Settable Pa	•		Parity Che			
	arity Check True		RTS Flow		- 1	Enable
Settable St			XOff Cha		19	
Settable R				t Threshold		
Supports 1			XOn Chai		17	
	6 Bit Mode False Special Characters	False		Threshold InFlow Co		0
Baud Rate		1 4150		OutFlow		0
Bits/Byte			IRQ Num		3	V
Stop Bits			I/O Port	0x02F8-0		
Parity	None		Driver			vers\serial.sys (62416, 5.00.2195.2780)
Busy	0			J. , , , 111111 /k	., 5001115 2 (df1	
	d/Write on Error	0				
	de Enabled -1		[Parallel]			

c:\winnt\system32\drivers\symmpi.sys (38512, 1.08.22.00) Item Value LSI Logic 1020/1030 Ultra320 SCSI Adapter LPT1 Name Name ACPI\PNP0400\3&13C0B0C5&0 LSI Logic 1020/1030 Ultra320 SCSI Adapter PNP Device ID Caption SYMMPI Driver Status PNP Device ID [Storage] PCI\VEN 1000&DEV 0030&SUBSYS 10001014&REV 07\5&21593F33&0 [Following are sub-categories of this main category] &19F810 Device ID PCI\VEN_1000&DEV_0030&SUBSYS_10001014&REV_07\5&21593F33&0 [Drives] &19F810 Item Value Device Map Not Available Not Available Drive Index Α. Description 3 1/2 Inch Floppy Drive Max Number Controlled Not Available IRQ Number 33 Drive C: I/O Port 0xA400-0xA4FF Local Fixed Disk Description Driver c:\winnt\system32\drivers\symmpi.sys (38512, 1.08.22.00) Compressed False File System NTFS 16.94 GB (18,194,284,544 bytes) [Printing] Size Free Space 13.01 GB (13,971,148,800 bytes) Volume Name Name Port Name Server Name Volume Serial Number 08674715 No printing information Partition Disk #0, Partition #0 Partition Size 16.94 GB (18,194,287,104 bytes) [Problem Devices] Starting Offset 32256 bytes PNP Device ID Error Code Drive Description Disk drive Device Drive Manufacturer (Standard disk drives) PCI Device PCI\VEN_8086&DEV_2551&SUBSYS_25511014&REV_03\3&13C0B0C5& Drive Model Drive BytesPerSector 512 0&01 Drive MediaLoaded True System Interrupt Controller Drive MediaType Fixed hard disk media PCI\VEN_8086&DEV_1461&SUBSYS_00000000&REV_03\4&4E4AD31&0 **Drive Partitions** &E010 Drive SCSIBus 0 System Interrupt Controller Drive SCSILogicalUnit 0 PCI\VEN 8086&DEV 1461&SUBSYS 00000000&REV 03\4&4E4AD31&0 Drive SCSIPort &F010 Drive SCSITargetId 5 PCI Device Drive SectorsPerTrack PCI\VEN_8086&DEV_2554&SUBSYS_25541014&REV_03\3&13C0B0C5& 63 Drive Size 18194319360 bytes 0&11 Drive TotalCylinders 2212 PCI Device Drive TotalSectors 35535780 PCI\VEN_8086&DEV_24C3&SUBSYS_027A1014&REV_02\3&13C0B0C5 Drive TotalTracks 564060 &0&FB Drive TracksPerCylinder Not Available ACPI\IBM37D6\2&DABA3FF&0 255 28 ACPI\ASF0001\2&DABA3FF&0 28 Not Available E: Drive Description Network Connection [USB] Provider Name \\192.168.132.253\e\$ PNP Device ID Device No USB Devices [SCSI] [Software Environment] Item LSI Logic 1020/1030 Ultra320 SCSI Adapter [Following are sub-categories of this main category] Name LSI Logic 1020/1030 Ultra320 SCSI Adapter Caption Driver SYMMPI [Drivers] OK Status PNP Device ID Name Description File Type Started Start Mode PCI\VEN 1000&DEV 0030&SUBSYS 10001014&REV 07\5&21593F33&0 State Status Error Control Accept Pause Accept Stop &18F810 abiosdsk Abiosdsk Not Available Kernel Driver False Device ID Disabled Stopped OK False False Ignore PCI\VEN 1000&DEV 0030&SUBSYS 10001014&REV 07\5&21593F33&0 abp480n5 abp480n5 Not Available Kernel Driver False &18F810 Disabled False Stopped OK Normal False Device Map Not Available acpi Microsoft ACPI Driver c:\winnt\system32\drivers\acpi.sys Index Not Available Kernel Driver True **Boot** Running OK Normal Max Number Controlled Not Available False True IRQ Number 32

Driver

I/O Port 0xA000-0xA0FF

acpiec	ACPIEC	c:\winnt\sv	/stem32\dri	ivers\acpiec	SVS	Kernel	cnafcalm	cpqfcalm	Not Avail	able	Kernel Dri	iver	False
Driver	False	Disabled		OK	Normal	False	Disabled	1 1	OK	Normal	False Kernel Dri	False	False
False	adpu160m	Not Availa	able	Kernel Dr	iver	False	Disabled	1 1	OK	Normal	False	False	raise
Disabled		OK	Normal	False	False	i uisc		dac960nt			Kernel Dri		False
afd		vorking Sup	port Enviro	onment			Disabled	Stopped	OK	Normal	False	False	
	ystem32\dr				True	Auto		deckzpsx			Kernel Dri		False
Running		Normal	False	True		F 1	Disabled		OK	Normal	False	False	ъ:
aha154x		Not Availa OK		Kernel Dri False		False	dfsdriver True			ystem32\dri OK	vers\dfs.sys Normal	s File Syster False	
Disabled aic116x	Stopped aic116x	Not Availa	Normal able	Kernel Dr	False iver	False	disk	Boot Disk Drive	Running		ystem32\dri		True
Disabled		OK	Normal	False	False	i aisc	Kernel Dr		True	Boot	Running		Normal
aic78u2	aic78u2	Not Availa		Kernel Dr		False	False	True					
Disabled	Stopped	OK	Normal	False	False		diskperf	Diskperf	c:\winnt\s	ystem32\dri	vers\diskpe	erf.sys	Kernel
aic78xx	aic78xx	Not Availa		Kernel Dr		False	Driver	True	Boot	Running	OK	Normal	False
Disabled	Stopped	OK	Normal	False	False	F 1	True		· · ·	. 20/1:			77 1
ami0nt	ami0nt	Not Availa		Kernel Dri False	iver False	False	dmboot Driver	dmboot False		ystem32\dri Stopped		-	Kernel False
Disabled amsint	Stopped amsint	OK Not Availa	Normal ble	Kernel Dr		False	False	raise	Disabled	Stopped	UK	Normal	raise
Disabled	Stopped	OK	Normal	False	False	i aisc	dmio	Logical D	isk Manage	er Driver			
asc	asc	Not Availa		Kernel Dr		False		ystem32\dri	_		Kernel Dri	iver	True
Disabled	Stopped	OK	Normal	False	False		Boot	Running	OK	Normal	False	True	
asc3350p	asc3350p	Not Availa	ıble	Kernel Dr	iver	False	dmload	dmload	c:\winnt\s	ystem32\dri	vers\dmloa	d.sys	Kernel
Disabled	1.1	OK	Normal	False	False		Driver	True	Boot	Running	OK	Normal	False
asc3550	asc3550	Not Availa		Kernel Dr		False	True	Y . 1(D) D	00.41	.			
Disabled	11	OK	Normal	False	False		e100b	` /	RO Adapte		V1 D		Т
	RAS Asyr ystem32\dri			Kernel Dri	ivor	False	Manual	ystem32\dri Running	OK	Normal	Kernel Dri False	True	True
Manual	Stopped	-	Normal	False	False	raise	efs	EFS		ystem32\dri			m Driver
atapi		DE/ESDI H			1 4150		True	Disabled	Running	-	Normal	False	True
-	ystem32\dri			Kernel Dr	iver	True	fastfat	Fastfat	·	ystem32\dri			File System
Boot	Running	OK	Normal	False	True		Driver	True		-	OK	Normal	False
atdisk	Atdisk	Not Availa	able	Kernel Dr	iver	False	True						
Disabled	Stopped	OK	Ignore	False	False		fd16_700	Fd16_700	Not Avail	able	Kernel Dri	iver	False
atirage3	atirage3			ivers\atimpa	-	Kernel	Disabled	11	OK	Normal	False	False	
Driver	True	Manual	Running	OK	Ignore	False	fdc	113	sk Controll		-	•	vers\fdc.sys
True													
4	ATMADI	OCI: A D	. 1				Kernel Dr		True	Manual	Running	OK	Normal
atmarpc		Client Pro		Varnal Dr	Ivor	Enlan	False	True					
c:\winnt\s	ystem32\dri	vers\atmar _l	c.sys	Kernel Dri		False	False fips	True Fips	c:\winnt\s	ystem32\dri	vers\fips.sy	/S	Kernel
c:\winnt\sy Manual	ystem32\dri Stopped	vers\atmarı OK	oc.sys Normal	False	False		False fips Driver	True			vers\fips.sy		
c:\winnt\s	ystem32\dri Stopped Audio Stu	vers\atmarı OK	oc.sys Normal c:\winnt\sy	False ystem32\dri	False ivers\audstu		False fips Driver True	True Fips True	c:\winnt\s Auto	ystem32\dri Running	vers\fips.sy	⁄s Normal	Kernel
c:\winnt\sy Manual audstub	ystem32\dri Stopped Audio Stu	vers\atmarı OK b Driver	oc.sys Normal	False	False ivers\audstu	ıb.sys	False fips Driver	True Fips	c:\winnt\s	ystem32\dri Running	vers\fips.sy OK	⁄s Normal	Kernel False
c:\winnt\sy Manual audstub Kernel Dr	ystem32\dri Stopped Audio Stu iver True	vers\atmarı OK b Driver	oc.sys Normal c:\winnt\sy Manual	False ystem32\dri Running	False ivers\audstu	ıb.sys	False fips Driver True fireport	True Fips True fireport Stopped flashpnt	c:\winnt\s Auto	ystem32\dri Running able Normal	vers\fips.sy OK Kernel Dri	vs Normal iver False	Kernel False
c:\winnt\sy Manual audstub Kernel Dr False b57w2k c:\winnt\sy	ystem32\dri Stopped Audio Stu iver True Broadcom ystem32\dri	vers\atmarp OK b Driver True NetXtreme vers\b57w2	Normal c:\winnt\sy Manual c:Gigabit Et	False ystem32\dri Running thernet Kernel Dri	False ivers\audstu OK	ıb.sys	False fips Driver True fireport Disabled flashpnt Disabled	True Fips True fireport Stopped flashpnt Stopped	c:\winnt\s Auto Not Avail OK Not Avail OK	ystem32\dri Running able Normal able Normal	vers\fips.sy OK Kernel Dri False Kernel Dri False	Normal iver False iver False	Kernel False False
c:\winnt\sy Manual audstub Kernel Dr False b57w2k c:\winnt\sy Manual	ystem32\dri Stopped Audio Stu iver True Broadcom ystem32\dri Running	vers\atmarp OK b Driver True NetXtreme vers\b57w2 OK	Normal c:\winnt\sy Manual c:Gigabit Et ck.sys Normal	False ystem32\dri Running thernet Kernel Dri False	False ivers\audstu OK iver True	ıb.sys Normal True	False fips Driver True fireport Disabled flashpnt Disabled flpydisk	True Fips True fireport Stopped flashpnt Stopped Floppy Di	c:\winnt\s Auto Not Avail OK Not Avail OK sk Driver	ystem32\dri Running able Normal able Normal c:\winnt\sy	vers\fips.sy OK Kernel Dri False Kernel Dri False ystem32\dri	Normal iver False iver False ivers\flpydi	Kernel False False False sk.sys
c:\winnt\sy Manual audstub Kernel Dr False b57w2k c:\winnt\sy Manual beep	ystem32\dri Stopped Audio Stu iver True Broadcom ystem32\dri Running Beep	vers\atmarp OK b Driver True NetXtreme vers\b57w2 OK c:\winnt\sy	oc.sys Normal c:\winnt\sy Manual e: Gigabit Et k.sys Normal vstem32\driv	False ystem32\dri Running thernet Kernel Dri False ivers\beep.s	False ivers\audstu OK iver True ys	ab.sys Normal True Kernel	False fips Driver True fireport Disabled flashpnt Disabled flpydisk Kernel Dr	True Fips True fireport Stopped flashpnt Stopped Floppy Di iver	c:\winnt\s Auto Not Avail OK Not Avail OK	ystem32\dri Running able Normal able Normal	vers\fips.sy OK Kernel Dri False Kernel Dri False	Normal iver False iver False ivers\flpydi	Kernel False False
c:\winnt\s' Manual audstub Kernel Dr False b57w2k c:\winnt\s' Manual beep Driver	ystem32\dri Stopped Audio Stu iver True Broadcom ystem32\dri Running	vers\atmarp OK b Driver True NetXtreme vers\b57w2 OK	Normal c:\winnt\sy Manual c:Gigabit Et ck.sys Normal	False ystem32\dri Running thernet Kernel Dri False ivers\beep.s	False ivers\audstu OK iver True	ıb.sys Normal True	False fips Driver True fireport Disabled flashpnt Disabled flpydisk Kernel Dr False	True Fips True fireport Stopped flashpnt Stopped Floppy Di iver True	c:\winnt\s Auto Not Avail OK Not Avail OK sk Driver True	ystem32\dri Running able Normal able Normal c:\winnt\sy Manual	vers\fips.sy OK Kernel Dri False Kernel Dri False ystem32\dri	Normal iver False iver False ivers\flpydi	Kernel False False False sk.sys
c:\winnt\s' Manual audstub Kernel Dr False b57w2k c:\winnt\s' Manual beep Driver True	ystem32\dri Stopped Audio Stu iver True Broadcom ystem32\dri Running Beep True	vers\atmarp OK b Driver True NetXtreme vers\b57w2 OK c:\winnt\sy System	Normal c:\winnt\sy Manual e:Gigabit Et e: Sys Normal ystem32\dri Running	False ystem32\dri Running thernet Kernel Dri False ivers\beep.s	False ivers\audstu OK iver True ys Normal	ab.sys Normal True Kernel False	False fips Driver True fireport Disabled flashpnt Disabled flpydisk Kernel Dr False ftdisk	True Fips True fireport Stopped flashpnt Stopped Floppy Di iver True Volume M	c:\winnt\s Auto Not Avail OK Not Avail OK sk Driver True	ystem32\dri Running able Normal able Normal c:\winnt\sy Manual	vers\fips.sy OK Kernel Dri False Kernel Dri False ystem32\dri Running	Normal iver False iver False ivers\flpydia	Kernel False False False sk.sys Normal
c:\winnt\sy Manual audstub Kernel Dr False b57w2k c:\winnt\sy Manual beep Driver True buslogic	ystem32\dri Stopped Audio Stu iver True Broadcom ystem32\dri Running Beep True	vers\atmarp OK b Driver True NetXtreme vers\b57w2 OK c:\winnt\sy System Not Availa	Normal c:\winnt\sy Manual e:Gigabit Et ek.sys Normal vstem32\dri Running able	False ystem32\dri Running thernet Kernel Dri False ivers\beep.s OK Kernel Dri	False ivers\audstu OK iver True ys Normal	ab.sys Normal True Kernel	False fips Driver True fireport Disabled flashpnt Disabled flpydisk Kernel Dr False ftdisk c:\winnt\s;	True Fips True fireport Stopped flashpnt Stopped Floppy Di iver True	c:\winnt\s Auto Not Avail OK Not Avail OK sk Driver True Ianager Drivers\ftdisk	ystem32\dri Running able Normal able Normal c:\winnt\sy Manual	vers\fips.sy OK Kernel Dri False Kernel Dri False ystem32\dri Running Kernel Dri	Normal iver False iver False ivers\flpydia	Kernel False False False sk.sys
c:\winnt\s' Manual audstub Kernel Dr False b57w2k c:\winnt\s' Manual beep Driver True	ystem32\dri Stopped Audio Stu iver True Broadcom ystem32\dri Running Beep True BusLogic	vers\atmarp OK b Driver True NetXtreme vers\b57w2 OK c:\winnt\sy System	oc.sys Normal c:\winnt\sy Manual e: Gigabit Et ek.sys Normal vstem32\dri Running able Normal	False ystem32\dri Running thernet Kernel Dri False ivers\beep.s	False ivers\audstu OK iver True ys Normal iver False	ab.sys Normal True Kernel False	False fips Driver True fireport Disabled flashpnt Disabled flpydisk Kernel Dr False ftdisk	True Fips True fireport Stopped flashpnt Stopped Floppy Di iver True Volume M ystem32\dri Running	c:\winnt\s Auto Not Avail OK Not Avail OK sk Driver True Ianager Drivers\ftdisk	ystem32\dri Running able Normal able Normal c:\winnt\sy Manual iver .sys Normal	vers\fips.sy OK Kernel Dri False Kernel Dri False ystem32\dri Running	Normal iver False iver False ivers\flpydic OK	Kernel False False False sk.sys Normal
c:\winnt\sy Manual audstub Kernel Dr False b57w2k c:\winnt\sy Manual beep Driver True buslogic Disabled	ystem32\dri Stopped Audio Stu iver True Broadcom ystem32\dri Running Beep True BusLogic Stopped	vers\atmarp OK b Driver True NetXtreme vers\b57w2 OK c:\winnt\sy System Not Availa OK	oc.sys Normal c:\winnt\sy Manual e: Gigabit Et ek.sys Normal vstem32\dri Running able Normal	False ystem32\dri Running thernet Kernel Dri False ivers\beep.s OK Kernel Dri False	False ivers\audstu OK iver True ys Normal iver False	ab.sys Normal True Kernel False False	False fips Driver True fireport Disabled flashpnt Disabled flpydisk Kernel Dr False ftdisk c:\winnt\s; Boot gpc	True Fips True fireport Stopped flashpnt Stopped Floppy Di iver True Volume M ystem32\dri Running	c:\winnt\s Auto Not Avail OK Not Avail OK sk Driver True Ianager Drivers\ftdisk OK acket Class	ystem32\dri Running able Normal able Normal c:\winnt\sy Manual iver .sys Normal ifier	vers\fips.sy OK Kernel Dri False Kernel Dri False ystem32\dri Running Kernel Dri	Normal iver False iver False ivers\flpydia OK iver	Kernel False False False sk.sys Normal
c:\winnt\sy Manual audstub Kernel Dr False b57w2k c:\winnt\sy Manual beep Driver True buslogic Disabled cd20xrnt Disabled cdaudio	ystem32\dri Stopped Audio Stu iver True Broadcom ystem32\dri Running Beep True BusLogic Stopped cd20xrnt Stopped Cdaudio	vers\atmarp OK b Driver True NetXtreme vers\b57w2 OK c:\winnt\sy System Not Availa OK Not Availa OK	oc.sys Normal c:\winnt\sy Manual c:\winnt\sy Normal c:\winnt\sy Manual	False ystem32\dri Running thernet Kernel Dri False ivers\beep.s OK Kernel Dri False Kernel Dri False ivers\cdaudi	False ivers\audstu OK iver True ys Normal iver False iver False	nb.sys Normal True Kernel False False False Kernel	False fips Driver True fireport Disabled flashpnt Disabled flydisk Kernel Dr False fldisk c:\winnt\s; Boot gpc c:\winnt\s; Manual	True Fips True fireport Stopped flashpnt Stopped Floppy Di iver True Volume M ystem32\dri Running Generic P: ystem32\dri Running	c:\winnt\s Auto Not Avail OK Not Avail OK sk Driver True Ianager Drivers\ftdisk OK acket Class vers\msgpc OK	ystem32\dri Running able Normal able Normal c:\winnt\sy Manual iver .sys Normal ifier c.sys Normal	vers\fips.sy OK Kernel Dri False Kernel Dri Running Kernel Dri False Kernel Dri False Kernel Dri False	Normal iver False iver False ivers\flpydi OK iver True iver	Kernel False False False sk.sys Normal
c:\winnt\sy Manual audstub Kernel Dr False b57w2k c:\winnt\sy Manual beep Driver True buslogic Disabled cd20xrnt Disabled cdaudio Driver	ystem32\dri Stopped Audio Stu iver True Broadcom ystem32\dri Running Beep True BusLogic Stopped cd20xrnt Stopped	vers\atmarp OK b Driver True NetXtreme vers\b57w2 OK c:\winnt\sy System Not Availa OK Not Availa OK	oc.sys Normal c:\winnt\sy Manual	False ystem32\dri Running thernet Kernel Dri False ivers\beep.s OK Kernel Dri False Kernel Dri False ivers\cdaudi	False ivers\audstu OK iver True ys Normal iver False iver False	nb.sys Normal True Kernel False False	False fips Driver True fireport Disabled flashpnt Disabled flpydisk Kernel Dr False ftdisk c:\winnt\s; Boot gpc c:\winnt\s; Manual i8042prt	True Fips True fireport Stopped flashpnt Stopped Floppy Di iver True Volume M ystem32\dri Running Generic P ystem32\dri Running i8042 Key	c:\winnt\s Auto Not Avail OK Not Avail OK sk Driver True Ianager Drivers\ftdisk OK acket Class vers\msgpc OK board and	ystem32\dri Running able Normal able Normal c:\winnt\sy Manual iver .sys Normal ifier c.sys Normal pS/2 Mouse	vers\fips.sy OK Kernel Dri False Kernel Dri Running Kernel Dri False Kernel Dri False Kernel Dri False	Normal iver False iver False ivers\flpydi OK iver True iver True	Kernel False False False sk.sys Normal True
c:\winnt\sy Manual audstub Kernel Dr False b57w2k c:\winnt\sy Manual beep Driver True buslogic Disabled cd20xrnt Disabled cdaudio Driver False	ystem32\dri Stopped Audio Stu iver True Broadcom ystem32\dri Running Beep True BusLogic Stopped cd20xrnt Stopped Cdaudio False	vers\atmarp OK b Driver True NetXtreme vers\b57w2 OK c:\winnt\sy System Not Availa OK Not Availa OK c:\winnt\sy System	nc.sys Normal c:\winnt\sy Manual c:\winnt\sy Manual c:\digabit Etck.sys Normal rystem32\dri Running able Normal able Normal rystem32\dri Stopped	False ystem32\dri Running thernet Kernel Dri False ivers\beep.s OK Kernel Dri False Kernel Dri False ivers\cdaudi OK	False ivers\audstu OK iver True ivs Normal iver False iver False io.sys Ignore	nb.sys Normal True Kernel False False False Kernel False	False fips Driver True fireport Disabled flashpnt Disabled flpydisk Kernel Dr False ftdisk c:\winnt\s; Boot gpc c:\winnt\s; Manual i8042prt c:\winnt\s;	True Fips True fireport Stopped flashpnt Stopped Floppy Di iver True Volume M ystem32\dri Running Generic P ystem32\dri Running i8042 Key ystem32\dri	c:\winnt\s Auto Not Avail OK Not Avail OK sk Driver True Ianager Drivers\ftdisk OK acket Class vers\msgpc OK board and vers\i8042	ystem32\dri Running able Normal able Normal c:\winnt\sy Manual iver .sys Normal ifier c.sys Normal pS/2 Mouse prt.sys	vers\fips.sy OK Kernel Dri False Kernel Dri Running Kernel Dri False Port Drive Kernel Drive	Normal iver False iver False ivers\flpydi OK iver True iver True er iver	Kernel False False False sk.sys Normal
c:\winnt\s' Manual audstub Kernel Dr False b57w2k c:\winnt\s' Manual beep Driver True buslogic Disabled cd20xrnt Disabled cdaudio Driver False cdfs	ystem32\dri Stopped Audio Stu iver True Broadcom ystem32\dri Running Beep True BusLogic Stopped cd20xrnt Stopped Cdaudio False	vers\atmarp OK b Driver True NetXtreme vers\b57w2 OK c:\winnt\sy System Not Availa OK Not Availa OK c:\winnt\sy System c:\winnt\sy	nc.sys Normal c:\winnt\sy Manual c:\winnt\sy Manual c:\winnt\sy Manual c:\winnt\sy Manual c:\winnt\sy Manual c:\winnt\sy Manual c:\winnt\sy Normal	False ystem32\dri Running thernet Kernel Dri False ivers\beep.s OK Kernel Dri False Kernel Dri False ivers\cdaudi OK	False evers\audstu OK iver True eys Normal iver False iver False io.sys Ignore	tb.sys Normal True Kernel False False False Kernel False Kernel File System	False fips Driver True fireport Disabled flashpnt Disabled flpydisk Kernel Dr False ftdisk c:\winnt\s; Boot gpc c:\winnt\s; Manual i8042prt c:\winnt\s; System	True Fips True fireport Stopped flashpnt Stopped Floppy Di iver True Volume M ystem32\dri Running Generic P: ystem32\dri Running i8042 Key ystem32\dri Running	c:\winnt\s Auto Not Avail OK Not Avail OK sk Driver True Ianager Drivers\ftdisk OK acket Class vers\msgpc OK board and vers\i8042 OK	ystem32\dri Running able Normal able Normal c:\winnt\sy Manual iver .sys Normal ifier c.sys Normal PS/2 Mouse prt.sys Normal	vers\fips.sy OK Kernel Dri False Kernel Dri Running Kernel Dri False Kernel Dri False Kernel Dri False Kernel Dri False Kernel Dri False Port Drive Kernel Dri False	Normal iver False iver False ivers\flpydi OK iver True iver True or iver True	Kernel False False False sk.sys Normal True True
c:\winnt\s' Manual audstub Kernel Dr False b57w2k c:\winnt\s' Manual beep Driver True buslogic Disabled cd20xrnt Disabled cdaudio Driver False cdfs Driver	ystem32\dri Stopped Audio Stu iver True Broadcom ystem32\dri Running Beep True BusLogic Stopped cd20xrnt Stopped Cdaudio False	vers\atmarp OK b Driver True NetXtreme vers\b57w2 OK c:\winnt\sy System Not Availa OK Not Availa OK c:\winnt\sy System c:\winnt\sy	nc.sys Normal c:\winnt\sy Manual c:\winnt\sy Manual c:\digabit Etck.sys Normal rystem32\dri Running able Normal able Normal rystem32\dri Stopped	False ystem32\dri Running thernet Kernel Dri False ivers\beep.s OK Kernel Dri False Kernel Dri False ivers\cdaudi OK	False ivers\audstu OK iver True ivs Normal iver False iver False io.sys Ignore	nb.sys Normal True Kernel False False False Kernel False	False fips Driver True fireport Disabled flashpnt Disabled flpydisk Kernel Dr False ftdisk c:\winnt\s; Boot gpc c:\winnt\s; Manual i8042prt c:\winnt\s; System ini910u	True Fips True fireport Stopped flashpnt Stopped Floppy Di iver True Volume M yystem32\dri Running Generic Pi yystem32\dri Running i8042 Key ystem32\dri Running ini910u	c:\winnt\s Auto Not Avail OK Not Avail OK sk Driver True Ianager Drivers\ftdisk OK acket Class vers\msgpc OK board and vers\i8042 OK Not Avail	ystem32\dri Running able Normal able Normal c:\winnt\sy Manual iver .sys Normal ifier c.sys Normal PS/2 Mouse prt.sys Normal able	vers\fips.sy OK Kernel Dri False Kernel Dri Running Kernel Dri False Running	Normal iver False iver False ivers\flpydi OK iver True iver True iver True iver True iver	Kernel False False False sk.sys Normal True
c:\winnt\s' Manual audstub Kernel Dr False b57w2k c:\winnt\s' Manual beep Driver True buslogic Disabled cd20xrnt Disabled cdaudio Driver False cdfs Driver True	ystem32\dri Stopped Audio Stu iver True Broadcom ystem32\dri Running Beep True BusLogic Stopped cd20xrnt Stopped Cdaudio False	vers\atmarp OK b Driver True NetXtreme vers\b57w2 OK c:\winnt\sy System Not Availa OK Not Availa OK c:\winnt\sy System c:\winnt\sy Disabled	Normal c:\winnt\sy Manual c:\winnt\sy Manual c:\winnt\sy Manual c:Gigabit Etck.sys Normal restem32\dri Running able Normal able Normal restem32\dri Stopped restem32\dri Running	False ystem32\dri Running thernet Kernel Dri False ivers\beep.s OK Kernel Dri False Kernel Dri False ivers\cdaudi OK ivers\cdaudi OK	False iver Saudstu OK iver True iver False iver False iver False iver False iver False iver Normal	tb.sys Normal True Kernel False False Kernel False Kernel False	False fips Driver True fireport Disabled flashpnt Disabled flpydisk Kernel Dr False ftdisk c:\winnt\s; Boot gpc c:\winnt\s; Manual i8042prt c:\winnt\s; System ini910u Disabled	True Fips True fireport Stopped flashpnt Stopped Floppy Di iver True Volume M ystem32\dri Running Generic Pi ystem32\dri Running in042 Key ystem32\dri Running ini910u Stopped	c:\winnt\s Auto Not Avail OK Not Avail OK sk Driver True Ianager Drivers\ftdisk OK acket Class vers\msgpc OK board and vers\i8042 OK Not Avail OK	ystem32\dri Running able Normal able Normal c:\winnt\sy Manual iver .sys Normal ifier c.sys Normal PS/2 Mouse prt.sys Normal able Normal	vers\fips.sy OK Kernel Dri False Kernel Dri False ystem32\dri Running Kernel Dri False	Normal iver False iver False ivers\flpydic OK iver True iver True iver True iver False	Kernel False False False sk.sys Normal True True True False
c:\winnt\sy Manual audstub Kernel Dr False b57w2k c:\winnt\sy Manual beep Driver True buslogic Disabled cd20xrnt Disabled cdaudio Driver False cdfs Driver True	ystem32\dri Stopped Audio Stuiver True Broadcom ystem32\dri Running Beep True BusLogic Stopped cd20xrnt Stopped Cdaudio False Cdfs True	vers\atmarp OK b Driver True NetXtreme vers\b57w2 OK c:\winnt\sy System Not Availa OK Not Availa OK c:\winnt\sy System c:\winnt\sy Disabled Driver	nc.sys Normal c:\winnt\sy Manual c:\winnt\sy Manual c:\winnt\sy Manual c:\k.sys Normal restem32\dri Running able Normal restem32\dri Stopped restem32\dri Running c:\winnt\sy	False ystem32\dri Running thernet Kernel Dri False ivers\beep.s OK Kernel Dri False Kernel Dri False kernel Dri False ivers\cdaudi OK ivers\cdaudi OK ivers\cdfs.sy OK	False iver Saudstu OK iver True iver False iver False iver Ignore ys Normal	tb.sys Normal True Kernel False False False Kernel False Kernel False System False	False fips Driver True fireport Disabled flashpnt Disabled flpydisk Kernel Dr False ftdisk c:\winnt\s; Boot gpc c:\winnt\s; Manual i8042prt c:\winnt\s; System ini910u Disabled intelide	True Fips True fireport Stopped flashpnt Stopped Floppy Di iver True Volume M ystem32\dri Running Generic P ystem32\dri Running i8042 Key ystem32\dri Running ini910u Stopped IntelIde	c:\winnt\s Auto Not Avail OK Not Avail OK sk Driver True Ianager Drivers\ftdisk OK acket Class vers\msgpc OK board and vers\i8042 OK Not Avail OK Not Avail	ystem32\dri Running able Normal able Normal c:\winnt\sy Manual iver .sys Normal ifier c.sys Normal pFS/2 Mouse prt.sys Normal able	vers\fips.sy OK Kernel Dri False Kernel Dri Running Kernel Dri False Kernel Dri	Normal iver False iver False ivers\flpydic OK iver True iver True iver True iver False iver	Kernel False False False sk.sys Normal True True
c:\winnt\s' Manual audstub Kernel Dr False b57w2k c:\winnt\s' Manual beep Driver True buslogic Disabled cd20xrnt Disabled cdaudio Driver False cdfs Driver True	ystem32\dri Stopped Audio Stuiver True Broadcom ystem32\dri Running Beep True BusLogic Stopped cd20xrnt Stopped Cdaudio False Cdfs True	vers\atmarp OK b Driver True NetXtreme vers\b57w2 OK c:\winnt\sy System Not Availa OK Not Availa OK c:\winnt\sy System c:\winnt\sy Disabled	Normal c:\winnt\sy Manual c:\winnt\sy Manual c:\winnt\sy Manual c:Gigabit Etck.sys Normal restem32\dri Running able Normal able Normal restem32\dri Stopped restem32\dri Running	False ystem32\dri Running thernet Kernel Dri False ivers\beep.s OK Kernel Dri False Kernel Dri False ivers\cdaudi OK ivers\cdaudi OK	False iver Saudstu OK iver True iver False iver False iver Ignore ys Normal	tb.sys Normal True Kernel False False Kernel False Kernel False	False fips Driver True fireport Disabled flashpnt Disabled flpydisk Kernel Dr False ftdisk c:\winnt\s; Boot gpc c:\winnt\s; Manual i8042prt c:\winnt\s; System ini910u Disabled	True Fips True fireport Stopped flashpnt Stopped Floppy Di iver True Volume M ystem32\dri Running Generic P ystem32\dri Running i8042 Key ystem32\dri Running ini910u Stopped IntelIde Stopped	c:\winnt\s Auto Not Avail OK Not Avail OK sk Driver True Ianager Drivers\ftdisk OK acket Class vers\msgpc OK board and vers\i8042 OK Not Avail OK Not Avail OK	ystem32\dri Running able Normal able Normal c:\winnt\sy Manual iver .sys Normal ifier c.sys Normal PS/2 Mouse prt.sys Normal able Normal	vers\fips.sy OK Kernel Dri False Kernel Dri Running Kernel Dri False	Normal iver False iver False ivers\flpydic OK iver True iver True iver True iver False	Kernel False False False sk.sys Normal True True True False
c:\winnt\sy Manual audstub Kernel Dr False b57w2k c:\winnt\sy Manual beep Driver True buslogic Disabled cd20xrnt Disabled cdaudio Driver False cdfs Driver True cdrom Kernel Dr	ystem32\dri Stopped Audio Stu iver True Broadcom ystem32\dri Running Beep True BusLogic Stopped cd20xrnt Stopped Cdaudio False Cdfs True CD-ROM iver	vers\atmarp OK b Driver True NetXtreme vers\b57w2 OK c:\winnt\sy System Not Availa OK Not Availa OK c:\winnt\sy System c:\winnt\sy Disabled Driver	nc.sys Normal c:\winnt\sy Manual c:\winnt\sy Manual c:\winnt\sy Manual c:\winnt\sy Manual c:\winnt\sy Manual c:\winnt\sy Normal cystem32\dri Stopped cystem32\dri Running c:\winnt\sy System	False ystem32\dri Running thernet Kernel Dri False ivers\beep.s OK Kernel Dri False Kernel Dri False kernel Dri False ivers\cdaudi OK ivers\cdaudi OK ivers\cdfs.sy OK	False ivers\audstu OK iver True ys Normal iver False iver False iver Normal iver Support	tb.sys Normal True Kernel False False False Kernel False Kernel False System False	False fips Driver True fireport Disabled flashpnt Disabled flpydisk Kernel Dr False ftdisk c:\winnt\s; Boot gpc c:\winnt\s; Manual i8042prt c:\winnt\s; System ini910u Disabled intelide Disabled ipfilterdriv	True Fips True fireport Stopped flashpnt Stopped Floppy Di iver True Volume M ystem32\dri Running Generic P ystem32\dri Running i8042 Key ystem32\dri Running ini910u Stopped IntelIde Stopped	c:\winnt\s Auto Not Avail OK Not Avail OK sk Driver True Ianager Drivers\ftdisk OK acket Class vers\msgpc OK board and vers\i8042 OK Not Avail OK Not Avail OK IP Traffic	ystem32\dri Running able Normal able Normal c:\winnt\sy Manual iver .sys Normal ifier c.sys Normal able Normal able Normal able Normal able Normal	vers\fips.sy OK Kernel Dri False Kernel Dri Running Kernel Dri False	Normal iver False iver False ivers\flpydic OK iver True iver True iver True iver False iver False	Kernel False False False sk.sys Normal True True True False
c:\winnt\s; Manual audstub Kernel Dr False b57w2k c:\winnt\s; Manual beep Driver True buslogic Disabled cd20xrnt Disabled cdaudio Driver False cdfs Driver True cdrom Kernel Dr False	ystem32\dri Stopped Audio Stu iver True Broadcom ystem32\dri Running Beep True BusLogic Stopped cd20xrnt Stopped Cdaudio False Cdfs True CD-ROM iver True	vers\atmarp OK b Driver True NetXtreme vers\b57w2 OK c:\winnt\sy System Not Availa OK Not Availa OK c:\winnt\sy System c:\winnt\sy Disabled Driver True	nc.sys Normal c:\winnt\sy Manual c:\winnt\sy Manual c:\winnt\sy Manual c:\winnt\sy Manual c:\winnt\sy Manual c:\winnt\sy Normal cystem32\dri Stopped cystem32\dri Running c:\winnt\sy System	False ystem32\dri Running thernet Kernel Dri False ivers\beep.s OK Kernel Dri False Kernel Dri False Kernel Dri False ivers\cdaudi OK ivers\cdfs.sy OK	False ivers\audstu OK iver True ys Normal iver False iver False iver Normal iver Support	True Kernel False False Kernel False File System False Lsys Normal	False fips Driver True fireport Disabled flashpnt Disabled flpydisk Kernel Dr False ftdisk c:\winnt\s; Boot gpc c:\winnt\s; Manual i8042prt c:\winnt\s; System ini910u Disabled intelide Disabled ipfilterdriv	True Fips True fireport Stopped flashpnt Stopped Floppy Di iver True Volume M ystem32\dri Running Generic P ystem32\dri Running i8042 Key ystem32\dri Running ini910u Stopped IntelIde Stopped yer	c:\winnt\s Auto Not Avail OK Not Avail OK sk Driver True Ianager Drivers\ftdisk OK acket Class vers\msgpc OK board and vers\i8042 OK Not Avail OK Not Avail OK IP Traffic	ystem32\dri Running able Normal able Normal c:\winnt\sy Manual iver .sys Normal ifier c.sys Normal able Normal able Normal able Normal able Normal	vers\fips.sy OK Kernel Dri False Kernel Dri Running Kernel Dri False Kernel Dri False Port Drive Kernel Dri False Evernel Dri False Kernel Dri False Kernel Dri False Kernel Dri False Kernel Dri False	Normal iver False iver False ivers\flpydic OK iver True iver True iver True iver False iver False	Kernel False False False sk.sys Normal True True True False
c:\winnt\s; Manual audstub Kernel Dr False b57w2k c:\winnt\s; Manual beep Driver True buslogic Disabled cd20xrnt Disabled cdaudio Driver False cdfs Driver True cdrom Kernel Dr False changer System cpqarray	ystem32\dri Stopped Audio Stu iver True Broadcom ystem32\dri Running Beep True BusLogic Stopped cd20xrnt Stopped Cdaudio False Cdfs True CD-ROM iver True Changer Stopped Cpqarray	vers\atmarp OK b Driver True NetXtreme vers\b57w2 OK c:\winnt\sy System Not Availa OK Not Availa OK c:\winnt\sy System c:\winnt\sy Disabled Driver True Not Availa OK Not Availa OK Not Availa OK Not Availa OK Not Availa	oc.sys Normal c:\winnt\sy Manual c:\winnt\sy Manual c:\winnt\sy Manual c:\winnt\sy Manual c:\winnt\sy Manual c:\winnt\sy Normal cystem32\dri Stopped cystem32\dri Running c:\winnt\sy System able Ignore able	False ystem32\dri Running thernet Kernel Dri False ivers\beep.s OK Kernel Dri False Kernel Dri False ivers\cdaudi OK ivers\cdfs.sy OK ystem32\dri Running Kernel Dri False ivers\cdfs.sy OK	False ivers\audstu OK iver True ys Normal iver False iver False io.sys Ignore ys Normal ivers\cdrom OK iver False	True Kernel False False Kernel False File System False Lsys Normal	False fips Driver True fireport Disabled flashpnt Disabled flydisk Kernel Dr False ftdisk c:\winnt\s; Boot gpc c:\winnt\s; Manual i8042prt c:\winnt\s; System ini910u Disabled intelide Disabled ipfilterdriv c:\winnt\s; Manual ipinip	True Fips True fireport Stopped flashpnt Stopped flooppy Di iver True Volume M ystem32\dri Running Generic P ystem32\dri Running i8042 Key ystem32\dri Running ini910u Stopped IntelIde Stopped ver ystem32\dri Stopped IP in IP To	c:\winnt\s Auto Not Avail OK Not Avail OK sk Driver True Ianager Drivers\ftdisk OK acket Class vers\msgpc OK board and vers\i8042 OK Not Avail OK Not Avail OK IP Traffic vers\ipfltdi OK	ystem32\dri Running able Normal able Normal c:\winnt\sy Manual iver .sys Normal ifier c.sys Normal able Normal able Normal able Normal able Normal	vers\fips.sy OK Kernel Dri False Kernel Dri False ystem32\dri Running Kernel Dri False Kernel Dri False E Port Drive Kernel Dri False	Normal iver False iver False ivers\flpydic OK iver True iver True iver True iver False iver False iver False	Kernel False False False False sk.sys Normal True True True False False False
c:\winnt\s; Manual audstub Kernel Dr False b57w2k c:\winnt\s; Manual beep Driver True buslogic Disabled cd20xrnt Disabled cdaudio Driver False cdfs Driver True cdrom Kernel Dr False changer System cpqarray Disabled	ystem32\dri Stopped Audio Stu iver True Broadcom ystem32\dri Running Beep True BusLogic Stopped cd20xrnt Stopped Cdaudio False CD-ROM iver True Changer Stopped Cpqarray Stopped	vers\atmarp OK b Driver True NetXtreme vers\b57w2 OK c:\winnt\sy System Not Availa OK Not Availa OK c:\winnt\sy System c:\winnt\sy Disabled Driver True Not Availa OK Not Availa OK Not Availa OK OK	oc.sys Normal c:\winnt\sy Manual c:\winnt\sy Manual c:\winnt\sy Manual c:\winnt\sy Manual c:\winnt\sy Manual c:\winnt\sy Normal cystem32\dri Running dble Normal cystem32\dri Stopped c:\winnt\sy System dble Ignore dble Normal	False ystem32\dri Running thernet Kernel Dri False ivers\beep.s OK Kernel Dri False Kernel Dri False ivers\cdaudi OK ivers\cdfs.sy OK ystem32\dri Running Kernel Dri False Kernel Dri False	False ivers\audstu OK iver True ys Normal iver False iver False io.sys Ignore ys Normal ivers\cdrom OK iver False	th.sys Normal True Kernel False False False Kernel False File System False .sys Normal False False	False fips Driver True fireport Disabled flashpnt Disabled flpydisk Kernel Dr False ftdisk c:\winnt\s; Boot gpc c:\winnt\s; Manual i8042prt c:\winnt\s; System ini910u Disabled intelide Disabled intelide Disabled ipfilterdriv c:\winnt\s; Manual ipinip Kernel Dr	True Fips True fireport Stopped flashpnt Stopped Floppy Di iver True Volume M ystem32\dri Running Generic P ystem32\dri Running i8042 Key ystem32\dri Running ini910u Stopped IntelIde Stopped ver ystem32\dri Stopped IP in IP To iver	c:\winnt\s Auto Not Avail OK Not Avail OK sk Driver True Ianager Drivers\ftdisk OK acket Class vers\msgpc OK board and vers\i8042 OK Not Avail OK Not Avail OK IP Traffic vers\ipfltdi OK	ystem32\dri Running able Normal able Normal c:\winnt\sy Manual iver .sys Normal ifier c.sys Normal able Normal able Normal able Normal able Normal able Normal	vers\fips.sy OK Kernel Dri False Kernel Dri False Running Kernel Dri False Kernel Dri False Port Drive Kernel Dri False vernel Dri False	Normal iver False iver False ivers\flpydic OK iver True iver True iver True iver False iver False iver False iver False iver	Kernel False False False False sk.sys Normal True True True False False False
c:\winnt\s; Manual audstub Kernel Dr False b57w2k c:\winnt\s; Manual beep Driver True buslogic Disabled cd20xrnt Disabled cdaudio Driver False cdfs Driver True cdrom Kernel Dr False changer System cpqarray Disabled cpqarry2	ystem32\dri Stopped Audio Stu iver True Broadcom ystem32\dri Running Beep True BusLogic Stopped cd20xrnt Stopped Cdaudio False Cdfs True CD-ROM iver True Changer Stopped Cpqarray Stopped cpqarry2	vers\atmarp OK b Driver True NetXtreme vers\b57w2 OK c:\winnt\sy System Not Availa OK Not Availa OK c:\winnt\sy System c:\winnt\sy Disabled Driver True Not Availa OK Not Availa OK Not Availa OK Not Availa OK Not Availa	oc.sys Normal c:\winnt\sy Manual c:\winnt\sy Manual c:\winnt\sy Manual c:\winnt\sy Manual c:\winnt\sy Manual c:\winnt\sy Normal cystem32\dri Running dble Normal cystem32\dri Stopped c:\winnt\sy System dble Ignore dble Normal	False ystem32\dri Running thernet Kernel Dri False ivers\beep.s OK Kernel Dri False Kernel Dri False ivers\cdaudi OK ivers\cdfs.sy OK ystem32\dri Running Kernel Dri False ivers\cdfs.sy OK	False ivers\audstu OK iver True ys Normal iver False iver False io.sys Ignore ys Normal ivers\cdrom OK iver False	True Kernel False False Kernel False File System False Lsys Normal False	False fips Driver True fireport Disabled flashpnt Disabled flydisk Kernel Dr False ftdisk c:\winnt\s; Boot gpc c:\winnt\s; Manual i8042prt c:\winnt\s; System ini910u Disabled intelide Disabled ipfilterdriv c:\winnt\s; Manual ipinip	True Fips True fireport Stopped flashpnt Stopped flooppy Di iver True Volume M ystem32\dri Running Generic P ystem32\dri Running i8042 Key ystem32\dri Running ini910u Stopped IntelIde Stopped ver ystem32\dri Stopped IP in IP To	c:\winnt\s Auto Not Avail OK Not Avail OK sk Driver True Ianager Drivers\ftdisk OK acket Class vers\msgpo OK board and vers\i8042 OK Not Avail OK Not Avail OK IP Traffic vers\ipfltdi OK unnel Drive	ystem32\dri Running able Normal able Normal c:\winnt\sy Manual iver .sys Normal ifier c.sys Normal able rv.sys Normal	vers\fips.sy OK Kernel Dri False Kernel Dri False Running Kernel Dri False Kernel Dri False Port Drive Kernel Dri False vernel Dri False	Normal iver False iver False ivers\flpydic OK iver True iver True iver True iver False iver False iver False iver False iver	Kernel False False False sk.sys Normal True True True False False False False

ipnat IP Kernel Driver		Address T False	ranslator Manual	c:\winnt\sy Stopped		vers\ipnat.sys Normal	netbios File System	NetBIOS	Interface True		ystem32\dri		s.sys Normal
False Fal		1 4150		эторреа	011	110111111	False	True	1100	System	11411111119	011	110111111
1	SEC driv				vers\ipsec.s		netbt	NetBios o	1 1		ystem32\dri		-
Kernel Driver		True	Manual	Running	OK	Normal	Kernel Dr		True	System	Running	OK	Normal
False Tru ipsraidn ips		Not Availa	hla	Kernel Dri	ivor	False	False	True NatDetect	c:\winnt\s	vetam32\dr	ivers\netdte	et eve	Kernel
		OK	Normal	False	False	raisc	Driver	False	Manual	Stopped		Normal	False
		ISA Bus D					False			~~~~			
c:\winnt\syste				Kernel Dri	iver	True	npfs	Npfs			ivers\npfs.s		File System
	inning			False	True		Driver	True	System	Running	OK	Normal	False
kbdclass Ke	_	Class Drive		Kernel Dri	ivor	True	True ntfs	Ntfs	c:\xxinnt\cs	vetem32\dr	ivers\ntfs.sy	ie.	File System
-	inning		-	False	True	Truc	Driver	True		Running		Normal	False
-	_			vers\ksecdo		Kernel	True						
Driver Tru	ue	Boot	Running	OK	Normal	False	null	Null		•	ivers\null.sy		Kernel
True	.01	NY / A '1		W 1D:		F 1	Driver	True	System	Running	OK	Normal	False
		Not Availa OK	ible Ignore	Kernel Dri False	iver False	False	True	IDV Troff	ic Filter Dri				
•	1.1	Not Availa	_	Kernel Dri		False			ivers\nwlnk		Kernel Dr	iver	False
		OK	Normal	False	False		Manual	Stopped		Normal	False	False	
mnmdd mn	nmdd	c:\winnt\sy	stem32\dri	vers\mnmd	d.sys	Kernel	nwlnkfwd	IPX Traff	ic Forwarde	er Driver			
Driver Tru	ue	System	Running	OK	Ignore	False	· ·	•	ivers\nwlnk	-	Kernel Dr		False
True							Manual	Stopped		Normal	False	False	
				vers\moden	-	Kernel	parallel				ystem32\dri		
Driver Fal False	lse	Manual	Stopped	OK	Ignore	False	Kernel Dr False	rver True	True	Manual	Running	UK	Normal
mouclass Mo	ouse Cla	ss Driver	c·\winnt\sv	stem32\dri	vers\moucl	ass svs	parport	Parallel po	ort driver	c·\winnt\s	ystem32\dri	vers\narnoi	rt svs
Kernel Driver		True	,	Running		Normal	Kernel Dr	1	True		Running		Ignore
False Tru	ue		,	C			False	True		,	Č		U
mountmgr Mo	ountMgr	c:\winnt\sy	stem32\dri	vers\mount	mgr.sys	Kernel	partmgr	PartMgr	c:\winnt\sy		ivers\partm	gr.sys	Kernel
Driver Tru	ue	Boot	Running	OK	Normal	False	Driver	True	Boot	Running	OK	Normal	False
True	: 12 5	NI-4 A:1-	1.1.	V1 D.::		F-1	True	DX/4	\	4 22\ 4	:\ -		I/1
mraid35x mr Disabled Sto	opped			Kernel Dri False	rver False	False	parvdm Driver	ParVdm True	Auto	ystem32\ar Running	ivers\parvdi	n.sys Ignore	Kernel False
				vers\mrxsm		File System	True	True	Auto	Kullillig	OK	ignore	raise
Driver Tru			Running		Normal	False	pci	PCI Bus I	Driver	c:\winnt\s	ystem32\dri	vers\pci.sv	s Kernel
True		,	Ü				Driver	True	Boot	Running	•	Critical	False
msfs Ms		-		vers\msfs.s	-	File System	True						
Driver Tru	ue	System	Running	OK	Normal	False	pcidump		Not Availa		Kernel Dr		False
True							System	Stopped	OK	Ignore	False	False	77 1
		_	Service Pro	xy Kernel Dri	ivor	False	pciide Driver	PCIIde True	-	•	ivers\pciide	.sys Normal	Kernel False
c:\winnt\system Manual Sto	opped		V.sys Normal	False	False	raise	True	True	Boot	Running	OK	Normai	raise
mspclock Mi					1 disc		pemeia	Pemeia	c:\winnt\sv	vstem32\dr	ivers\pcmci	a.svs	Kernel
c:\winnt\syste		_		Kernel Dri	iver	False	Driver	False	-	Stopped		Normal	False
Manual Sto			Normal		False		False			• •			
mspqm Mi							pdcomp		Not Availa	able	Kernel Dr		False
c:\winnt\syste				Kernel Dri		False	Manual	Stopped		Ignore	False	False	
		OK	Normal	False	False	File System	pdframe	PDFRAM		Not Avail		Kernel Dr	
mup Mu Driver Tru		Boot	Running	vers\mup.sy	ys Normal	False	False pdreli	Manual	Stopped Not Availa	OK able	Ignore Kernel Dr	False	False False
True	uc	Door	Ruining	OK	rvormar	1 alsc	Manual	Stopped		Ignore	False	False	1 alse
	ere710	Not Availa	ble	Kernel Dri	iver	False		PDRFRA		Not Avail		Kernel Dr	iver
Disabled Sto	opped	OK	Normal	False	False		False	Manual	Stopped	OK	Ignore	False	False
ndis NI	DIS Syste	em Driver	c:\winnt\sy	/stem32\dri	vers\ndis.sy	ys	pptpminip			niport (PPT	P)		
Kernel Driver		True	Boot	Running	OK	Normal			ivers\rasppt		Kernel Dr		True
False Tru		MDIG	TADID:				Manual	Running		Normal	False	True	
ndistapi Re			TAPI Driv	er Kernel Dri	ivor	True	ptilink		allel Link E ivers\ptilink		Kernel Dr	ivor	True
,	inning	1	Normal	False	True	True	Manual	Running		Normal	False	True	True
	_		WAN Driv		Truc		gl1080	ql1080	Not Availa		Kernel Dr		False
c:\winnt\syste				Kernel Dri	iver	True	Disabled		OK	Normal	False	False	
Manual Ru	ınning	OK	Normal	False	True		ql10wnt	Ql10wnt	Not Availa		Kernel Dr	iver	False
	OIS Prox	-			vers\ndpro		Disabled	Stopped	OK	Normal	False	False	
Kernel Driver		True	Manual	Running	OK	Normal	q11240	ql1240	Not Availa		Kernel Dr		False
False Tru	ue						Disabled	Stopped	OK	Normal	False	False	

ql2100 ql2100 Disabled Stopped	Not Avail OK	able Normal	Kernel Dr False	river False	False	tdpipe Driver	TDPIPE False		ystem32\dri Stopped		.sys Ignore	Kernel False
rasacd Remote	Access Auto	Connection	n Driver			False						
c:\winnt\system32\d	rivers\rasacd	l.sys	Kernel Dr	river	True	tdspx	TDSPX		ystem32\dri	vers\tdspx.	sys	Kernel
System Running	OK	Normal	False	True		Driver	False	Manual	Stopped	OK	Ignore	False
	iniport (L2T					False						
c:\winnt\system32\d	rivers\rasl2tı	o.sys	Kernel Dr	river	True	tdtcp	TDTCP	c:\winnt\s	ystem32\dri	vers\tdtcp.s	sys	Kernel
Manual Running	OK	Normal	False	True		Driver	False	Manual	Stopped	OK	Ignore	False
raspti Direct Pa	ırallel	c:\winnt\s	ystem32\dri	ivers\raspti.	sys	False						
Kernel Driver	True	Manual	Running	OK	Normal	termdd	Terminal l	Device Driv	/er			
False True						c:\winnt\sy	ystem32\dri	vers\termd	d.sys	Kernel Dr	iver	False
rca Microso	t Streaming	Network Ra	aw Channel	l Access		Disabled	Stopped	OK	Normal	False	False	
c:\winnt\system32\d	rivers\rca.sy	s Kernel Dr	iver	False	Manual	tga	tga	Not Availa	able	Kernel Dr	iver	False
Stopped OK	Normal	False	False			System	Stopped	OK	Ignore	False	False	
rdbss Rdbss	c:\winnt\s	ystem32\dri	ivers\rdbss.	sys	File System	udfs	Udfs	c:\winnt\s	ystem32\dri	vers\udfs.s	ys	File System
Driver True	System	Running		Normal	False	Driver	False	Disabled		OK	Normal	False
True		0				False			rr			
rdpwd RDPWD	c:\winnt\s	vstem32\dri	ivers\rdpwd	l.svs	Kernel	ultra66	ultra66	Not Availa	able	Kernel Dr	iver	False
Driver False	Manual	Stopped		Ignore	False	Disabled	Stopped	OK	Normal	False	False	
False				8		update		Update Di				
	D Audio Pla	vback Filte	er Driver				vstem32\dri			Kernel Dr	iver	True
c:\winnt\system32\d			Kernel Dr	iver	False	Manual	Running	1	Normal	False	True	1140
System Stopped		Normal	False	False	1 4150	vgasave	·		ystem32\dri			Kernel
	Filter Drive				ım eve	Driver	True	System	•	OK	Ignore	False
Kernel Driver	True	Manual	Running		Normal	True	Truc	System	Ruining	OK	ignore	Taisc
False True	Truc	Manuai	Kullillig	OK	Normai		Pamata A	ccess IP AI	D Driver			
	ut duissau	a.\immt\.a	viatama 20\ dini	irrana\aania1	arra	wanarp	vstem32\dri			Kernel Dr		True
1			ystem32\dri		-		,		1 2			True
Kernel Driver	True	System	Running	UK	Ignore	Manual	Running	Not Avails	Normal	False	True	False
False True	\		:\-G		W1	wdica	WDICA			Kernel Dr		raise
sfloppy Sfloppy			ivers\sflopp		Kernel	Manual	Stopped	UK	Ignore	False	False	
Driver False	System	Stopped	OK	Ignore	False							
False	37 . 4 . 9				P 1	Environm	nent Variab	les]				
sglfb sglfb	Not Avail		Kernel Dr		False							
						** * 1.1	* * *	** **				
System Stopped	OK	Normal	False	False		Variable		User Nam			~~~~~	_
simbad Simbad	Not Avail	able	Kernel Dr	river	False	ComSpec	%SystemF	Root%\syste	em32\cmd.e		- <systen< td=""><td></td></systen<>	
simbad Simbad Disabled Stopped	Not Avail	able Normal	Kernel Dr False	iver False		ComSpec Os2LibPat	%SystemF	Root%\syste				/I> <system></system>
simbad Simbad Disabled Stopped sparrow Sparrow	Not Avail: OK Not Avail:	able Normal able	Kernel Dr False Kernel Dr	iver False iver	False False	ComSpec Os2LibPat Path	%SystemF th	Root%\syste %SystemF	em32\cmd.e Root%\syste	m32\os2\d	11;	<system></system>
simbad Simbad Disabled Stopped sparrow Sparrow Disabled Stopped	Not Avail OK Not Avail OK	able Normal able Normal	Kernel Dr False Kernel Dr False	iver False iver False	False	ComSpec Os2LibPat Path %SystemF	%SystemF th Root%\syste	Root%\syste %SystemF em32;%Sys	em32\cmd.e Root%\syste temRoot%;	m32\os2\d %SystemR	ll; oot%\Syste	<system> em32\Wbem;</system>
simbad Simbad Disabled Stopped sparrow Sparrow Disabled Stopped spud Special I	Not Avail OK Not Avail OK Ourpose Utili	able Normal able Normal ty Driver	Kernel Dr False Kernel Dr False c:\winnt\sy	river False river False ystem32\dr		ComSpec Os2LibPat Path %SystemF C:\Program	%SystemF th Root%\syste	Root%\syste %SystemF em32;%Sys	em32\cmd.e Root%\syste	m32\os2\d %SystemR	ll; oot%\Syste	<system></system>
simbad Simbad Disabled Stopped sparrow Sparrow Disabled Stopped	Not Avail OK Not Avail OK	able Normal able Normal	Kernel Dr False Kernel Dr False	river False river False ystem32\dr	False	ComSpec Os2LibPat Path %SystemF C:\Program windir	%SystemF th Root%\syste m Files\Mic %SystemF	Root%\syste %SystemF em32;%Sys crosoft SQL Root%	em32\cmd.e Root%\syste temRoot%; Server\80\' <system< td=""><td>m32\os2\d %SystemR Γools\BINI I></td><td>ll; oot%\Syste</td><td><system> em32\Wbem;</system></td></system<>	m32\os2\d %SystemR Γools\BINI I>	ll; oot%\Syste	<system> em32\Wbem;</system>
simbad Simbad Disabled Stopped sparrow Sparrow Disabled Stopped spud Special I Kernel Driver False True	Not Avails OK Not Avails OK Purpose Utili True	able Normal able Normal ty Driver Manual	Kernel Dr False Kernel Dr False c:\winnt\sy Running	False iver False False ystem32\dr OK	False ivers\spud.sys Normal	ComSpec Os2LibPat Path %SystemF C:\Program windir OS	%SystemF th Root%\syste m Files\Mic %SystemF Windows_	Root%\syste %SystemF em32;%Sys crosoft SQL Root% _NT	em32\cmd.e Root%\syste temRoot%; Server\80\' <system <system< td=""><td>m32\os2\d %SystemR Fools\BINI I> I></td><td>ll; oot%\Syste N</td><td><system> em32\Wbem; <system></system></system></td></system<></system 	m32\os2\d %SystemR Fools\BINI I> I>	ll; oot%\Syste N	<system> em32\Wbem; <system></system></system>
simbad Simbad Disabled Stopped sparrow Sparrow Disabled Stopped spud Special I Kernel Driver False True srv Srv	Not Avails OK Not Avails OK Purpose Utili True c:\winnt\s	able Normal able Normal ty Driver Manual	Kernel Dr False Kernel Dr False c:\winnt\sy Running	False iver False iver False ystem32\dr OK s File Syste	False ivers\spud.sys Normal m Driver	ComSpec Os2LibPat Path %SystemF C:\Program windir OS PROCESS	%SystemF th Root%\system Files\Mic %SystemF Windows_ SOR_ARCF	Root%\system %SystemF em32;%Sys crosoft SQL Root% NT HITECTUR	em32\cmd.e Root%\syste temRoot%; Server\80\' <system <system E</system </system 	m32\os2\d %SystemR Γools\BINI I> x86	ll; oot%\Syste N <system< td=""><td><system> em32\Wbem; <system></system></system></td></system<>	<system> em32\Wbem; <system></system></system>
simbad Simbad Disabled Stopped sparrow Sparrow Disabled Special I Kernel Driver False True srv Srv True Manual	Not Avail: OK Not Avail: OK Purpose Utili True c:\winnt\s; Running	able Normal able Normal ty Driver Manual ystem32\dri OK	Kernel Dr False Kernel Dr False c:\winnt\sy Running ivers\srv.sy Normal	iver False iver False ystem32\dr OK s File Syste False	False ivers\spud.sys Normal m Driver True	ComSpec Os2LibPat Path %SystemF C:\Program windir OS PROCESS PROCESS	%SystemFith Root%\systemFiles\Mic %SystemF Windows SOR_ARCF SOR_LEVE	Root%\system %SystemI em32;%Sys crosoft SQL Root% NT HITECTUR EL	em32\cmd.e Root%\syste temRoot%; Server\80\' <system E 15</system 	m32\os2\d %SystemR Γools\BIN! [> [> x86 <system< td=""><td>ll; oot%\Syste N <system 4></system </td><td><system> em32\Wbem; <system> 4></system></system></td></system<>	ll; oot%\Syste N <system 4></system 	<system> em32\Wbem; <system> 4></system></system>
simbad Simbad Disabled Stopped sparrow Sparrow Disabled Stopped spud Special I Kernel Driver False True srv Srv True Manual swenum Software	Not Avails OK Not Avails OK Purpose Utili True c:\winnt\s	able Normal able Normal ty Driver Manual ystem32\dri OK	Kernel Dr False Kernel Dr False c:\winnt\sy Running ivers\srv.sy Normal	iver False iver False ystem32\dr OK s File Syste False	False ivers\spud.sys Normal m Driver True	ComSpec Os2LibPat Path %SystemF C:\Program windir OS PROCESS PROCESS	%SystemF th Root%\system Files\Mic %SystemF Windows_ SOR_ARCF	Root%\system %SystemI em32;%Sys crosoft SQL Root% NT HITECTUR EL	em32\cmd.e Root%\syste temRoot%; Server\80\' <system E 15</system 	m32\os2\d %SystemR Γools\BIN! [> [> x86 <system< td=""><td>ll; oot%\Syste N <system< td=""><td><system> em32\Wbem; <system> 4></system></system></td></system<></td></system<>	ll; oot%\Syste N <system< td=""><td><system> em32\Wbem; <system> 4></system></system></td></system<>	<system> em32\Wbem; <system> 4></system></system>
simbad Simbad Disabled Stopped sparrow Sparrow Disabled Special I Kernel Driver False True srv Srv True Manual	Not Avail: OK Not Avail: OK Purpose Utili True c:\winnt\s; Running	able Normal able Normal ty Driver Manual ystem32\dri OK	Kernel Dr False Kernel Dr False c:\winnt\sy Running ivers\srv.sy Normal	False river False river False ystem32\dr OK s File Syste False ivers\swent	False ivers\spud.sys Normal m Driver True	ComSpec Os2LibPat Path %SystemF C:\Program windir OS PROCESS PROCESS	%SystemFth Root%\systemFiles\Mic %SystemFWindows_ SOR_ARCFOR_LEVE SOR_IDEN	Root%\system %SystemI em32;%Sys crosoft SQL Root% NT HITECTUR EL	em32\cmd.e Root%\syste temRoot%; Server\80\' <system E 15 x86 Family</system 	m32\os2\d %SystemR Γools\BIN1 [> [> x86 <system 15="" model<="" td="" y=""><td>II; oot%\Syste N <system 4=""> 2 Stepping</system></td><td><system> em32\Wbem; <system> 4></system></system></td></system>	II; oot%\Syste N <system 4=""> 2 Stepping</system>	<system> em32\Wbem; <system> 4></system></system>
simbad Simbad Disabled Stopped sparrow Sparrow Disabled Stopped spud Special I Kernel Driver False True srv Srv True Manual swenum Software Kernel Driver False True	Not Avail: OK Not Avail: OK Purpose Utili True c:\winnt\s; Running Bus Driver True	able Normal able Normal ty Driver Manual ystem32\dri OK c:\winnt\s; Manual	Kernel Dr False Kernel Dr False c:\winnt\sy Running ivers\srv.sy: Normal ystem32\dri Running	iver False iver False ystem32\dr OK s File Syste False ivers\swent OK	False ivers\spud.sys Normal m Driver True um.sys Normal	ComSpec Os2LibPar Path %SystemF C:\Program windir OS PROCESS PROCESS GenuineIn PROCESS	%SystemFth Root%\systemFth Files\Mic %SystemF Windows_ GOR_ARCF GOR_LEVE GOR_IDEN atel GOR_REVI	Root%\system %SystemI cm32;%Sys crosoft SQL Root% NT HITECTUR EL TIFIER <system sion<="" td=""><td>em32\cmd.e Root%\syste temRoot%; Server\80\ <system <system E 15 x86 Family 4> 0207</system </system </td><td>m32\os2\d %SystemR Fools\BINI [> </td><td>II; oot%\Syste N <system 4=""> 2 Stepping 4></system></td><td><system> em32\Wbem; <system> 4></system></system></td></system>	em32\cmd.e Root%\syste temRoot%; Server\80\ <system <system E 15 x86 Family 4> 0207</system </system 	m32\os2\d %SystemR Fools\BINI [>	II; oot%\Syste N <system 4=""> 2 Stepping 4></system>	<system> em32\Wbem; <system> 4></system></system>
simbad Simbad Disabled Stopped sparrow Sparrow Disabled Stopped spud Special I Kernel Driver False True srv Srv True Manual swenum Software Kernel Driver False True symc810 symc810	Not Availe OK Not Availe OK Purpose Utili True c:\winnt\s; Running Bus Driver True Not Availe	able Normal able Normal ty Driver Manual ystem32\dri OK c:\winnt\s; Manual	Kernel Dr False Kernel Dr False c:\winnt\sy Running ivers\srv.sy Normal ystem32\dri Running Kernel Dr	iver False iver False ystem32\dr OK s File Syste False ivers\swent OK	False ivers\spud.sys Normal m Driver True um.sys	ComSpec Os2LibPar Path %SystemF C:\Program windir OS PROCESS PROCESS GenuineIn PROCESS NUMBER	%SystemFth Root%\systemFth Root%\systemFth %SystemFth Windows_ SOR_ARCFT SOR_LEVE SOR_IDEN Ittel SOR_REVI 2_OF_PROOF	Root%\system %SystemI cm32;%Sys crosoft SQL Root% NT HITECTUR EL TIFIER <system sion<="" td=""><td>em32\cmd.e Root%\syste temRoot%; Server\80\ <system <system E 15 x86 Family 4> 0207</system </system </td><td>m32\os2\d %SystemR Γools\BIN1 [> [> x86 <system 15="" model<="" td="" y=""><td>II; oot%\Syste N <system 4=""> 2 Stepping 4></system></td><td><system> em32\Wbem; <system> 4></system></system></td></system></td></system>	em32\cmd.e Root%\syste temRoot%; Server\80\ <system <system E 15 x86 Family 4> 0207</system </system 	m32\os2\d %SystemR Γools\BIN1 [> [> x86 <system 15="" model<="" td="" y=""><td>II; oot%\Syste N <system 4=""> 2 Stepping 4></system></td><td><system> em32\Wbem; <system> 4></system></system></td></system>	II; oot%\Syste N <system 4=""> 2 Stepping 4></system>	<system> em32\Wbem; <system> 4></system></system>
simbad Simbad Disabled Stopped sparrow Sparrow Disabled Stopped spud Special I Kernel Driver False True srv Srv True Manual swenum Software Kernel Driver False True symc810 symc810 Disabled Stopped	Not Availe OK Not Availe OK Purpose Utili True c:\winnt\s; Running Bus Driver True Not Availe OK	able Normal able Normal ty Driver Manual ystem32\dri OK c:\winnt\s; Manual	Kernel Dr False Kernel Dr False c:\winnt\sy Running ivers\srv.sy: Normal ystem32\dri Running	iver False iver False ystem32\dr OK s File Syste False ivers\swent OK	False ivers\spud.sys Normal m Driver True um.sys Normal	ComSpec Os2LibPai Path %SystemF C:\Program windir OS PROCESS PROCESS GenuineIn PROCESS NUMBER PATHEX	%SystemIth Root%\systemIth Root%\systemIth %SystemIth Windows SOR_ARCH SOR_IDEN Ittel SOR_REVI R_OF_PROOF	Root%\system %SystemI cm32;%Sys crosoft SQL Root% NT IITECTUR EL TIFIER <system cessors<="" sion="" td=""><td>em32\cmd.e Root%\syste temRoot%; Server\80\ <system <system E 15 x86 Family 4> 0207 2</system </system </td><td>m32\os2\d %SystemR Fools\BIN1 F F System F System F System F System System System System System</td><td>II; oot%\SysteN <system 4=""> 2 Stepping 4> 4></system></td><td><system> em32\Wbem; <system> 4></system></system></td></system>	em32\cmd.e Root%\syste temRoot%; Server\80\ <system <system E 15 x86 Family 4> 0207 2</system </system 	m32\os2\d %SystemR Fools\BIN1 F F System F System F System F System System System System System	II; oot%\SysteN <system 4=""> 2 Stepping 4> 4></system>	<system> em32\Wbem; <system> 4></system></system>
simbad Simbad Disabled Stopped sparrow Sparrow Disabled Stopped spud Special I Kernel Driver False True srv Srv True Manual swenum Software Kernel Driver False True symc810 symc810	Not Availe OK Not Availe OK Purpose Utili True c:\winnt\s; Running Bus Driver True Not Availe OK	able Normal able Normal ty Driver Manual ystem32\dri OK c:\winnt\s; Manual able Normal	Kernel Dr False Kernel Dr False c:\winnt\sy Running ivers\srv.sy Normal ystem32\dri Running Kernel Dr	river False river False ystem32\dr OK s File Syste False rivers\swent OK river False	False ivers\spud.sys Normal m Driver True um.sys Normal	ComSpec Os2LibPai Path %SystemF C:\Program windir OS PROCESS PROCESS GenuineIn PROCESS NUMBER PATHEX	%SystemIth Root%\systemIth Root%\systemIth %SystemIth Windows SOR_ARCH SOR_IDEN Ittel SOR_REVI R_OF_PROOF	Root%\system %SystemI cm32;%Sys crosoft SQL Root% NT IITECTUR EL TIFIER <system cessors<="" sion="" td=""><td>em32\cmd.e Root%\syste temRoot%; Server\80\ <system <system E 15 x86 Family 4> 0207</system </system </td><td>m32\os2\d %SystemR Fools\BIN1 F F System F System F System F System System System System System</td><td>II; oot%\SysteN <system 4=""> 2 Stepping 4> 4></system></td><td><system> em32\Wbem; <system> 4></system></system></td></system>	em32\cmd.e Root%\syste temRoot%; Server\80\ <system <system E 15 x86 Family 4> 0207</system </system 	m32\os2\d %SystemR Fools\BIN1 F F System F System F System F System System System System System	II; oot%\SysteN <system 4=""> 2 Stepping 4> 4></system>	<system> em32\Wbem; <system> 4></system></system>
simbad Simbad Disabled Stopped sparrow Sparrow Disabled Stopped spud Special I Kernel Driver False True srv Srv True Manual swenum Software Kernel Driver False True symc810 symc810 Disabled Stopped	Not Avail: OK Not Avail: OK Purpose Utili True c:\winnt\s; Running Bus Driver True Not Avail: OK Not Avail: OK	able Normal able Normal ty Driver Manual ystem32\dr: OK c:\winnt\s; Manual able Normal able Normal	Kernel Dr False Kernel Dr False c:\winnt\sy Running ivers\srv.sy: Normal ystem32\dri Running Kernel Dr False Kernel Dr False	river False river False ystem32\dr OK s File Syste False rivers\swent OK river False river False river	False ivers\spud.sys Normal m Driver True um.sys Normal False	ComSpec Os2LibPai Path %SystemF C:\Program windir OS PROCESS PROCESS GenuineIn PROCESS NUMBER PATHEX	%SystemFth Root%\systemFth Root%\systemFiles\Mic %SystemF Windows GOR_ARCF GOR_LEVE GOR_IDEN Ittel GOR_REVI B_OF_PROOF KE;BAT;.C	Root%\system %SystemI cm32;%Sys crosoft SQL Root% NT IITECTUR EL TIFIER <system cessors<="" sion="" td=""><td>em32\cmd.e Root%\syste temRoot%; Server\80\' <system E 15 x86 Family 4> 0207 2</system </td><td>m32\os2\d %SystemR Fools\BIN1 F S S S S S S S S S S S S S S S S S S</td><td>II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4></system></td><td><system> m32\Wbem; <system> 4> 37,</system></system></td></system>	em32\cmd.e Root%\syste temRoot%; Server\80\' <system E 15 x86 Family 4> 0207 2</system 	m32\os2\d %SystemR Fools\BIN1 F S S S S S S S S S S S S S S S S S S	II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4></system>	<system> m32\Wbem; <system> 4> 37,</system></system>
simbad Simbad Disabled Stopped sparrow Sparrow Disabled Stopped spud Special I Kernel Driver False True srv Srv True Manual swenum Software Kernel Driver False True symc810 symc810 Disabled Stopped symc8xx Disabled Stopped symmpi symmpi	Not Avail: OK Not Avail: OK Purpose Utili True c:\winnt\s; Running Bus Driver True Not Avail: OK Not Avail: OK	able Normal able Normal ty Driver Manual ystem32\dr: OK c:\winnt\s; Manual able Normal able Normal	Kernel Dr False Kernel Dr False c:\winnt\sy Running ivers\srv.sy: Normal ystem32\dri Running Kernel Dr False Kernel Dr	river False river False ystem32\dr OK s File Syste False rivers\swent OK river False river False river	False ivers\spud.sys Normal m Driver True um.sys Normal False	ComSpec Os2LibPai Path %SystemF C:\Program windir OS PROCESS PROCESS GenuineIn PROCESS NUMBER PATHEX' .COM;.EX	%SystemFth Root%\systemFth Root%\systemFiles\Mic %SystemFWindows_ SOR_ARCFSOR_LEVE SOR_IDEN Itel SOR_REVI R_OF_PROOF KE;BAT;.C %SystemF	Root%\system %SystemI %SystemI em32;%Sys trosoft SQL Root% _NT HITECTUR EL TIFIER <system cessors="" emd;.vbs;="" root%\tem="" root%\tem<="" sion="" td=""><td>em32\cmd.e Root%\syste Root%\syste temRoot%; .Server\80\' <system E 15 x86 Family 4> 0207 2 .VBE;.JS;.J</system </td><td>m32\os2\d %SystemR Fools\BIN1 F S S S S S S S S S S S S S S S S S S</td><td>II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4></system></td><td><system> m32\Wbem; <system> 4> 37,</system></system></td></system>	em32\cmd.e Root%\syste Root%\syste temRoot%; .Server\80\' <system E 15 x86 Family 4> 0207 2 .VBE;.JS;.J</system 	m32\os2\d %SystemR Fools\BIN1 F S S S S S S S S S S S S S S S S S S	II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4></system>	<system> m32\Wbem; <system> 4> 37,</system></system>
simbad Simbad Disabled Stopped sparrow Sparrow Disabled Stopped spud Special I Kernel Driver False True srv Srv True Manual swenum Software Kernel Driver False True symc810 symc810 Disabled Stopped symc8xx Disabled Stopped	Not Avail: OK Not Avail: OK Purpose Utili True c:\winnt\s; Running Bus Driver True Not Avail: OK Not Avail: OK	able Normal able Normal ty Driver Manual ystem32\dr: OK c:\winnt\s; Manual able Normal able Normal	Kernel Dr False Kernel Dr False c:\winnt\sy Running ivers\srv.sy: Normal ystem32\dri Running Kernel Dr False Kernel Dr False ivers\symm	river False river False ystem32\dr OK s File Syste False rivers\swent OK river False river False river	False ivers\spud.sys Normal m Driver True um.sys Normal False False	ComSpec Os2LibPai Path %SystemF C:\Program windir OS PROCESS PROCESS GenuineIn PROCESS NUMBER PATHEX' .COM;,EX	%SystemFth Root%\systemFth Root%\systemFiles\Mic %SystemFWindows_ SOR_ARCFSOR_LEVE SOR_IDEN Itel SOR_REVI R_OF_PROOF KE;BAT;.C %SystemF	Root%\system %SystemI %SystemI em32;%Sys trosoft SQL Root% _NT HITECTUR EL TIFIER <system cessors="" emd;.vbs;="" root%\tem="" root%\tem<="" sion="" td=""><td>em32\cmd.e Root%\syste temRoot%; Server\80\' <system E 15 x86 Family 4> 0207 2 .VBE;.JS;.J</system </td><td>m32\os2\d %SystemR Fools\BIN1 F S S S S S S S S S S S S S S S S S S</td><td>II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4></system></td><td><system> m32\Wbem; <system> 4> 37,</system></system></td></system>	em32\cmd.e Root%\syste temRoot%; Server\80\' <system E 15 x86 Family 4> 0207 2 .VBE;.JS;.J</system 	m32\os2\d %SystemR Fools\BIN1 F S S S S S S S S S S S S S S S S S S	II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4></system>	<system> m32\Wbem; <system> 4> 37,</system></system>
simbad Simbad Disabled Stopped sparrow Sparrow Disabled Stopped spud Special I Kernel Driver False True srv Srv True Manual swenum Software Kernel Driver False True symc810 symc810 Disabled Stopped symc8xx Disabled Stopped symmpi symmpi	Not Avail: OK Not Avail: OK Purpose Utili True c:\winnt\s; Running Bus Driver True Not Avail: OK Not Avail: OK c:\winnt\s;	able Normal able Normal ty Driver Manual ystem32\dri OK c:\winnt\s; Manual able Normal able Normal ystem32\dri able	Kernel Dr False Kernel Dr False c:\winnt\sy Running ivers\srv.sy: Normal ystem32\dri Running Kernel Dr False Kernel Dr False ivers\symm	river False river False ystem32\dr OK s File Syste False rivers\swent OK river False river False river False river False river False river palse rives	False ivers\spud.sys Normal m Driver True um.sys Normal False False Kernel	ComSpec Os2LibPar Path %SystemF C:\Program windir OS PROCESS PROCESS Genuinelm PROCESS NUMBER PATHEX' .COM;.EX TEMP TMP	%SystemFth Root%\systemFth Root%\systemFiles\Mic %SystemFWindows_ SOR_ARCFSOR_LEVE SOR_IDEN Itel SOR_REVI R_OF_PROOF KE;BAT;.C %SystemF	Root%\system %SystemI %SystemI em32;%Sys trosoft SQL Root% NT HITECTUR EL TIFIER <system cessors="" emd;.vbs;="" roofile%\<="" root%\tem="" sion="" td=""><td>em32\cmd.e Root%\syste Root%\syste temRoot%; .Server\80\' <system E 15 x86 Family 4> 0207 2 .VBE;.JS;.J</system </td><td>m32\os2\d %SystemR Fools\BIN1 F S S S S S S S S S S S S S S S S S S</td><td>II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4></system></td><td><system> m32\Wbem; <system> 4> 37,</system></system></td></system>	em32\cmd.e Root%\syste Root%\syste temRoot%; .Server\80\' <system E 15 x86 Family 4> 0207 2 .VBE;.JS;.J</system 	m32\os2\d %SystemR Fools\BIN1 F S S S S S S S S S S S S S S S S S S	II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4></system>	<system> m32\Wbem; <system> 4> 37,</system></system>
simbad Simbad Disabled Stopped sparrow Sparrow Disabled Stopped spud Special I Kernel Driver False True srv Srv True Manual swenum Software Kernel Driver False True symc810 symc810 Disabled Stopped symc8xx Disabled Stopped symmpi Driver True	Not Avail: OK Not Avail: OK Purpose Utili True c:\winnt\s; Running Bus Driver True Not Avail: OK Not Avail: OK c:\winnt\s;	able Normal able Normal ty Driver Manual ystem32\dri OK c:\winnt\s; Manual able Normal able Normal ystem32\dri Running	Kernel Dr False Kernel Dr False c:\winnt\sy Running ivers\srv.sy: Normal ystem32\dri Running Kernel Dr False Kernel Dr False ivers\symm	False iver False iver False ystem32\dr OK s File Syste False ivers\swent OK iver False iver False iver False iver False iver False iver False ppi.sys Normal	False ivers\spud.sys Normal m Driver True um.sys Normal False False Kernel	ComSpec Os2LibPar Path %SystemF C:\Program windir OS PROCESS PROCESS Genuinelm PROCESS NUMBER PATHEX' .COM;.EX TEMP TMP	%SystemFth Root%\systemFth Root%\systemFiles\Mic %SystemFWindows_ SOR_ARCF SOR_LEVE SOR_IDEN ttel \$COF_PROOT KE;BAT;C %SystemF %SystemF %USERPI 0\Administ	Root%\system %SystemI %SystemI em32;%Sys trosoft SQL Root% _NT HITECTUR EL TIFIER <system cessors="" emd;.vbs;="" rator<="" rofile%\="" root%\tem="" sion="" td=""><td>em32\cmd.e Root%\syste Root%\syste temRoot%; .Server\80\' <system E 15 x86 Family 4> 0207 2 .VBE;.JS;.J</system </td><td>m32\os2\d %SystemR Fools\BINI F S S S S S S S S S S S S S S S S S S</td><td>II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4></system></td><td><system> m32\Wbem; <system> 4> 37,</system></system></td></system>	em32\cmd.e Root%\syste Root%\syste temRoot%; .Server\80\' <system E 15 x86 Family 4> 0207 2 .VBE;.JS;.J</system 	m32\os2\d %SystemR Fools\BINI F S S S S S S S S S S S S S S S S S S	II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4></system>	<system> m32\Wbem; <system> 4> 37,</system></system>
simbad Simbad Disabled Stopped Sparrow Disabled Stopped Special I Kernel Driver False True Srv Srv True Manual Swenum Software Kernel Driver False True symc810 Symc810 Symc8xx Disabled Symmpi Driver True True True True	Not Avail: OK Not Avail: OK Purpose Utili True c:\winnt\s; Running Bus Driver True Not Avail: OK Not Avail: OK c:\winnt\s; Boot	able Normal able Normal ty Driver Manual ystem32\dri OK c:\winnt\s; Manual able Normal able Normal ystem32\dri Running	Kernel Dr False Kernel Dr False c:\winnt\sy Running ivers\srv.sy: Normal ystem32\dri Running Kernel Dr False Kernel Dr False ivers\symm OK	False iver False iver False ystem32\dr OK s File Syste False ivers\swent OK iver False iver False iver False iver False iver False iver False ppi.sys Normal	False ivers\spud.sys Normal m Driver True um.sys Normal False False Kernel False	ComSpec Os2LibPar Path %SystemF C:\Program windir OS PROCESS PROCESS GenuineIn PROCESS NUMBER PATHEX .COM;.EX TEMP TMP TEMP CLIENTI- TMP	%SystemFth Root%\systemFth Root%\systemFiles\Mic %SystemFWindows_ SOR_ARCF SOR_LEVE SOR_IDEN ttel \$COF_PROOT KE;BAT;C %SystemF %SystemF %USERPI 0\Administ	Root%\system %SystemI %SystemI em32;%Sys trosoft SQL Root% INT HITECTUR EL TIFIER <system cessors="" emd;.vbs;="" rator="" rofile%\="" rofile%\<="" root%\tem="" sion="" td=""><td>em32\cmd.e Root%\syste temRoot%; Server\80\' <system E 15 x86 Family 4> 0207 2 .VBE;.JS;.J</system </td><td>m32\os2\d %SystemR Fools\BINI F S S S S S S S S S S S S S S S S S S</td><td>II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4></system></td><td><system> m32\Wbem; <system> 4> 37,</system></system></td></system>	em32\cmd.e Root%\syste temRoot%; Server\80\' <system E 15 x86 Family 4> 0207 2 .VBE;.JS;.J</system 	m32\os2\d %SystemR Fools\BINI F S S S S S S S S S S S S S S S S S S	II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4></system>	<system> m32\Wbem; <system> 4> 37,</system></system>
simbad Simbad Disabled Stopped Sparrow Sparrow Disabled Stopped Special I Kernel Driver False True Srv Srv True Manual Swenum Software Kernel Driver False True Symc810 Symc8xx Disabled Stopped Symmpi Driver True True Sym_hi Disabled Stopped Symphi Sym_hi Disabled Stopped Stopped Symphi Sym_hi Disabled Stopped Stopped Symphi Sym_hi Disabled Stopped Stopped Stopped Stopped Stopped Symphi Sym_hi Disabled Stopped S	Not Availe OK Not Availe OK Purpose Utili True c:\winnt\s; Running Bus Driver True Not Availe OK Not Availe OK c:\winnt\s; Boot Not Availe	able Normal able Normal ty Driver Manual ystem32\dr. OK c:\winnt\s; Manual able Normal able Normal ystem32\dr. Running able Normal	Kernel Dr False Kernel Dr False c:\winnt\sy Running ivers\srv.sy: Normal ystem32\dri Running Kernel Dr False Kernel Dr False ivers\symm OK	river False river False ystem32\dr OK s File Syste False rivers\swent OK river False	False ivers\spud.sys Normal m Driver True um.sys Normal False False Kernel False	ComSpec Os2LibPar Path %SystemF C:\Program windir OS PROCESS PROCESS GenuineIn PROCESS NUMBER PATHEX .COM;.EX TEMP TMP TEMP CLIENTI- TMP	%SystemFth Root%\systemFth Root%\systemFuller\Mickle %SystemFuller\Mickle SOR_ARCFSOR_LEVE SOR_IDEN ALCOF_PROOF KE;BAT;C %SystemFuller\Mickle %SystemFuller\Mickle %USERPI %USERPI %USERPI %USERPI	Root%\system %SystemI %SystemI em32;%Sys trosoft SQL Root% INT HITECTUR EL TIFIER <system cessors="" emd;.vbs;="" rator="" rofile%\="" rofile%\<="" root%\tem="" sion="" td=""><td>em32\cmd.e Root%\syste temRoot%; Server\80\' <system E 15 x86 Family 4> 0207 2 .VBE;.JS;.J</system </td><td>m32\os2\d %SystemR Fools\BINI F S S S S S S S S S S S S S S S S S S</td><td>II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4></system></td><td><system> m32\Wbem; <system> 4> 37,</system></system></td></system>	em32\cmd.e Root%\syste temRoot%; Server\80\' <system E 15 x86 Family 4> 0207 2 .VBE;.JS;.J</system 	m32\os2\d %SystemR Fools\BINI F S S S S S S S S S S S S S S S S S S	II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4></system>	<system> m32\Wbem; <system> 4> 37,</system></system>
simbad Simbad Disabled Stopped Sparrow Sparrow Disabled Stopped Special I Kernel Driver False True Srv Srv True Manual Swenum Software Kernel Driver False True symc810 Symc810 Symc8xx Disabled Symc8xx Disabled Symmpi Driver True True Sym_hi Disabled Stopped Symphi Sym_hi Disabled Stopped Symphi Sym_hi Disabled Stopped Symphi Sym_hi Disabled Stopped Stopped Symphi Sym_hi Disabled Stopped Stopped Symphi Sym_hi Disabled Stopped	Not Availe OK Not Availe OK Purpose Utili True c:\winnt\s; Running Bus Driver True Not Availe OK Not Availe OK Silvinint\s; Boot Not Availe OK	able Normal able Normal ty Driver Manual ystem32\dr. OK c:\winnt\s; Manual able Normal able Normal ystem32\dr. Running able Normal	Kernel Dr False Kernel Dr False c:\winnt\sy Running ivers\srv.sy: Normal ystem32\dri Running Kernel Dr False Kernel Dr False ivers\symm OK	river False river False ystem32\dr OK sFile Syste False rivers\swent OK river False system32\dr	False ivers\spud.sys Normal m Driver True um.sys Normal False False Kernel False False	ComSpec Os2LibPar Path %SystemF C:\Program windir OS PROCESS PROCESS GenuineIn PROCESS NUMBER PATHEX .COM;.EX TEMP TMP TEMP CLIENTI- TMP	%SystemFth Root%\systemFth Root%\systemFuller\Mickle %SystemFuller\Mickle SOR_ARCFSOR_LEVE SOR_IDEN ALCOF_PROOF KE;BAT;C %SystemFuller\Mickle %SystemFuller\Mickle %USERPI %USERPI %USERPI %USERPI	Root%\system %SystemI %SystemI em32;%Sys trosoft SQL Root% INT HITECTUR EL TIFIER <system cessors="" emd;.vbs;="" rator="" rofile%\="" rofile%\<="" root%\tem="" sion="" td=""><td>em32\cmd.e Root%\syste temRoot%; Server\80\' <system E 15 x86 Family 4> 0207 2 .VBE;.JS;.J</system </td><td>m32\os2\d %SystemR Fools\BINI F S S S S S S S S S S S S S S S S S S</td><td>II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4></system></td><td><system> m32\Wbem; <system> 4> 37,</system></system></td></system>	em32\cmd.e Root%\syste temRoot%; Server\80\' <system E 15 x86 Family 4> 0207 2 .VBE;.JS;.J</system 	m32\os2\d %SystemR Fools\BINI F S S S S S S S S S S S S S S S S S S	II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4></system>	<system> m32\Wbem; <system> 4> 37,</system></system>
simbad Simbad Disabled Stopped Sparrow Sparrow Disabled Stopped Spud Special I Kernel Driver False True Manual Swenum Software Kernel Driver False True symc810 Disabled Symc8xx Disabled Symmpi Driver True Sym_hi Disabled tcpip Stopped Stopped Sympi Symmpi Symmp	Not Availe OK Not Availe OK Purpose Utili True c:\winnt\s; Running Bus Driver True Not Availe OK Not Availe OK c:\winnt\s; Boot Not Availe OK Protocol Driver	able Normal able Normal ty Driver Manual ystem32\dr. OK c:\winnt\s; Manual able Normal able Normal ystem32\dr. Running able Normal	Kernel Dr False Kernel Dr False c:\winnt\sy Running ivers\srv.sy: Normal ystem32\dri Running Kernel Dr False kernel Dr False ivers\symm OK Kernel Dr False c:\winnt\sy	river False river False ystem32\dr OK sFile Syste False rivers\swent OK river False system32\dr	False ivers\spud.sys Normal m Driver True um.sys Normal False False Kernel False False False ivers\tcpip.sys	ComSpec Os2LibPat Path %SystemF C:\Program windir OS PROCESS PROCESS PROCESS GenuineIn PROCESS NUMBER PATHEX' .COM;.EX TEMP TMP TEMP CLIENTI TMP CLIENTI	%SystemFth Root%\systemFth Root%\systemFuller\Mickle %SystemFuller\Mickle SOR_ARCFSOR_LEVE SOR_IDEN ALCOF_PROOF KE;BAT;C %SystemFuller\Mickle %SystemFuller\Mickle %USERPI %USERPI %USERPI %USERPI	Root%\system %SystemI %SystemI em32;%Sys trosoft SQL Root% INT HITECTUR EL TIFIER <system cessors="" emd;.vbs;="" rator="" rofile%\="" rofile%\<="" root%\tem="" sion="" td=""><td>em32\cmd.e Root%\syste temRoot%; Server\80\' <system E 15 x86 Family 4> 0207 2 .VBE;.JS;.J</system </td><td>m32\os2\d %SystemR Fools\BINI F S S S S S S S S S S S S S S S S S S</td><td>II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4></system></td><td><system> m32\Wbem; <system> 4> 37,</system></system></td></system>	em32\cmd.e Root%\syste temRoot%; Server\80\' <system E 15 x86 Family 4> 0207 2 .VBE;.JS;.J</system 	m32\os2\d %SystemR Fools\BINI F S S S S S S S S S S S S S S S S S S	II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4></system>	<system> m32\Wbem; <system> 4> 37,</system></system>
simbad Simbad Disabled Stopped Sparrow Disabled Stopped Spud Special I Kernel Driver False True Srv Srv True Manual Swenum Software Kernel Driver False True Symc810 Symc8xx Disabled Symc8xx Disabled Symmpi Driver True True True Sym_hi Sym_hi Disabled Stopped Symmpi Driver True True True Sym_hi Sym_hi Disabled Stopped Sympi Sym_hi Disabled Stopped Sympi Sym_hi Sym_hi Disabled Stopped Sympi Sym_hi Disabled Stopped Sympi Sym_hi Disabled Stopped Sympi Driver True True True True True True True Tr	Not Availe OK Not Availe OK Purpose Utili True c:\winnt\s; Running Bus Driver True Not Availe OK Not Availe OK c:\winnt\s; Boot Not Availe OK Protocol Driv True	able Normal able Normal ty Driver Manual ystem32\dri OK c:\winnt\s; Manual able Normal able Normal ystem32\dri Running able Normal erer System	Kernel Dr False Kernel Dr False c:\winnt\sy Running ivers\srv.sy Normal ystem32\dri Running Kernel Dr False Kernel Dr False ivers\symm OK Kernel Dr False c:\winnt\sy Running	river False river False system32\dr OK sFile Syste False rivers\swent OK river False river	False ivers\spud.sys Normal m Driver True um.sys Normal False False Kernel False False False ivers\tcpip.sys Normal	ComSpec Os2LibPat Path %SystemF C:\Program windir OS PROCESS PROCESS GenuineIn PROCESS NUMBER PATHEX' .COM;.EX TEMP TMP TEMP CLIENTI TMP CLIENTI [Jobs]	%SystemIth Root%\systemIth Root%\systemIth %SystemIf Windows_ SOR_ARCI SOR_LEVE SOR_IDEN Ittel SOR_REVI R_OF_PROOF KE;.BAT;.C %SystemIf %SystemIf %USERPI %USERPI 0\Administ	Root%\system %SystemI %SystemI em32;%Sys crosoft SQL Root% NT TIFIER <system cessors="" emd;.vbs;="" rator="" rator<="" roofile%\="" root%\tem="" sion="" td=""><td>em32\cmd.e Root%\syste temRoot%; Server\80\' <system <system E 15 x86 Family 4> 0207 2 .VBE;.JS;.J IP IP Local Settir</system </system </td><td>m32\os2\d %SystemR Fools\BINI F X86 SYSTEM Y 15 Model SYSTEM /td><td>II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4></system></td><td><system> m32\Wbem; <system> 4> 37,</system></system></td></system>	em32\cmd.e Root%\syste temRoot%; Server\80\' <system <system E 15 x86 Family 4> 0207 2 .VBE;.JS;.J IP IP Local Settir</system </system 	m32\os2\d %SystemR Fools\BINI F X86 SYSTEM Y 15 Model SYSTEM	II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4></system>	<system> m32\Wbem; <system> 4> 37,</system></system>
simbad Simbad Disabled Stopped Sparrow Sparrow Disabled Stopped Spud Special I Kernel Driver False True Manual Swenum Software Kernel Driver False True Symc810 Symc8xx Disabled Symc8xx Disabled Symmpi Driver True Sym_hi Disabled Stopped Symphi Symmpi Driver True Sym_hi Sym_hi Disabled Stopped SymcBt Symphi Sym_hi Disabled Stopped SymcBt Sym_hi Sym_hi Disabled Stopped Symphi Sym_hi Disabled Stopped TCP/IP I Kernel Driver	Not Availe OK Not Availe OK Purpose Utili True c:\winnt\s; Running Bus Driver True Not Availe OK Not Availe OK c:\winnt\s; Boot Not Availe OK Protocol Driv True	able Normal able Normal ty Driver Manual ystem32\dri OK c:\winnt\s; Manual able Normal able Normal ystem32\dri Running able Normal cer System c:\winnt\s;	Kernel Dr False Kernel Dr False c:\winnt\sy Running ivers\srv.sy Normal ystem32\dri Running Kernel Dr False Kernel Dr False ivers\symm OK Kernel Dr False c:\winnt\sy Running	river False river False ystem32\dr OK s File Syste False rivers\swent OK river False river False river False river False river False system32\dr OK river False river respirates river	False ivers\spud.sys Normal m Driver True um.sys Normal False False Kernel False False ivers\tcpip.sys Normal nc.sys	ComSpec Os2LibPat Path %SystemF C:\Program windir OS PROCESS PROCESS GenuineIn PROCESS NUMBER PATHEX' .COM;.EX TEMP TMP TEMP CLIENTI TMP CLIENTI [Jobs]	%SystemIth Root%\systemIth Root%\systemIth %SystemIf Windows_ SOR_ARCI SOR_LEVE SOR_IDEN Ittel SOR_REVI R_OF_PROOF KE;.BAT;.C %SystemIf %SystemIf %USERPI %USERPI 0\Administ	Root%\system %SystemI %SystemI em32;%Sys crosoft SQL Root% NT TIFIER <system cessors="" emd;.vbs;="" rator="" rator<="" roofile%\="" root%\tem="" sion="" td=""><td>em32\cmd.e Root%\syste temRoot%; Server\80\' <system E 15 x86 Family 4> 0207 2 .VBE;.JS;.J</system </td><td>m32\os2\d %SystemR Fools\BINI F X86 SYSTEM Y 15 Model SYSTEM /td><td>II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4></system></td><td><system> m32\Wbem; <system> 4> 37,</system></system></td></system>	em32\cmd.e Root%\syste temRoot%; Server\80\' <system E 15 x86 Family 4> 0207 2 .VBE;.JS;.J</system 	m32\os2\d %SystemR Fools\BINI F X86 SYSTEM Y 15 Model SYSTEM	II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4></system>	<system> m32\Wbem; <system> 4> 37,</system></system>
simbad Simbad Disabled Stopped Sparrow Disabled Stopped Spud Special I Kernel Driver False True Srv Srv True Manual Swenum Software Kernel Driver False True symc810 Symc8xx Disabled Symc8xx Disabled Symmpi Driver True True True Sym_hi Sym_hi Disabled Stopped Symmpi Driver True True True Sym_hi Sym_hi Disabled Stopped Symmpi Driver True True True Sym_hi Sym_hi Disabled Stopped Stopped Symmpi Driver True True True Sym_hi Sym_hi Disabled Stopped Stopped Stopped TCP/IP I Kernel Driver False True tdasync TDASY: Kernel Driver	Not Availe OK Not Availe OK Purpose Utili True c:\winnt\s; Running Bus Driver True Not Availe OK Not Availe OK c:\winnt\s; Boot Not Availe OK Protocol Driv True	able Normal able Normal ty Driver Manual ystem32\dri OK c:\winnt\s; Manual able Normal able Normal ystem32\dri Running able Normal erer System	Kernel Dr False Kernel Dr False c:\winnt\sy Running ivers\srv.sy Normal ystem32\dri Running Kernel Dr False Kernel Dr False ivers\symm OK Kernel Dr False c:\winnt\sy Running	river False river False ystem32\dr OK s File Syste False rivers\swent OK river False river False river False river False river False system32\dr OK river False river respirates river	False ivers\spud.sys Normal m Driver True um.sys Normal False False Kernel False False False ivers\tcpip.sys Normal	ComSpec Os2LibPat Path %SystemF C:\Program windir OS PROCESS PROCESS GenuineIn PROCESS NUMBER PATHEX' .COM;.EX TEMP TMP TEMP CLIENTI- TMP CLIENTI- [Jobs] [Followin	%SystemIth Root%\systemIth Root%\systemIth %SystemIf Windows_ SOR_ARCI SOR_LEVE SOR_IDEN Ittel SOR_REVI R_OF_PROOF KE;.BAT;.C %SystemIf %SystemIf %USERPI %USERPI 0\Administ	Root%\system %SystemI %SystemI em32;%Sys crosoft SQL Root% NT TIFIER <system cessors="" emd;.vbs;="" rator="" rator<="" roofile%\="" root%\tem="" sion="" td=""><td>em32\cmd.e Root%\syste temRoot%; Server\80\' <system <system E 15 x86 Family 4> 0207 2 .VBE;.JS;.J IP IP Local Settir</system </system </td><td>m32\os2\d %SystemR Fools\BINI F X86 SYSTEM Y 15 Model SYSTEM /td><td>II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4></system></td><td><system> m32\Wbem; <system> 4> 37,</system></system></td></system>	em32\cmd.e Root%\syste temRoot%; Server\80\' <system <system E 15 x86 Family 4> 0207 2 .VBE;.JS;.J IP IP Local Settir</system </system 	m32\os2\d %SystemR Fools\BINI F X86 SYSTEM Y 15 Model SYSTEM	II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4></system>	<system> m32\Wbem; <system> 4> 37,</system></system>
simbad Simbad Disabled Stopped Sparrow Disabled Stopped Special I Kernel Driver False True Srv Srv True Manual Swenum Software Kernel Driver False True Symc810 Symc8xx Disabled Symc8xx Disabled Symmpi Driver True True True Sym_hi Sym_hi Disabled Stopped Sympi Sympi Driver True True True True Sym_hi Sym_hi Disabled Stopped Sympi Sympi Driver True True True Sym_hi Sym_hi Disabled Stopped Stopped Stopped Stopped Stopped Stopped Stopped Stopped TCP/IP I Kernel Driver False True tdasync TDASY Kernel Driver False False	Not Avail: OK Not Avail: OK Purpose Utili True c:\winnt\s; Running Bus Driver True Not Avail: OK Not Avail: OK c:\winnt\s; Boot Not Avail: OK Protocol Driv True NC False	able Normal able Normal ty Driver Manual ystem32\dri OK c:\winnt\s; Manual able Normal able Normal ystem32\dri Running able Normal ystem32\dri Cer System c:\winnt\s; Manual	Kernel Dr False Kernel Dr False c:\winnt\sy Running ivers\srv.sy: Normal ystem32\dri Running Kernel Dr False Kernel Dr False ivers\symm OK Kernel Dr False c:\winnt\sy Running	river False river False ystem32\dr OK s File Syste False rivers\swent OK river False river OK river False river river False river river False river ri	False ivers\spud.sys Normal m Driver True um.sys Normal False False Kernel False False ivers\tcpip.sys Normal nc.sys	ComSpec Os2LibPat Path %SystemF C:\Program windir OS PROCESS PROCESS GenuineIn PROCESS NUMBER PATHEX' .COM;.EX TEMP TMP TEMP CLIENTI TMP CLIENTI [Jobs]	%SystemIth Root%\systemIth Root%\systemIth %SystemIf Windows_ SOR_ARCI SOR_LEVE SOR_IDEN Ittel SOR_REVI R_OF_PROOF KE;.BAT;.C %SystemIf %SystemIf %USERPI %USERPI 0\Administ	Root%\system %SystemI %SystemI em32;%Sys crosoft SQL Root% NT TIFIER <system cessors="" emd;.vbs;="" rator="" rator<="" roofile%\="" root%\tem="" sion="" td=""><td>em32\cmd.e Root%\syste temRoot%; Server\80\' <system <system E 15 x86 Family 4> 0207 2 .VBE;.JS;.J IP IP Local Settir</system </system </td><td>m32\os2\d %SystemR Fools\BINI F X86 SYSTEM Y 15 Model SYSTEM /td><td>II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4></system></td><td><system> m32\Wbem; <system> 4> 37,</system></system></td></system>	em32\cmd.e Root%\syste temRoot%; Server\80\' <system <system E 15 x86 Family 4> 0207 2 .VBE;.JS;.J IP IP Local Settir</system </system 	m32\os2\d %SystemR Fools\BINI F X86 SYSTEM Y 15 Model SYSTEM	II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4></system>	<system> m32\Wbem; <system> 4> 37,</system></system>
simbad Simbad Disabled Stopped Sparrow Sparrow Disabled Stopped Special I Kernel Driver False True Symc810 Symc810 Symc810 Disabled Symc8xx Disabled Symmpi Driver True Sym_hi Disabled Stopped Symmpi Symmpi Driver True True Sym_hi Sym_hi Disabled Stopped Symphi Symmpi Driver True True Sym_hi Sym_hi Disabled Stopped Symphi Symmpi Driver True True Sym_hi Sym_hi Disabled Stopped Symmpi TCP/IP I Kernel Driver False True tdasync TDASY Kernel Driver False False tdipx TDIPX	Not Avail: OK Not Avail: OK Purpose Utili True c:\winnt\s; Running Bus Driver True Not Avail: OK Not Avail: OK c:\winnt\s; Boot Not Avail: OK Protocol Driv True NC False c:\winnt\s;	able Normal able Normal ty Driver Manual ystem32\dri OK c:\winnt\s; Manual able Normal able Normal ystem32\dri Running able Normal ystem32\dri Cer System c:\winnt\s; Manual	Kernel Dr False Kernel Dr False c:\winnt\sy Running ivers\srv.sy: Normal ystem32\dri Running Kernel Dr False Kernel Dr False ivers\symm OK Kernel Dr False c:\winnt\sy Running ystem32\dri Stopped	iver False iver False ystem32\dr OK s File Syste False ivers\swent OK iver False iver False iver False iver False iver False iver OK iver Cok iver False iver False iver False iver False iver False ystem32\dr OK ivers\tdasyr OK	False ivers\spud.sys Normal m Driver True um.sys Normal False False Kernel False False ivers\tcpip.sys Normal nc.sys Ignore Kernel	ComSpec Os2LibPat Path %SystemF C:\Program windir OS PROCESS PROCESS PROCESS OGenuineIn PROCESS NUMBER PATHEX' .COM;,EX TEMP TMP TEMP CLIENTI TMP CLIENTI [Jobs] [Followin [Print]	%SystemFth Root%\system Files\Mic %SystemF Windows SOR_ARCF SOR_LEVE SOR_IDEN Itel SOR_REVI COF_PROOT KE;.BAT;.C %SystemF %SystemF %USERPI 0\Administ %USERPI 0\Administ	Root%\system %SystemI %SystemI em32;%Sys crosoft SQL Root% NT TIFIER <system cessors="" emd;.vbs;="" rator="" rator<="" roofile%\="" root%\tem="" sion="" td=""><td>em32\cmd.e Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Server\80\' <\SYSTEM E 15 x86 Family 4> 0207 2 .VBE;.JS;.J AP AP Local Settir Local Settir</td><td>m32\os2\d %SystemR Fools\BINI f> I> x86 <system 15="" <system="" ags\temp<="" model="" system="" systemp="" td="" y=""><td>II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4> 4> 4></system></td><td><system> em32\Wbem; <system> d> s7, <system></system></system></system></td></system></td></system>	em32\cmd.e Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Server\80\' <\SYSTEM E 15 x86 Family 4> 0207 2 .VBE;.JS;.J AP AP Local Settir Local Settir	m32\os2\d %SystemR Fools\BINI f> I> x86 <system 15="" <system="" ags\temp<="" model="" system="" systemp="" td="" y=""><td>II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4> 4> 4></system></td><td><system> em32\Wbem; <system> d> s7, <system></system></system></system></td></system>	II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4> 4> 4></system>	<system> em32\Wbem; <system> d> s7, <system></system></system></system>
simbad Simbad Disabled Stopped Sparrow Sparrow Disabled Stopped Special I Kernel Driver False True Symc810 Symc810 Disabled Symc8xx Disabled Symmpi Symmpi Driver True Sym_hi Sym_hi Disabled Stopped Symphi Symmpi Driver True Sym_hi Sym_hi Disabled Stopped Symphi Symmpi Driver True Sym_hi Sym_hi Disabled Stopped Symphi Symphi Symphi Symphi Symphi Symphi Symphi Symphi Driver True Sym_hi Symphi Driver True Sym_hi Symphi Driver True True Sym_hi Symphi	Not Avail: OK Not Avail: OK Purpose Utili True c:\winnt\s; Running Bus Driver True Not Avail: OK Not Avail: OK c:\winnt\s; Boot Not Avail: OK Protocol Driv True NC False	able Normal able Normal ty Driver Manual ystem32\dri OK c:\winnt\s; Manual able Normal able Normal ystem32\dri Running able Normal ystem32\dri Cer System c:\winnt\s; Manual	Kernel Dr False Kernel Dr False c:\winnt\sy Running ivers\srv.sy: Normal ystem32\dri Running Kernel Dr False Kernel Dr False ivers\symm OK Kernel Dr False c:\winnt\sy Running ystem32\dri Stopped	river False river False ystem32\dr OK s File Syste False rivers\swent OK river False river OK river False river river False river river False river ri	False ivers\spud.sys Normal m Driver True um.sys Normal False False Kernel False False ivers\tcpip.sys Normal nc.sys Ignore	ComSpec Os2LibPar Path %SystemF C:\Prograr windir OS PROCESS P	%SystemIth Root%\system Files\Mic %SystemI Windows SOR_ARCF SOR_LEVE SOR_IDEN Itel SOR_REVI LOF_PROOT KE;.BAT;.C %SystemI %SystemI %USERPI 0\Administ %USERPI 0\Administ	Root%\system %SystemI %SystemI em32;%Sys erosoft SQL Root% NT HITECTUR EL TIFIER <system ategories="" cessors="" emd;.vbs;="" of="" owner<="" rator="" rofile%\="" root%\tem="" sion="" td=""><td>em32\cmd.e Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste SysTeM SysTeM E 15 x86 Family 40 2007 2 .VBE;.JS;.J MP MP Local Settin Local Settin Rotify</td><td>m32\os2\d %SystemR Fools\BINI Foo</td><td>II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4> 4> Time Subi</system></td><td><system> em32\Wbem; <system> d> 47, <system></system></system></system></td></system>	em32\cmd.e Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste SysTeM SysTeM E 15 x86 Family 40 2007 2 .VBE;.JS;.J MP MP Local Settin Local Settin Rotify	m32\os2\d %SystemR Fools\BINI Foo	II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4> 4> Time Subi</system>	<system> em32\Wbem; <system> d> 47, <system></system></system></system>
simbad Simbad Disabled Stopped Sparrow Disabled Stopped Special I Kernel Driver False True Symc810 Stopped Symc8xx Disabled Symc8xx Disabled Symmpi Driver True Sym_hi Disabled Stopped Symmpi Driver True True Sym_hi Sym_hi Disabled Stopped Symphi Symmpi Driver True True Sym_hi Stopped S	Not Avail: OK Not Avail: OK Purpose Utili True c:\winnt\s; Running Bus Driver True Not Avail: OK Not Avail: OK C:\winnt\s; Boot Not Avail: OK Protocol Driv True NC False c:\winnt\s; Manual	able Normal able Normal ty Driver Manual ystem32\dri OK c:\winnt\s; Manual able Normal able Normal ystem32\dri Running able Normal ystem32\dri Running able Normal ystem32\dri Running able Normal ystem32\dri Running able Normal ystem32\dri Stystem	Kernel Dr False Kernel Dr False c:\winnt\sy Running ivers\srv.sy: Normal ystem32\dri Running Kernel Dr False Kernel Dr False ivers\symm OK Kernel Dr False c:\winnt\sy Running ystem32\dri Stopped ivers\tdipx.s	river False river False ystem32\dr OK sFile Syste False rivers\swent OK river False river False river False river False river False pi.sys Normal river False ystem32\dr OK rivers\tdasyr OK rivers\tdasyr OK rivers\tdasyr OK sys Ignore	False ivers\spud.sys Normal m Driver True um.sys Normal False False Kernel False False ivers\tcpip.sys Normal nc.sys Ignore Kernel False	ComSpec Os2LibPat Path %SystemF C:\Prograt windir OS PROCESS PROCESS PROCESS PROCESS PROCESS PROCESS PROCESS OF TEMP TEMP TEMP TEMP CLIENTI TMP CLIENTI [Jobs] [Followin [Print] Document Start Time	%SystemIth Root%\system Files\Mic %SystemI Windows SOR_ARCF SOR_LEVE SOR_IDEN Itel SOR_REVI LOF_PROOT KE;.BAT;.C %SystemI %SystemI %USERPI 0\Administ %USERPI 0\Administ g are sub-c	Root%\system %SystemI %SystemI em32;%Sys erosoft SQL Root% NT HITECTUR EL TIFIER <system ategories="" cessors="" e="" elapsed="" emd;.vbs;="" of="" owner="" rator="" rofile%\="" root%\tem="" sion="" t<="" td=""><td>em32\cmd.e Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Server\80\' <\syste SYSTEM E 15 x86 Family 40 0207 2 .VBE;.JS;.J MP MP Local Settin Local Settin Notify ime</td><td>m32\os2\d %SystemR Fools\BINI Foo</td><td>II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4> 4> Time Subrited</system></td><td><pre> <system> m32\Wbem; <system> d> cry cry cry cry cry cry cry cry cry cr</system></system></pre></td></system>	em32\cmd.e Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Server\80\' <\syste SYSTEM E 15 x86 Family 40 0207 2 .VBE;.JS;.J MP MP Local Settin Local Settin Notify ime	m32\os2\d %SystemR Fools\BINI Foo	II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4> 4> Time Subrited</system>	<pre> <system> m32\Wbem; <system> d> cry cry cry cry cry cry cry cry cry cr</system></system></pre>
simbad Simbad Disabled Stopped Sparrow Disabled Stopped Special I Kernel Driver False True Symc810 Symc8xx Disabled Symc8xx Disabled Symmpi Driver True Sym_hi Disabled Stopped Symmpi Driver True True Sym_hi Disabled Stopped Symmpi Driver True True Sym_hi Disabled Stopped Symmpi Driver True True Sym_hi Driver True Sym_hi Stopped St	Not Avail: OK Not Avail: OK Purpose Utili True c:\winnt\s; Running Bus Driver True Not Avail: OK Not Avail: OK C:\winnt\s; Boot Not Avail: OK Protocol Driv True NC False c:\winnt\s; Manual 3 c:\winnt\s;	able Normal able Normal ty Driver Manual ystem32\dri OK c:\winnt\s; Manual able Normal able Normal ystem32\dri Running able Normal ystem32\dri Running able Vormal ystem32\dri System c:\winnt\s; Manual ystem32\dri System dri System32\dri Stopped	Kernel Dr False Kernel Dr False c:\winnt\sy Running ivers\srv.sy: Normal ystem32\dri Running Kernel Dr False Kernel Dr False ivers\symm OK Kernel Dr False ivers\symm OK Kernel Dr False c:\winnt\sy Running ystem32\dri Stopped ivers\tdipx.so OK	iver False iver False ystem32\dr OK s File Syste False ivers\swent OK iver False iver False iver False iver False iver OK iver False iver False pi.sys Normal iver False ystem32\dr OK ivers\tdasyr OK sys Ignore 0.sys	False ivers\spud.sys Normal m Driver True um.sys Normal False False Kernel False ivers\tepip.sys Normal nc.sys Ignore Kernel False Kernel False	ComSpec Os2LibPat Path %SystemF C:\Program windir OS PROCESS PROCESS PROCESS GenuineIn PROCESS NUMBER PATHEX .COM;,EX TEMP TMP TEMP CLIENTI TMP CLIENTI [Jobs] [Followin [Print] Document Start Time Priority	%SystemIth Root%\system Files\Mic %SystemI Windows SOR_ARCI SOR_LEVE SOR_LEVE GOR_REVI R_OF_PROO I KE;.BAT;.C %SystemI %SystemI %SystemI %USERPI 0\Administ %USERPI 0\Administ g are sub-c E Size Until Time Parameter	Root%\system %SystemI %SystemI em32;%Sys erosoft SQL Root% NT HITECTUR EL TIFIER <system ategories="" cessors="" driver="" eelapsed="" emd;.vbs;="" na<="" of="" owner="" rator="" rofile%\="" root%\tem="" s="" sion="" t="" td=""><td>em32\cmd.e Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Server\80\' <\syste SYSTEM E 15 x86 Family 40 2007 2 .VBE;.JS;.J MP MP Local Settin Local Settin Notify ime</td><td>m32\os2\d %SystemR Fools\BINI Foo</td><td>II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4> 4> Time Subrited</system></td><td><pre> <system> m32\Wbem; <system> 4> cstystem > cs</system></system></pre></td></system>	em32\cmd.e Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Server\80\' <\syste SYSTEM E 15 x86 Family 40 2007 2 .VBE;.JS;.J MP MP Local Settin Local Settin Notify ime	m32\os2\d %SystemR Fools\BINI Foo	II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4> 4> Time Subrited</system>	<pre> <system> m32\Wbem; <system> 4> cstystem > cs</system></system></pre>
simbad Simbad Disabled Stopped Sparrow Disabled Stopped Special I Kernel Driver False True Symc810 Stopped Symc8xx Disabled Symc8xx Disabled Symmpi Driver True Sym_hi Disabled Stopped Symmpi Driver True True Sym_hi Sym_hi Disabled Stopped Symphi Symmpi Driver True True Sym_hi Stopped S	Not Avail: OK Not Avail: OK Purpose Utili True c:\winnt\s; Running Bus Driver True Not Avail: OK Not Avail: OK C:\winnt\s; Boot Not Avail: OK Protocol Driv True NC False c:\winnt\s; Manual 3 c:\winnt\s;	able Normal able Normal ty Driver Manual ystem32\dri OK c:\winnt\s; Manual able Normal able Normal ystem32\dri Running able Normal ystem32\dri Running able Normal ystem32\dri Running able Normal ystem32\dri Running able Normal ystem32\dri Stystem	Kernel Dr False Kernel Dr False c:\winnt\sy Running ivers\srv.sy: Normal ystem32\dri Running Kernel Dr False Kernel Dr False ivers\symm OK Kernel Dr False c:\winnt\sy Running ystem32\dri Stopped ivers\tdipx.soK	river False river False ystem32\dr OK sFile Syste False rivers\swent OK river False river False river False river False river False pi.sys Normal river False ystem32\dr OK rivers\tdasyr OK rivers\tdasyr OK rivers\tdasyr OK sys Ignore	False ivers\spud.sys Normal m Driver True um.sys Normal False False Kernel False False ivers\tcpip.sys Normal nc.sys Ignore Kernel False	ComSpec Os2LibPat Path %SystemF C:\Prograt windir OS PROCESS PROCESS PROCESS PROCESS PROCESS PROCESS PROCESS OF TEMP TEMP TEMP TEMP CLIENTI TMP CLIENTI [Jobs] [Followin [Print] Document Start Time	%SystemIth Root%\system Files\Mic %SystemI Windows SOR_ARCF SOR_LEVE SOR_IDEN Itel SOR_REVI LOF_PROOT KE;.BAT;.C %SystemI %SystemI %USERPI 0\Administ %USERPI 0\Administ g are sub-c	Root%\system %SystemI %SystemI em32;%Sys erosoft SQL Root% NT HITECTUR EL TIFIER <system ategories="" cessors="" driver="" eelapsed="" emd;.vbs;="" na<="" of="" owner="" rator="" rofile%\="" root%\tem="" s="" sion="" t="" td=""><td>em32\cmd.e Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Server\80\' <\syste SYSTEM E 15 x86 Family 40 2007 2 .VBE;.JS;.J MP MP Local Settin Local Settin Notify ime</td><td>m32\os2\d %SystemR Fools\BINI Foo</td><td>II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4> 4> Time Subrited</system></td><td><pre> <system> m32\Wbem; <system> d> cry cry cry cry cry cry cry cry cry cr</system></system></pre></td></system>	em32\cmd.e Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Root%\syste Server\80\' <\syste SYSTEM E 15 x86 Family 40 2007 2 .VBE;.JS;.J MP MP Local Settin Local Settin Notify ime	m32\os2\d %SystemR Fools\BINI Foo	II; oot%\SysteN <system 4=""> 2 Stepping 4> 4> WSH 4> 4> Time Subrited</system>	<pre> <system> m32\Wbem; <system> d> cry cry cry cry cry cry cry cry cry cr</system></system></pre>

No print jobs			dfssvc.exe c:\winnt\system32\dfssvc.exe 1020 8 204800 1413120 7/7/2003 7:37:52 PM 5.00.2195.2841 88.27 KB (90,384 bytes) 2/10/2003 5:29:52 PM
[Network Connections]			explorer.exe c:\winnt\explorer.exe 1144 8 204800 1413120 7/7/2003 7:38:22 PM 5.00.3315.2846 237.27 KB (242,960
Local Name Remote Name Typ E: \\192.168.132.253\e\$ Disk OK		User Name	bytes) 2/10/2003 5:30:17 PM mdm.exe c:\winnt\system32\mdm.exe 1268 8 204800 1413120 7/7/2003 7:38:46 PM 6.00.8424 121.29 KB (124,200 bytes)
CLIENT10\Administrator [Running Tasks]			2/10/2003 11:59:09 AM mmc.exe c:\winnt\system32\mmc.exe 1316 8 204800 1413120 7/7/2003 7:38:53 PM 5.00.2195.2301 589.27 KB (603,408
Working Set Start Time Version Size		Max	bytes) 2/10/2003 5:29:59 PM svchost.exe c:\winnt\system32\svchost.exe 364 8 204800 1413120 7/7/2003 7:39:03 PM 5.00.2134.1 7.77 KB
system idle process Not Available 0 Available Not Available Not Available Unknown	0 Unknown	Not Unknown	(7,952 bytes) 12/7/1999 7:00:00 AM [Loaded Modules]
system Not Available 8 8 Not Available Unknown Unknown Unk	0 mown	1413120	Name Version Size File Date Manufacturer Path
Smss.exe c:\winnt\system32\smss.exe 164 1413120 7/7/2003 7:37:18 PM 5.00.2195.2901 bytes) 12/7/1999 7:00:00 AM	11	204800 (45,328	tapisrv.dll 5.00.2195.2955 169.27 KB (173,328 bytes) 2/10/2003 5:30:15 PM Microsoft Corporation c:\winnt\system32\tapisrv.dll
csrss.exe Not Available 188 13	Not Avail		wbemprox.dll 1.50.1085.0045 40.08 KB (41,040 bytes)
Not Available 7/7/2003 7:37:22 PM Unk winlogon.exe c:\winnt\system32\winlogo		Unknown 13	2/10/2003 5:30:22 PM Microsoft Corporation c:\winnt\system32\wbem\wbemprox.dll
204800 1413120 7/7/2003 7:37:24 PM 5.00		173.77 KB	rassapi.dll 5.00.2188.1 14.27 KB (14,608 bytes) 12/7/1999
(177,936 bytes) 12/7/1999 7:00:00 AM services.exe c:\winnt\system32\services	s.exe 236	9	7:00:00 AM Microsoft Corporation c:\winnt\system32\rassapi.dll
204800 1413120 7/7/2003 7:37:25 PM 5.00 (88,848 bytes) 12/7/1999 7:00:00 AM		86.77 KB	adsnt.dll 5.00.2195.2778 195.27 KB (199,952 bytes) 2/10/2003 5:29:48 PM Microsoft Corporation
lsass.exe c:\winnt\system32\lsass.exe 248	9	204800	c:\winnt\system32\adsnt.dll
1413120 7/7/2003 7:37:25 PM 5.00.2195.2964 bytes) 12/7/1999 7:00:00 AM	32.77 KB	(33,552	dbghelp.dll 5.00.2195.2104 159.27 KB (163,088 bytes) 5/4/2001 1:05:02 PM Microsoft Corporation
svchost.exe c:\winnt\system32\svchost	.exe 416	8	c:\winnt\system32\dbghelp.dll
204800 1413120 7/7/2003 7:37:29 PM 5.00 (7,952 bytes) 12/7/1999 7:00:00 AM	0.2134.1	7.77 KB	localsec.dll 5.00.2195.2130 230.27 KB (235,792 bytes) 2/10/2003 5:29:58 PM Microsoft Corporation
spoolsv.exe c:\winnt\system32\spoolsv		8 42.77.KD	c:\winnt\system32\localsec.dll
204800 1413120 7/7/2003 7:37:30 PM 5.00 (44,816 bytes) 2/10/2003 11:47:14 AM	0.2161.1	43.77 KB	devmgr.dll 5.00.2166.1 215.77 KB (220,944 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation
msdtc.exe c:\winnt\system32\msdtc.exe 476		204800	c:\winnt\system32\devmgr.dll
1413120 7/7/2003 7:37:30 PM 1999.9.3421.3 2/10/2003 11:57:24 AM		6,928 bytes)	filemgmt.dll 5.00.2195.2165 287.27 KB (294,160 bytes) 2/10/2003 5:29:55 PM Microsoft Corporation
svchost.exe c:\winnt\system32\svchost 204800 1413120 7/7/2003 7:37:33 PM 5.00		8 7.77 KB	c:\winnt\system32\filemgmt.dll pdh.dll 5.00.2195.2739 147.77 KB (151,312 bytes) 2/10/2003
(7,952 bytes) 12/7/1999 7:00:00 AM			5:30:10 PM Microsoft Corporation
llssrv.exe c:\winnt\system32\llssrv.exe 636 1413120 7/7/2003 7:37:33 PM 5.00.2195.2649		204800 B (117,008	c:\winnt\system32\pdh.dll smlogcfg.dll 5.00.2195.2485 273.27 KB (279,824 bytes)
bytes) 5/4/2001 1:05:02 PM	114.2710	B (117,000	2/10/2003 5:30:14 PM Microsoft Corporation
regsvc.exe c:\winnt\system32\regsvc.exe 680 1413120 7/7/2003 7:37:34 PM 5.00.2195.2104		204800	c:\winnt\system32\smlogcfg.dll cabinet.dll 5.00.2147.1 54.77 KB (56,080 bytes) 12/7/1999
bytes) 2/10/2003 5:30:11 PM	03.27 KD	(00,032	7:00:00 AM Microsoft Corporation
rsys.exe Not Available 784 8	Not Avail		c:\winnt\system32\cabinet.dll
Not Available 7/7/2003 7:37:35 PM Unk rsvp.exe c:\winnt\system32\rsvp.exe 696		204800	msinfo32.dll 5.00.2177.1 312.27 KB (319,760 bytes) 2/10/2003 5:01:28 PM Microsoft Corporation c:\program
1413120 7/7/2003 7:37:50 PM 5.00.2167.1	172.77 K	B (176,912	files\common files\microsoft shared\msinfo32.dll
bytes) 12/7/1999 7:00:00 AM mstask.exec:\winnt\system32\mstask.exe 852	8	204800	riched20.dll 5.30.23.1205 421.27 KB (431,376 bytes) 2/10/2003 5:30:11 PM Microsoft Corporation
1413120 7/7/2003 7:37:50 PM 4.71.2195.1		B (118,032	c:\winnt\system32\riched20.dll
bytes) 2/10/2003 5:30:05 PM tcpsvcs.exe c:\winnt\system32\tcpsvcs	.exe 884	8	riched32.dll 5.00.2134.1 3.77 KB (3,856 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation
204800 1413120 7/7/2003 7:37:50 PM 5.00		24.77 KB	c:\winnt\system32\riched32.dll
(25,360 bytes) 12/7/1999 7:00:00 AM		020	els.dll 5.00.2175.1 151.27 KB (154,896 bytes) 12/7/1999
winngmt.exe c:\winnt\system32\wbem\\ 204800 1413120 7/7/2003 7:37:50 PM 1.50	•	920 8 192.08 KB	7:00:00 AM Microsoft Corporation c:\winnt\system32\els.dll
(196,685 bytes) 2/10/2003 5:30:22 PM			ntmsmgr.dll 1,0,0,1 427.77 KB (438,032 bytes) 12/7/1999
inetinfo.exe c:\winnt\system32\inetsrv\ 204800 1413120 7/7/2003 7:37:50 PM 5.00		952 8 (14.608	7:00:00 AM Microsoft Corporation and HighGround Systems, Inc. c:\winnt\system32\ntmsmgr.dll
bytes) 2/10/2003 5:31:05 PM	7.0707 1 7. 2/ KD	(17,000	C. William System 52 ministrigit dir

mmfutil.dll1.50.1085.0000 32.06 KB (32,829 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation	mshtml.dll 5.00.3315.2870 2.24 MB (2,345,232 bytes) 2/10/2003 5:30:00 PM Microsoft Corporation
c:\winnt\system32\mmfutil.dll	c:\winnt\system32\mshtml.dll
logdrive.dll 1.50.1085.0000 200.06 KB (204,863 bytes)	mlang.dll 5.00.3103.1000 510.77 KB (523,024 bytes) 2/10/2003
12/7/1999 7:00:00 AM Microsoft Corporation	5:29:59 PM Microsoft Corporation
1	F
c:\winnt\system32\logdrive.dll	c:\winnt\system32\mlang.dll
dfrgres.dll 5.00.2150.1 27.50 KB (28,160 bytes) 12/7/1999	urlmon.dll 5.00.3315.1000 441.27 KB (451,856 bytes) 2/10/2003
7:00:00 AM Executive Software International, Inc.	5:30:15 PM Microsoft Corporation
c:\winnt\system32\dfrgres.dll	c:\winnt\system32\urlmon.dll
dfrgsnap.dll 5.00.2195.2104 41.77 KB (42,768 bytes)	linkinfo.dll5.00.2134.1 15.77 KB (16,144 bytes) 12/7/1999
2/10/2003 5:29:52 PM Executive Software International, Inc.	7:00:00 AM Microsoft Corporation
c:\winnt\system32\dfrgsnap.dll	c:\winnt\system32\linkinfo.dll
dmdskres.dll 2195.2104.297.3 119.50 KB (122,368 bytes)	msi.dll 2.0.2600.0 1.90 MB (1,991,168 bytes) 2/10/2003 5:30:01 PM
2/10/2003 5:29:52 PM Microsoft Corp., VERITAS Software	Microsoft Corporation c:\winnt\system32\msi.dll
c:\winnt\system32\dmdskres.dll	faxshell.dll5.00.2134.1 8.27 KB (8,464 bytes) 12/7/1999 7:00:00 AM
dmutil.dll 2195.2104.297.3 42.27 KB (43,280 bytes) 2/10/2003	Microsoft Corporation c:\winnt\system32\faxshell.dll
5:29:52 PM VERITAS Software Corp.	msacm32.dll 5.00.2134.1 65.27 KB (66,832 bytes)
c:\winnt\system32\dmutil.dll	12/7/1999 7:00:00 AM Microsoft Corporation
ntmsapi.dll5.00.1948.1 51.77 KB (53,008 bytes) 2/10/2003	c:\winnt\system32\msacm32.dll
5:30:08 PM Microsoft Corporation	avifil32.dll5.00.2134.1 76.27 KB (78,096 bytes) 12/7/1999
c:\winnt\system32\ntmsapi.dll	7:00:00 AM Microsoft Corporation
dmdskmgr.dll 2215.2215.297.3 160.27 KB (164,112 bytes)	c:\winnt\system32\avifil32.dll
2/10/2003 5:29:52 PM Microsoft Corp., VERITAS Software	
c:\winnt\system32\dmdskmgr.dll	12/7/1999 7:00:00 AM Microsoft Corporation
mycomput.dll 5.00.2134.1 107.77 KB (110,352 bytes)	c:\winnt\system32\msvfw32.dll
12/7/1999 7:00:00 AM Microsoft Corporation	docprop2.dll 5.00.2178.1 297.77 KB (304,912 bytes)
c:\winnt\system32\mycomput.dll	12/7/1999 7:00:00 AM Microsoft Corporation
mmcndmgr.dll 5.00.2178.1 815.27 KB (834,832 bytes)	c:\winnt\system32\docprop2.dll
12/7/1999 7:00:00 AM Microsoft Corporation	hhsetup.dll4.74.8702 66.27 KB (67,856 bytes) 12/7/1999 7:00:00 AM
c:\winnt\system32\mmcndmgr.dll	Microsoft Corporation c:\winnt\system32\hhsetup.dll
mmc.exe 5.00.2195.2301 589.27 KB (603,408 bytes) 2/10/2003	mmcshext.dll 5.00.2153.1 24.27 KB (24,848 bytes)
5:29:59 PM Microsoft Corporation	12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\mmc.exe	c:\winnt\system32\mmcshext.dll
mdm.exe 6.00.8424 121.29 KB (124,200 bytes) 2/10/2003 11:59:09	browselc.dll 5.00.3315.2846 34.50 KB (35,328 bytes)
AM Microsoft Corporation c:\winnt\system32\mdm.exe	2/10/2003 5:29:48 PM Microsoft Corporation
imm32.dll 5.00.2195.2821 94.27 KB (96,528 bytes) 2/10/2003	c:\winnt\system32\browselc.dll
5:29:56 PM Microsoft Corporation	powrprof.dll 5.00.3103.1000 13.27 KB (13,584 bytes)
c:\winnt\system32\imm32.dll	2/10/2003 5:30:10 PM Microsoft Corporation
netplwiz.dll 5.00.2195.2370 169.77 KB (173,840 bytes)	c:\winnt\system32\powrprof.dll
2/10/2003 5:30:07 PM Microsoft Corporation	batmeter.dll 5.00.3103.1000 20.27 KB (20,752 bytes)
c:\winnt\system32\netplwiz.dll	2/10/2003 5:29:48 PM Microsoft Corporation
y i	c:\winnt\system32\batmeter.dll
7:00:00 AM Microsoft Corporation	stobject.dll5.00.2195.2780 79.27 KB (81,168 bytes) 2/10/2003
c:\winnt\system32\netmsg.dll	5:30:14 PM Microsoft Corporation
netui2.dll 5.00.2134.1 280.27 KB (286,992 bytes) 12/7/1999	c:\winnt\system32\stobject.dll
7:00:00 AM Microsoft Corporation	webcheck.dll 5.00.3315.1000 251.77 KB (257,808 bytes)
c:\winnt\system32\netui2.dll	2/10/2003 5:30:16 PM Microsoft Corporation
mprui.dll 5.00.2195.2104 54.77 KB (56,080 bytes) 2/10/2003	c:\winnt\system32\webcheck.dll
5:29:59 PM Microsoft Corporation	ntshrui.dll 5.00.2134.1 46.77 KB (47,888 bytes) 12/7/1999
c:\winnt\system32\mprui.dll	7:00:00 AM Microsoft Corporation
imgutil.dll 5.00.3315.2870 30.77 KB (31,504 bytes) 2/10/2003	c:\winnt\system32\ntshrui.dll
5:29:56 PM Microsoft Corporation	mydocs.dll5.00.2920.0000 55.77 KB (57,104 bytes) 12/7/1999
c:\winnt\system32\imgutil.dll	7:00:00 AM Microsoft Corporation
webvw.dll 5.00.2920.0000 1.06 MB (1,115,408 bytes) 12/7/1999	c:\winnt\system32\mydocs.dll
7:00:00 AM Microsoft Corporation	browseui.dll 5.00.3315.2846 788.77 KB (807,696 bytes)
c:\winnt\system32\webvw.dll	2/10/2003 5:29:48 PM Microsoft Corporation
msls31.dll 3.10.337.0 145.27 KB (148,752 bytes) 12/7/1999 7:00:00 AM	c:\winnt\system32\browseui.dll
Microsoft Corporation c:\winnt\system32\msls31.dll	shdocvw.dll 5.00.3315.2879 1.05 MB (1,104,144 bytes)
wininet.dll 5.00.3315.1000 456.77 KB (467,728 bytes) 2/10/2003	2/10/2003 5:30:13 PM Microsoft Corporation
i i i i i i i i i i i i i i i i i i i	<u> </u>
5:30:16 PM Microsoft Corporation	c:\winnt\system32\shdocvw.dll
c:\winnt\system32\wininet.dll	explorer.exe 5.00.3315.2846 237.27 KB (242,960 bytes)
msdbg.dll 6.00.8424 67.50 KB (69,120 bytes) 2/10/2003 11:59:10	2/10/2003 5:30:17 PM Microsoft Corporation
AM Microsoft Corporation c:\winnt\system32\msdbg.dll	c:\winnt\explorer.exe
shdoclc.dll 5.00.3315.2879 324.50 KB (332,288 bytes) 2/10/2003	
5 20 12 DM M' 0 C /	dfssvc.exe 5.00.2195.2841 88.27 KB (90,384 bytes) 2/10/2003
5:30:13 PM Microsoft Corporation	5:29:52 PM Microsoft Corporation
c:\winnt\system32\shdoclc.dll	5:29:52 PM Microsoft Corporation c:\winnt\system32\dfssvc.exe
c:\winnt\system32\shdoclc.dll pdm.dll 6.00.8424 179.27 KB (183,574 bytes) 2/10/2003 11:59:10	5:29:52 PM Microsoft Corporation c:\winnt\system32\dfssvc.exe iislog.dll 5.00.0984 75.27 KB (77,072 bytes) 2/10/2003 5:31:05 PM
c:\winnt\system32\shdoclc.dll	5:29:52 PM Microsoft Corporation c:\winnt\system32\dfssvc.exe

httpext.dll 0.9.3940.21 435.27 KB (445,712 bytes) 2/10/2003	ntlanman.dll 5.00.2157.1 35.27 KB (36,112 bytes)
5:31:04 PM Microsoft Corporation	12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\inetsrv\httpext.dll	c:\winnt\system32\ntlanman.dll
rpcproxy.dll 5.00.2195.2780 16.27 KB (16,656 bytes)	wshnetbs.dll 5.00.2134.1 7.77 KB (7,952 bytes) 12/7/1999
2/10/2003 5:30:58 PM Microsoft Corporation	7:00:00 AM Microsoft Corporation
c:\winnt\system32\rpcproxy\rpcproxy.dll	c:\winnt\system32\wshnetbs.dll
fpexedll.dll 4.0.2.4324 20.06 KB (20,541 bytes) 2/10/2003	ntmarta.dll 5.00.2195.2862 98.77 KB (101,136 bytes) 2/10/2003
5:30:57 PM Microsoft Corporation c:\program	5:30:08 PM Microsoft Corporation
files\common files\microsoft shared\web server extensions\40\bin\fpexedll.dll	c:\winnt\system32\ntmarta.dll
md5filt.dll 5.00.0984 32.77 KB (33,552 bytes) 2/10/2003 5:31:05 PM	provthrd.dll 1.50.1085.0000 68.07 KB (69,708 bytes)
Microsoft Corporation c:\winnt\system32\inetsrv\md5filt.dll	2/10/2003 5:01:22 PM Microsoft Corporation
gzip.dll 5.00.0984 30.27 KB (30,992 bytes) 2/10/2003 5:31:04 PM	c:\winnt\system32\wbem\provthrd.dll
Microsoft Corporation c:\winnt\system32\inetsrv\gzip.dll	ntevt.dll 1.50.1085.0000 192.06 KB (196,669 bytes) 12/7/1999
compfilt.dll 5.00.0984 22.77 KB (23,312 bytes) 2/10/2003	7:00:00 AM Microsoft Corporation
5:31:04 PM Microsoft Corporation	c:\winnt\system32\wbem\ntevt.dll
c:\winnt\system32\inetsrv\compfilt.dll	perfos.dll 5.00.2155.1 21.27 KB (21,776 bytes) 12/7/1999
sspifilt.dll 5.00.0984 43.27 KB (44,304 bytes) 2/10/2003 5:31:06 PM	7:00:00 AM Microsoft Corporation
Microsoft Corporation c:\winnt\system32\inetsrv\sspifilt.dll	c:\winnt\system32\perfos.dll
iscomlog.dll 5.00.0984 24.77 KB (25,360 bytes) 2/10/2003	psapi.dll 5.00.2134.1 28.27 KB (28,944 bytes) 12/7/1999
5:31:05 PM Microsoft Corporation	7:00:00 AM Microsoft Corporation
c:\winnt\system32\inetsrv\iscomlog.dll	c:\winnt\system32\psapi.dll
lonsint.dll 5.00.0984 11.77 KB (12,048 bytes) 2/10/2003 5:31:05 PM	framedyn.dll 1.50.1085.0000 164.05 KB (167,992 bytes)
Microsoft Corporation c:\winnt\system32\inetsrv\lonsint.dll	12/7/1999 7:00:00 AM Microsoft Corporation
inetsloc.dll5.00.0984 20.27 KB (20,752 bytes) 2/10/2003 5:29:56 PM	c:\winnt\system32\wbem\framedyn.dll
Microsoft Corporation c:\winnt\system32\inetsloc.dll	cimwin32.dll 1.50.1085.0038 1.02 MB (1,073,232 bytes)
iisfecnv.dll5.00.0984 7.27 KB (7,440 bytes) 2/10/2003 11:57:51 AM	2/10/2003 5:30:21 PM Microsoft Corporation
Microsoft Corporation c:\winnt\system32\inetsrv\iisfecnv.dll	c:\winnt\system32\wbem\cimwin32.dll
isatq.dll 5.00.0984 60.27 KB (61,712 bytes) 2/10/2003 5:31:05 PM	wbemsvc.dll 1.50.1085.0007 40.07 KB (41,036 bytes)
Microsoft Corporation c:\winnt\system32\inetsrv\isatq.dll	2/10/2003 5:30:22 PM Microsoft Corporation
infocomm.dll 5.00.0984 238.27 KB (243,984 bytes) 2/10/2003	c:\winnt\system32\wbem\wbemsvc.dll
5:31:05 PM Microsoft Corporation	wbemess.dll 1.50.1085.0039 364.07 KB (372,804 bytes)
c:\winnt\system32\inetsrv\infocomm.dll	2/10/2003 5:30:22 PM Microsoft Corporation
w3svc.dll 5.00.0984 343.27 KB (351,504 bytes) 2/10/2003 5:31:06 PM	c:\winnt\system32\wbem\wbemess.dll
Microsoft Corporation c:\winnt\system32\inetsrv\w3svc.dll	fastprox.dll 1.50.1085.0037 144.08 KB (147,536 bytes)
security.dll5.00.2154.1 5.77 KB (5,904 bytes) 12/7/1999 7:00:00 AM	2/10/2003 5:30:21 PM Microsoft Corporation
Microsoft Corporation c:\winnt\system32\security.dll	c:\winnt\system32\wbem\fastprox.dll
svcext.dll 5.00.0984 39.77 KB (40,720 bytes) 2/10/2003 5:31:06 PM	wbemcore.dll 1.50.1085.0036 628.07 KB (643,140 bytes)
Microsoft Corporation c:\winnt\system32\inetsrv\svcext.dll	2/10/2003 5:30:22 PM Microsoft Corporation
admexs.dll 5.00.0984 27.77 KB (28,432 bytes) 2/10/2003 5:31:04 PM	c:\winnt\system32\wbem\wbemcore.dll
Microsoft Corporation c:\winnt\system32\inetsrv\admexs.dll	wbemcomn.dll 1.50.1085.0021 692.07 KB (708,675 bytes)
wamreg.dll5.00.0984 45.77 KB (46,864 bytes) 2/10/2003 5:31:06 PM	2/10/2003 5:30:21 PM Microsoft Corporation
Microsoft Corporation c:\winnt\system32\inetsrv\wamreg.dll metadata.dll 5.00.0984 68.77 KB (70,416 bytes) 2/10/2003	c:\winnt\system32\wbem\wbemcomn.dll winmgmt.exe
metadata.dll 5.00.0984 68.77 KB (70,416 bytes) 2/10/2003 5:31:05 PM Microsoft Corporation	winmgmt.exe 1.50.1085.0029 192.08 KB (196,685 bytes) 2/10/2003 5:30:22 PM Microsoft Corporation
c:\winnt\system32\inetsrv\metadata.dll	c:\winnt\system32\wbem\winmgmt.exe
iismap.dll 5.00.0984 55.77 KB (57,104 bytes) 2/10/2003 5:29:56 PM	simptep.dll5.00.2134.1 19.27 KB (19,728 bytes) 2/10/2003
Microsoft Corporation c:\winnt\system32\iismap.dll	11:57:19 AM Microsoft Corporation
nsepm.dll 5.00.0984 43.27 KB (44,304 bytes) 2/10/2003 5:31:05 PM	c:\winnt\system32\simptcp.dll
Microsoft Corporation c:\winnt\system32\inetsrv\nsepm.dll	tcpsvcs.exe 5.00.2134.1 24.77 KB (25,360 bytes)
coadmin.dll 5.00.0984 39.27 KB (40,208 bytes) 2/10/2003	12/7/1999 7:00:00 AM Microsoft Corporation
5:31:04 PM Microsoft Corporation	c:\winnt\system32\tcpsvcs.exe
c:\winnt\system32\inetsry\coadmin.dll	msidle.dll 5.00.2920.0000 6.27 KB (6,416 bytes) 12/7/1999 7:00:00 AM
iisadmin.dll 5.00.0984 15.27 KB (15,632 bytes) 2/10/2003	Microsoft Corporation c:\winnt\system32\msidle.dll
5:31:05 PM Microsoft Corporation	mstask.exe4.71.2195.1 115.27 KB (118,032 bytes) 2/10/2003
c:\winnt\system32\inetsrv\iisadmin.dll	5:30:05 PM Microsoft Corporation
rpcref.dll 5.00.0984 4.27 KB (4,368 bytes) 2/10/2003 5:31:06 PM	c:\winnt\system32\mstask.exe
Microsoft Corporation c:\winnt\system32\inetsrv\rpcref.dll	traffic.dll 5.00.2139.1 30.77 KB (31,504 bytes) 12/7/1999
iisrtl.dll 5.00.0984 119.77 KB (122,640 bytes) 2/10/2003 5:29:56 PM	7:00:00 AM Microsoft Corporation
Microsoft Corporation c:\winnt\system32\iisrtl.dll	c:\winnt\system32\traffic.dll
inetinfo.exe 5.00.0984 14.27 KB (14,608 bytes) 2/10/2003	rsvp.exe 5.00.2167.1 172.77 KB (176,912 bytes) 12/7/1999
5:31:05 PM Microsoft Corporation	7:00:00 AM Microsoft Corporation
c:\winnt\system32\inetsrv\inetinfo.exe	c:\winnt\system32\rsvp.exe
netui1.dll 5.00.2134.1 210.27 KB (215,312 bytes) 12/7/1999	regsvc.exe 5.00.2195.2104 65.27 KB (66,832 bytes) 2/10/2003
7:00:00 AM Microsoft Corporation	5:30:11 PM Microsoft Corporation
c:\winnt\system32\netui1.dll	c:\winnt\system32\regsvc.exe
netui0.dll 5.00.2134.1 70.27 KB (71,952 bytes) 12/7/1999	llsrpc.dll 5.00.2149.1 45.77 KB (46,864 bytes) 12/7/1999
7:00:00 AM Microsoft Corporation	7:00:00 AM Microsoft Corporation
c:\winnt\system32\netui0.dll	c:\winnt\system32\llsrpc.dll

llssrv.exe 5.00.2195.2649 114.27 KB (117,008 bytes) 5/4/200 1:05:02 PM Microsoft Corporation	5:30:03 PM Microsoft Corporation
c:\winnt\system32\llssrv.exe rasdlg.dll 5.00.2195.2671 514.27 KB (526,608 bytes) 12/7/199 7:00:00 AM Microsoft Corporation	2/10/2003 5:30:03 PM Microsoft Corporation
c:\winnt\system32\rasdlg.dll netcfgx.dll 5.00.2195.2228 534.77 KB (547,600 bytes) 2/10/200 5:30:06 PM Microsoft Corporation	7:00:00 AM Microsoft Corporation
c:\winnt\system32\netcfgx.dll rasmans.dll	c:\winnt\system32\iasrad.dll iassam.dll 5.00.2160.1 96.27 KB (98,576 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\rasmans.dll netshell.dll5.00.2195.2779 457.27 KB (468,240 bytes) 2/10/200 5:30:07 PM Microsoft Corporation c:\winnt\system32\netshell.dll	c:\winnt\system32\iassam.dll iasads.dll 5.00.2134.1 73.77 KB (75,536 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\iasads.dll
netman.dll 5.00.2195.2779 89.27 KB (91,408 bytes) 2/10/200 5:30:06 PM Microsoft Corporation c:\winnt\system32\netman.dll	
ntmsdba.dll 5.00.2195.2779 167.27 KB (171,280 bytes) 2/10/2003 5:30:08 PM Microsoft Corporation c:\winnt\system32\ntmsdba.dll	iaspolcy.dll 5.00.2134.1 25.27 KB (25,872 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\iaspolcy.dll
sens.dll 5.00.2163.1 36.77 KB (37,648 bytes) 12/7/1997.00:00 AM Microsoft Corporation c:\winnt\system32\sens.dll	
iashlpr.dll 5.00.2184.1 33.27 KB (34,064 bytes) 12/7/1997:00:00 AM Microsoft Corporation c:\winnt\system32\iashlpr.dll	
iasacct.dll 5.00.2134.1 28.27 KB (28,944 bytes) 12/7/1997:00:00 AM Microsoft Corporation c:\winnt\system32\iasacct.dll	99 ias.dll 5.00.2134.1 7.27 KB (7,440 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation es.dll 2000.2.3471.1 222.27 KB (227,600 bytes) 2/10/2003
iasuserr.dll5.00.2134.1 25.77 KB (26,384 bytes) 12/7/1997:00:00 AM Microsoft Corporation c:\winnt\system32\iasuserr.dll	99 5:29:54 PM Microsoft Corporation c:\winnt\system32\es.dll mtxoci.dll 2000.2.3471.1 101.77 KB (104,208 bytes) 2/10/2003
iasnap.dll 5.00.2195.2104 58.77 KB (60,176 bytes) 2/10/200 5:29:55 PM Microsoft Corporation c:\winnt\system32\iasnap.dll	03 5:30:06 PM Microsoft Corporation c:\winnt\system32\mtxoci.dll resutils.dll 5.00.2195.2787 39.77 KB (40,720 bytes) 2/10/2003
iaspipe.dll 5.00.2134.1 41.77 KB (42,768 bytes) 12/7/1997:00:00 AM Microsoft Corporation c:\winnt\system32\iaspipe.dll	99 5:30:11 PM Microsoft Corporation c:\winnt\system32\resutils.dll clusapi.dll 5.00.2195.2104 54.27 KB (55,568 bytes) 2/10/2003
expsrv.dll 6.0.8540 370.27 KB (379,152 bytes) 2/10/2003 5:29:54 Microsoft Corporation c:\winnt\system32\expsrv.dll vbajet32.dll 6.1.8268 30.27 KB (30,992 bytes) 2/10/200	c:\winnt\system32\clusapi.dll msvcp50.dll 5.00.7051 552.50 KB (565,760 bytes) 12/7/1999
5:30:16 PM Microsoft Corporation c:\winnt\system32\vbajet32.dll msjtes40.dll 4.00.4229.0 236.27 KB (241,936 bytes)	7:00:00 AM Microsoft Corporation c:\winnt\system32\msvcp50.dll xolehlp.dll 1999.9.3421.3 17.27 KB (17,680 bytes) 2/10/2003
2/10/2003 5:30:04 PM Microsoft Corporation c:\winnt\system32\msjtes40.dll oledb32r.dll 2.70.9001.0 built by: Lab06_N(dagbuild) 64.00 K	· / • /
(65,536 bytes) 2/10/2003 5:55:10 PM Microsoft Corporat c:\program files\common files\system\ole db\oledb32r.dll comdlg32.dll 5.00.3103.1000 236.77 KB (242,448 bytes)	c:\winnt\system32\msdtclog.dll mtxclu.dll 2000.2.3471.1 51.27 KB (52,496 bytes) 2/10/2003
12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\comdlg32.dll msdart.dll 2.71.9031.4 built by: Lab06_N(dagbuild) 124.00 KB (126,97 bytes) 9/27/2002 12:22:40 PM Microsoft Corporation	5:30:06 PM Microsoft Corporation c:\winnt\system32\mtxclu.dll msdtcprx.dll 2000.2.3471.1 665.77 KB (681,744 bytes) 2/10/2003 5:30:00 PM Microsoft Corporation
c:\winnt\system32\msdart.dll oledb32.dll 2.71.9031.4 built by: Lab06_N(dagbuild) 408.00 I (417,792 bytes) 9/27/2002 12:22:42 PM Microsoft Corporat	c:\winnt\system32\msdtcprx.dll txfaux.dll 2000.2.3471.1 374.27 KB (383,248 bytes) 2/10/2003
c:\program files\common files\system\ole db\oledb32.dll msjint40.dll 4.00.2927.2 148.27 KB (151,824 bytes) 2/10/2003 5:30:03 PM Microsoft Corporation	c:\winnt\system32\txfaux.dll msdtctm.dll 2000.2.3471.1 1.07 MB (1,120,528 bytes) 2/10/2003 5:30:00 PM Microsoft Corporation
c:\winnt\system32\msjint40.dll msjter40.dll	c:\winnt\system32\msdtctm.dll msdtc.exe 1999.9.3421.3 6.77 KB (6,928 bytes) 2/10/2003 11:57:24 AM Microsoft Corporation c:\winnt\system32\msdtc.exe
c:\winnt\system32\msjter40.dll mswstr10.dll	wmi.dll 5.00.2191.1 6.27 KB (6,416 bytes) 12/7/1999 7:00:00 AM c:\winnt\system32\wmi.dll

admwprox.dll 5.00.0984 31.77 KB (32,528 bytes) 2 11:57:52 AM Microsoft Corporation	2/10/2003	sfmapi.dll 5.00.2134.1 38.77 KB (39,696 bytes) 12/7/199 7:00:00 AM Microsoft Corporation)9
c:\winnt\system32\admwprox.dll		c:\winnt\system32\sfmapi.dll	
inetpp.dll 5.00.2195.2842 65.27 KB (66,832 bytes) 2	2/10/2003	rassfm.dll 5.00.2195.2671 21.27 KB (21,776 bytes) 2/10/200)3
5:29:56 PM Microsoft Corporation		5:30:11 PM Microsoft Corporation	
c:\winnt\system32\inetpp.dll		c:\winnt\system32\rassfm.dll	
win32spl.dll 5.00.2195.2780 92.27 KB (94,480 bytes	s)	mpr.dll 5.00.2195.2779 53.27 KB (54,544 bytes) 2/10/200	13
12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\win32spl.dll		5:29:59 PM Microsoft Corporation c:\winnt\system32\mpr.dll	
*	2/10/2003	rsabase.dll 5.00.2195.2228 128.27 KB (131,344 bytes) 5/4/2001	ı
5:30:15 PM Microsoft Corporation	10/2003	1:05:02 PM Microsoft Corporation	•
c:\winnt\system32\usbmon.dll		c:\winnt\system32\rsabase.dll	
3	2/10/2003	schannel.dll 5.00.2195.2922 138.27 KB (141,584 bytes)	
5:30:15 PM Microsoft Corporation		5/4/2001 1:05:02 PM Microsoft Corporation	
c:\winnt\system32\tcpmon.dll		c:\winnt\system32\schannel.dll	
	1/30/1999	netlogon.dll 5.00.2195.2865 357.77 KB (366,352 bytes)	
6:39:36 PM Microsoft Corporation		2/10/2003 5:30:06 PM Microsoft Corporation	
c:\winnt\system32\pjlmon.dll	`	c:\winnt\system32\netlogon.dll	
cnbjmon.dll 5.00.2134.1 43.77 KB (44,816 bytes	s)	kerberos.dll 5.00.2195.2913 198.77 KB (203,536 bytes)	
11/30/1999 6:38:48 PM Microsoft Corporation c:\winnt\system32\cnbjmon.dll		2/10/2003 5:29:58 PM Microsoft Corporation c:\winnt\system32\kerberos.dll	
, , , , , , , , , , , , , , , , , , ,	2/7/1999	msprivs.dll5.00.2154.1 41.50 KB (42,496 bytes) 12/7/199	99
7:00:00 AM Microsoft Corporation	2///1999	7:00:00 AM Microsoft Corporation	
c:\winnt\system32\localspl.dll		c:\winnt\system32\msprivs.dll	
1	2/10/2003	samsrv.dll 5.00.2195.2918 369.77 KB (378,640 bytes) 12/7/199	99
11:47:14 AM Microsoft Corporation		7:00:00 AM Microsoft Corporation	
c:\winnt\system32\spoolss.dll		c:\winnt\system32\samsrv.dll	
spoolsv.exe 5.00.2161.1 43.77 KB (44,816 bytes	s)	lsasrv.dll 5.00.2195.2964 492.77 KB (504,592 bytes) 12/7/199	19
2/10/2003 11:47:14 AM Microsoft Corporation		7:00:00 AM Microsoft Corporation	
c:\winnt\system32\spoolsv.exe	1/10/2002	c:\winnt\system32\lsasrv.dll	00
rpcss.dll 5.00.2195.2815 231.27 KB (236,816 bytes) 2 5:30:12 PM Microsoft Corporation	2/10/2003	lsass.exe 5.00.2195.2964 32.77 KB (33,552 bytes) 12/7/199 7:00:00 AM Microsoft Corporation	19
c:\winnt\system32\rpcss.dll		c:\winnt\system32\lsass.exe	
svchost.exe 5.00.2134.1 7.77 KB (7,952 bytes) 1	2/7/1999	wmicore.dll 5.00.2195.2842 72.27 KB (74,000 bytes)	
7:00:00 AM Microsoft Corporation	2,,,,	2/10/2003 5:30:17 PM Microsoft Corporation	
c:\winnt\system32\svchost.exe		c:\winnt\system32\wmicore.dll	
(, , , , , , , , , , , , , , , , , , ,	2/10/2003	rasadhlp.dll 5.00.2168.1 7.27 KB (7,440 bytes) 12/7/199	19
5:31:00 PM Microsoft Corporation		7:00:00 AM Microsoft Corporation	
c:\winnt\system32\dssenh.dll		c:\winnt\system32\rasadhlp.dll	
	2/10/2003	winrnr.dll 5.00.2160.1 18.77 KB (19,216 bytes) 12/7/199	19
5:30:08 PM Microsoft Corporation c:\winnt\system32\oakley.dll		7:00:00 AM Microsoft Corporation c:\winnt\system32\winrr.dll	
	2/7/1999	rnr20.dll 5.00.2195.2871 35.77 KB (36,624 bytes) 2/10/200)3
7:00:00 AM Microsoft Corporation	2///1999	5:30:11 PM Microsoft Corporation	.5
c:\winnt\system32\mfc42u.dll		c:\winnt\system32\rnr20.dll	
polagent.dll 5.00.2183.1 108.27 KB (110,864 by	rtes)	wshtcpip.dll 5.00.2195.2104 17.27 KB (17,680 bytes)	
12/7/1999 7:00:00 AM Microsoft Corporation		2/10/2003 5:30:17 PM Microsoft Corporation	
c:\winnt\system32\polagent.dll		c:\winnt\system32\wshtcpip.dll	
	2/10/2003	msafd.dll 5.00.2195.2779 106.77 KB (109,328 bytes) 2/10/200	13
5:30:12 PM Microsoft Corporation		5:29:59 PM Microsoft Corporation	
c:\winnt\system32\scecli.dll atl.dll 3.00.8449 57.56 KB (58,938 bytes) 12/7/1999 7	·00·00 AM	c:\winnt\system32\msafd.dll mswsock.dll 5.00.2195.2871 62.77 KB (64,272 bytes)	
Microsoft Corporation c:\winnt\system32\atl.dll	.00.00 AW	2/10/2003 5:30:05 PM Microsoft Corporation	
	2/10/2003	c:\winnt\system32\mswsock.dll	
5:29:50 PM Microsoft Corporation		ntlsapi.dll 5.00.2134.1 6.77 KB (6,928 bytes) 12/7/1999 7:00:00 A	4M
c:\winnt\system32\certcli.dll		Microsoft Corporation c:\winnt\system32\ntlsapi.dll	
	2/10/2003	msgsvc.dll 5.00.2195.2939 34.27 KB (35,088 bytes) 12/7/199	19
5:29:54 PM Microsoft Corporation		7:00:00 AM Microsoft Corporation	
c:\winnt\system32\esent.dll	1/10/2002	c:\winnt\system32\msgsvc.dll	
1 , , , , , ,	2/10/2003	browser.dll 5.00.2195.2778 48.27 KB (49,424 bytes) 2/10/2003 5:29:48 PM Microsoft Corporation	
5:30:07 PM Microsoft Corporation c:\winnt\system32\ntdsatq.dll		2/10/2003 5:29:48 PM Microsoft Corporation c:\winnt\system32\browser.dll	
*	2/10/2003	alrsvc.dll 5.00.2134.1 17.77 KB (18,192 bytes) 12/7/199	9
5:30:07 PM Microsoft Corporation		7:00:00 AM Microsoft Corporation	
c:\winnt\system32\ntdsa.dll		c:\winnt\system32\alrsvc.dll	
	2/10/2003	trkwks.dll 5.00.2166.1 88.77 KB (90,896 bytes) 12/7/199)9
5:29:58 PM Microsoft Corporation		1 = 00 00 13 (
c:\winnt\system32\kdcsvc.dll		7:00:00 AM Microsoft Corporation c:\winnt\system32\trkwks.dll	

1 111 5.00.2125.1 15.77 KD (16.1441.4.)	
seclogon.dll 5.00.2135.1 15.77 KB (16,144 bytes)	scesrv.dll 5.00.2195.2780 226.27 KB (231,696 bytes) 2/10/2003
12/7/1999 7:00:00 AM Microsoft Corporation	5:30:12 PM Microsoft Corporation
c:\winnt\system32\seclogon.dll	c:\winnt\system32\scesrv.dll
psbase.dll 5.00.2195.2779 111.77 KB (114,448 bytes) 2/10/2003	umpnpmgr.dll 5.00.2182.1 86.27 KB (88,336 bytes)
5:30:10 PM Microsoft Corporation	12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\psbase.dll	c:\winnt\system32\umpnpmgr.dll
cryptsvc.dll 5.00.2181.1 61.77 KB (63,248 bytes)	services.exe 5.00.2195.2780 86.77 KB (88,848 bytes)
12/7/1999 7:00:00 AM Microsoft Corporation	12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\cryptsvc.dll	c:\winnt\system32\services.exe
cryptdll.dll5.00.2135.1 41.27 KB (42,256 bytes) 12/7/1999	msv1_0.dll5.00.2195.2900 111.77 KB (114,448 bytes) 12/7/1999
7:00:00 AM Microsoft Corporation	7:00:00 AM Microsoft Corporation
c:\winnt\system32\cryptdll.dll	c:\winnt\system32\msv1_0.dll
wkssvc.dll 5.00.2195.2780 95.27 KB (97,552 bytes) 12/7/1999	clbcatq.dll 2000.2.3471.1 496.77 KB (508,688 bytes) 2/10/2003
7:00:00 AM Microsoft Corporation	5:29:50 PM Microsoft Corporation
c:\winnt\system32\wkssvc.dll	c:\winnt\system32\clbcatq.dll
srvsvc.dll 5.00.2195.2904 79.27 KB (81,168 bytes) 12/7/1999	oleaut32.dll 2.40.4517 612.27 KB (626,960 bytes) 12/7/1999
7:00:00 AM Microsoft Corporation	7:00:00 AM Microsoft Corporation
c:\winnt\system32\srvsvc.dll	c:\winnt\system32\oleaut32.dll
cfgmgr32.dll 5.00.2134.1 16.77 KB (17,168 bytes)	cscui.dll 5.00.2195.2959 228.27 KB (233,744 bytes) 2/10/2003
12/7/1999 7:00:00 AM Microsoft Corporation	5:29:51 PM Microsoft Corporation
c:\winnt\system32\cfgmgr32.dll	c:\winnt\system32\cscui.dll
dmserver.dll 2195.2778.297.3 11.77 KB (12,048 bytes)	winspool.drv 5.00.2195.2780 109.77 KB (112,400 bytes)
2/10/2003 5:29:52 PM VERITAS Software Corp.	12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\dmserver.dll	c:\winnt\system32\winspool.drv
winsta.dll 5.00.2195.2386 36.77 KB (37,648 bytes) 2/10/2003	winscard.dll 5.00.2134.1 77.27 KB (79,120 bytes)
5:30:17 PM Microsoft Corporation	12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\winsta.dll	c:\winnt\system32\winscard.dll
lmhsvc.dll 5.00.2195.2778 9.77 KB (10,000 bytes) 12/7/1999	wlnotify.dll 5.00.2195.2780 53.77 KB (55,056 bytes)
7:00:00 AM Microsoft Corporation	2/10/2003 5:30:17 PM Microsoft Corporation
c:\winnt\system32\lmhsvc.dll	c:\winnt\system32\wlnotify.dll
dnsrslvr.dll 5.00.2195.2778 88.77 KB (90,896 bytes)	cscdll.dll 5.00.2195.2401 98.27 KB (100,624 bytes) 2/10/2003
2/10/2003 5:29:53 PM Microsoft Corporation	5:29:51 PM Microsoft Corporation
c:\winnt\system32\dnsrslvr.dll	c:\winnt\system32\cscdll.dll
tapi32.dll 5.00.2182.1 123.27 KB (126,224 bytes) 12/7/1999	lz32.dll 5.00.2134.1 9.77 KB (10,000 bytes) 12/7/1999
7:00:00 AM Microsoft Corporation	7:00:00 AM Microsoft Corporation
c:\winnt\system32\tapi32.dll rasman.dll 5.00.2195.2780 54.77 KB (56,080 bytes) 12/7/1999	c:\winnt\system32\lz32.dll version.dll 5.00.2134.1 15.77 KB (16,144 bytes) 12/7/1999
rasman.dll 5.00.2195.2780 54.77 KB (56,080 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation	version.dll 5.00.2134.1 15.77 KB (16,144 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\rasman.dll	
	c:\winnt\system32\version.dll
rasapi32.dll 5.00.2195.2671 189.77 KB (194,320 bytes)	rsaenh.dll 5.00.2195.2228 130.77 KB (133,904 bytes) 2/10/2003
rasapi32.dll 5.00.2195.2671 189.77 KB (194,320 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation	rsaenh.dll 5.00.2195.2228 130.77 KB (133,904 bytes) 2/10/2003 5:31:00 PM Microsoft Corporation 2/10/2003
rasapi32.dll 5.00.2195.2671 189.77 KB (194,320 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rasapi32.dll	rsaenh.dll 5.00.2195.2228 130.77 KB (133,904 bytes) 2/10/2003 5:31:00 PM Microsoft Corporation c:\winnt\system32\rsaenh.dll
rasapi32.dll 5.00.2195.2671 189.77 KB (194,320 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rasapi32.dll rtutils.dll 5.00.2168.1 43.77 KB (44,816 bytes) 12/7/1999	rsaenh.dll 5.00.2195.2228 130.77 KB (133,904 bytes) 2/10/2003 5:31:00 PM Microsoft Corporation c:\winnt\system32\rsaenh.dll mscat32.dll 5.131.2134.1 7.77 KB (7,952 bytes) 12/7/1999
rasapi32.dll 5.00.2195.2671 189.77 KB (194,320 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rasapi32.dll rtutils.dll 5.00.2168.1 43.77 KB (44,816 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation	rsaenh.dll 5.00.2195.2228 130.77 KB (133,904 bytes) 2/10/2003 5:31:00 PM Microsoft Corporation c:\winnt\system32\rsaenh.dll mscat32.dll 5.131.2134.1 7.77 KB (7,952 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation
rasapi32.dll 5.00.2195.2671 189.77 KB (194,320 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rasapi32.dll rtutils.dll 5.00.2168.1 43.77 KB (44,816 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rtutils.dll	rsaenh.dll 5.00.2195.2228 130.77 KB (133,904 bytes) 2/10/2003 5:31:00 PM Microsoft Corporation c:\winnt\system32\rsaenh.dll mscat32.dll 5.131.2134.1 7.77 KB (7,952 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mscat32.dll
rasapi32.dll 5.00.2195.2671 189.77 KB (194,320 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rasapi32.dll rtutils.dll 5.00.2168.1 43.77 KB (44,816 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rtutils.dll adsldpc.dll 5.00.2195.2842 127.27 KB (130,320 bytes) 2/10/2003	rsaenh.dll 5.00.2195.2228 130.77 KB (133,904 bytes) 2/10/2003 5:31:00 PM Microsoft Corporation c:\winnt\system32\rsaenh.dll mscat32.dll 5.131.2134.1 7.77 KB (7,952 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mscat32.dll ole32.dll 5.00.2195.2887 969.77 KB (993,040 bytes) 2/10/2003
rasapi32.dll 5.00.2195.2671 189.77 KB (194,320 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rasapi32.dll rtutils.dll 5.00.2168.1 43.77 KB (44,816 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rtutils.dll adsldpc.dll 5.00.2195.2842 127.27 KB (130,320 bytes) 2/10/2003 5:29:47 PM Microsoft Corporation	rsaenh.dll 5.00.2195.2228 130.77 KB (133,904 bytes) 2/10/2003 5:31:00 PM Microsoft Corporation c:\winnt\system32\rsaenh.dll mscat32.dll 5.131.2134.1 7.77 KB (7,952 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mscat32.dll ole32.dll 5.00.2195.2887 969.77 KB (993,040 bytes) 2/10/2003 5:30:09 PM Microsoft Corporation
rasapi32.dll 5.00.2195.2671 189.77 KB (194,320 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rasapi32.dll rtutils.dll 5.00.2168.1 43.77 KB (44,816 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rtutils.dll adsldpc.dll 5.00.2195.2842 127.27 KB (130,320 bytes) 2/10/2003 5:29:47 PM Microsoft Corporation c:\winnt\system32\adsldpc.dll	rsaenh.dll 5.00.2195.2228 130.77 KB (133,904 bytes) 2/10/2003 5:31:00 PM Microsoft Corporation c:\winnt\system32\rsaenh.dll mscat32.dll 5.131.2134.1 7.77 KB (7,952 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mscat32.dll ole32.dll 5.00.2195.2887 969.77 KB (993,040 bytes) 2/10/2003 5:30:09 PM Microsoft Corporation c:\winnt\system32\ole32.dll
rasapi32.dll 5.00.2195.2671 189.77 KB (194,320 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rasapi32.dll rtutils.dll 5.00.2168.1 43.77 KB (44,816 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rtutils.dll adsldpc.dll 5.00.2195.2842 127.27 KB (130,320 bytes) 2/10/2003 5:29:47 PM Microsoft Corporation c:\winnt\system32\adsldpc.dll activeds.dll 5.00.2195.2778 174.77 KB (178,960 bytes)	rsaenh.dll 5.00.2195.2228 130.77 KB (133,904 bytes) 2/10/2003 5:31:00 PM Microsoft Corporation c:\winnt\system32\rsaenh.dll mscat32.dll 5.131.2134.1 7.77 KB (7,952 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mscat32.dll ole32.dll 5.00.2195.2887 969.77 KB (993,040 bytes) 2/10/2003 5:30:09 PM Microsoft Corporation
rasapi32.dll 5.00.2195.2671 189.77 KB (194,320 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rasapi32.dll rtutils.dll 5.00.2168.1 43.77 KB (44,816 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rtutils.dll adsldpc.dll 5.00.2195.2842 127.27 KB (130,320 bytes) 2/10/2003 5:29:47 PM Microsoft Corporation c:\winnt\system32\adsldpc.dll activeds.dll 5.00.2195.2778 174.77 KB (178,960 bytes) 2/10/2003 5:29:41 PM Microsoft Corporation	rsaenh.dll 5.00.2195.2228 130.77 KB (133,904 bytes) 2/10/2003 5:31:00 PM Microsoft Corporation c:\winnt\system32\rsaenh.dll
rasapi32.dll 5.00.2195.2671 189.77 KB (194,320 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rasapi32.dll rtutils.dll 5.00.2168.1 43.77 KB (44,816 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rtutils.dll adsldpc.dll 5.00.2195.2842 127.27 KB (130,320 bytes) 2/10/2003 5:29:47 PM Microsoft Corporation c:\winnt\system32\adsldpc.dll activeds.dll 5.00.2195.2778 174.77 KB (178,960 bytes)	rsaenh.dll 5.00.2195.2228 130.77 KB (133,904 bytes) 2/10/2003 5:31:00 PM Microsoft Corporation c:\winnt\system32\rsaenh.dll mscat32.dll 5.131.2134.1 7.77 KB (7,952 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mscat32.dll ole32.dll 5.00.2195.2887 969.77 KB (993,040 bytes) 2/10/2003 5:30:09 PM Microsoft Corporation c:\winnt\system32\classed 2.dll imagehlp.dll 5.00.2195.2778 125.77 KB (128,784 bytes) 5/4/2001 1:05:02 PM Microsoft Corporation c:\winnt\system32\imagehlp.dll
rasapi32.dll 5.00.2195.2671 189.77 KB (194,320 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rasapi32.dll rtutils.dll 5.00.2168.1 43.77 KB (44,816 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rtutils.dll adsldpc.dll 5.00.2195.2842 127.27 KB (130,320 bytes) 2/10/2003 5:29:47 PM Microsoft Corporation c:\winnt\system32\adsldpc.dll activeds.dll 5.00.2195.2778 174.77 KB (178,960 bytes) 2/10/2003 5:29:41 PM Microsoft Corporation c:\winnt\system32\adsldpc.dll	rsaenh.dll 5.00.2195.2228 130.77 KB (133,904 bytes) 2/10/2003 5:31:00 PM Microsoft Corporation c:\winnt\system32\rsaenh.dll mscat32.dll 5.131.2134.1 7.77 KB (7,952 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mscat32.dll ole32.dll 5.00.2195.2887 969.77 KB (993,040 bytes) 2/10/2003 5:30:09 PM Microsoft Corporation c:\winnt\system32\ole32.dll imagehlp.dll 5.00.2195.2778 125.77 KB (128,784 bytes) 5/4/2001 1:05:02 PM Microsoft Corporation c:\winnt\system32\ole32\magehlp.dll msasn1.dll 5.00.2134.1 51.27 KB (52,496 bytes) 12/7/1999
rasapi32.dll 5.00.2195.2671 189.77 KB (194,320 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rasapi32.dll rtutils.dll 5.00.2168.1 43.77 KB (44,816 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rtutils.dll adsldpc.dll 5.00.2195.2842 127.27 KB (130,320 bytes) 2/10/2003 5:29:47 PM Microsoft Corporation c:\winnt\system32\adsldpc.dll activeds.dll 5.00.2195.2778 174.77 KB (178,960 bytes) 2/10/2003 5:29:41 PM Microsoft Corporation c:\winnt\system32\activeds.dll mprapi.dll 5.00.2181.1 79.27 KB (81,168 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation	rsaenh.dll 5.00.2195.2228 130.77 KB (133,904 bytes) 2/10/2003 5:31:00 PM Microsoft Corporation c:\winnt\system32\rsaenh.dll mscat32.dll 5.131.2134.1 7.77 KB (7,952 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mscat32.dll ole32.dll 5.00.2195.2887 969.77 KB (993,040 bytes) 2/10/2003 5:30:09 PM Microsoft Corporation c:\winnt\system32\ole32.dll imagehlp.dll 5.00.2195.2778 125.77 KB (128,784 bytes) 5/4/2001 1:05:02 PM Microsoft Corporation c:\winnt\system32\ole32\magehlp.dll msasn1.dll 5.00.2134.1 51.27 KB (52,496 bytes) 12/7/1999
rasapi32.dll 5.00.2195.2671 189.77 KB (194,320 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rasapi32.dll rtutils.dll 5.00.2168.1 43.77 KB (44,816 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rtutils.dll adsldpc.dll 5.00.2195.2842 127.27 KB (130,320 bytes) 2/10/2003 5:29:47 PM Microsoft Corporation c:\winnt\system32\adsldpc.dll activeds.dll 5.00.2195.2778 174.77 KB (178,960 bytes) 2/10/2003 5:29:41 PM Microsoft Corporation c:\winnt\system32\activeds.dll mprapi.dll 5.00.2181.1 79.27 KB (81,168 bytes) 7:00:00 AM Microsoft Corporation c:\winnt\system32\activeds.dll mprapi.dll 5.00.2181.1 79.27 KB (81,168 bytes) 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll	rsaenh.dll 5.00.2195.2228 130.77 KB (133,904 bytes) 2/10/2003 5:31:00 PM Microsoft Corporation c:\winnt\system32\rsaenh.dll mscat32.dll 5.131.2134.1 7.77 KB (7,952 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mscat32.dll ole32.dll 5.00.2195.2887 969.77 KB (993,040 bytes) 2/10/2003 5:30:09 PM Microsoft Corporation c:\winnt\system32\lole32.dll imagehlp.dll 5.00.2195.2778 125.77 KB (128,784 bytes) 5/4/2001 1:05:02 PM Microsoft Corporation c:\winnt\system32\mscaf4 Corporation
rasapi32.dll 5.00.2195.2671 189.77 KB (194,320 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rasapi32.dll rtutils.dll 5.00.2168.1 43.77 KB (44,816 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rtutils.dll adsldpc.dll 5.00.2195.2842 127.27 KB (130,320 bytes) 2/10/2003 5:29:47 PM Microsoft Corporation c:\winnt\system32\adsldpc.dll activeds.dll 5.00.2195.2778 174.77 KB (178,960 bytes) 2/10/2003 5:29:41 PM Microsoft Corporation c:\winnt\system32\activeds.dll mprapi.dll 5.00.2181.1 79.27 KB (81,168 bytes) 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll	rsaenh.dll 5.00.2195.2228 130.77 KB (133,904 bytes) 2/10/2003 5:31:00 PM Microsoft Corporation c:\winnt\system32\rsaenh.dll mscat32.dll 5.131.2134.1 7.77 KB (7,952 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mscat32.dll ole32.dll 5.00.2195.2887 969.77 KB (993,040 bytes) 2/10/2003 5:30:09 PM Microsoft Corporation c:\winnt\system32\ole32.dll imagehlp.dll 5.00.2195.2778 125.77 KB (128,784 bytes) 5/4/2001 1:05:02 PM Microsoft Corporation c:\winnt\system32\winagehlp.dll msasn1.dll 5.00.2134.1 51.27 KB (52,496 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\msasn1.dll crypt32.dll 5.131.2195.2833 451.27 KB (462,096 bytes) 2/10/2003
rasapi32.dll 5.00.2195.2671 189.77 KB (194,320 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rasapi32.dll rtutils.dll 5.00.2168.1 43.77 KB (44,816 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rtutils.dll adsldpc.dll 5.00.2195.2842 127.27 KB (130,320 bytes) 2/10/2003 5:29:47 PM Microsoft Corporation c:\winnt\system32\adsldpc.dll activeds.dll 5.00.2195.2778 174.77 KB (178,960 bytes) 2/10/2003 5:29:41 PM Microsoft Corporation c:\winnt\system32\adsldpc.dll mprapi.dll 5.00.2181.1 79.27 KB (81,168 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll iphlpapi.dll 5.00.2173.2 67.77 KB (69,392 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation	rsaenh.dll 5.00.2195.2228 130.77 KB (133,904 bytes) 2/10/2003 5:31:00 PM Microsoft Corporation c:\winnt\system32\rsaenh.dll mscat32.dll 5.131.2134.1 7.77 KB (7,952 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mscat32.dll ole32.dll 5.00.2195.2887 969.77 KB (993,040 bytes) 2/10/2003 5:30:09 PM Microsoft Corporation c:\winnt\system32\ole32.dll imagehlp.dll 5.00.2195.2778 125.77 KB (128,784 bytes) 5/4/2001 1:05:02 PM Microsoft Corporation c:\winnt\system32\imagehlp.dll msasn1.dll 5.00.2134.1 51.27 KB (52,496 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\imagehlp.dll crypt32.dll 5.131.2195.2833 451.27 KB (462,096 bytes) 2/10/2003 5:29:51 PM Microsoft Corporation
rasapi32.dll 5.00.2195.2671 189.77 KB (194,320 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rasapi32.dll rtutils.dll 5.00.2168.1 43.77 KB (44,816 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rtutils.dll adsldpc.dll 5.00.2195.2842 127.27 KB (130,320 bytes) 2/10/2003 5:29:47 PM Microsoft Corporation c:\winnt\system32\adsldpc.dll activeds.dll 5.00.2195.2778 174.77 KB (178,960 bytes) 2/10/2003 5:29:41 PM Microsoft Corporation c:\winnt\system32\adsldpc.dll mprapi.dll 5.00.2181.1 79.27 KB (81,168 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll iphlpapi.dll 5.00.2173.2 67.77 KB (69,392 bytes)	rsaenh.dll 5.00.2195.2228 130.77 KB (133,904 bytes) 2/10/2003 5:31:00 PM Microsoft Corporation c:\winnt\system32\rsaenh.dll mscat32.dll 5.131.2134.1 7.77 KB (7,952 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mscat32.dll ole32.dll 5.00.2195.2887 969.77 KB (993,040 bytes) 2/10/2003 5:30:09 PM Microsoft Corporation c:\winnt\system32\ole32.dll imagehlp.dll 5.00.2195.2778 125.77 KB (128,784 bytes) 5/4/2001 1:05:02 PM Microsoft Corporation c:\winnt\system32\winagehlp.dll msasn1.dll 5.00.2134.1 51.27 KB (52,496 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\msasn1.dll crypt32.dll 5.131.2195.2833 451.27 KB (462,096 bytes) 2/10/2003
rasapi32.dll 5.00.2195.2671 189.77 KB (194,320 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rasapi32.dll rtutils.dll 5.00.2168.1 43.77 KB (44,816 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rtutils.dll adsldpc.dll 5.00.2195.2842 127.27 KB (130,320 bytes) 2/10/2003 5:29:47 PM Microsoft Corporation c:\winnt\system32\adsldpc.dll activeds.dll 5.00.2195.2778 174.77 KB (178,960 bytes) 2/10/2003 5:29:41 PM Microsoft Corporation c:\winnt\system32\activeds.dll mprapi.dll 5.00.2181.1 79.27 KB (81,168 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll iphlpapi.dll 5.00.2173.2 67.77 KB (69,392 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll iphlpapi.dll 5.00.2173.2 67.77 KB (69,392 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll	rsaenh.dll 5.00.2195.2228 130.77 KB (133,904 bytes) 2/10/2003 5:31:00 PM Microsoft Corporation c:\winnt\system32\rsaenh.dll mscat32.dll 5.131.2134.1 7.77 KB (7,952 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rsaenh.sat32.dll ole32.dll 5.00.2195.2887 969.77 KB (993,040 bytes) 2/10/2003 5:30:09 PM Microsoft Corporation c:\winnt\system32\ole32.dll imagehlp.dll 5.00.2195.2778 125.77 KB (128,784 bytes) 5/4/2001 1:05:02 PM Microsoft Corporation c:\winnt\system32\imagehlp.dll msasn1.dll 5.00.2134.1 51.27 KB (52,496 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\imagehlp.dll crypt32.dll 5.131.2195.2833 451.27 KB (462,096 bytes) 2/10/2003 5:29:51 PM Microsoft Corporation c:\winnt\system32\crypt32.dll
rasapi32.dll 5.00.2195.2671 189.77 KB (194,320 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rasapi32.dll rtutils.dll 5.00.2168.1 43.77 KB (44,816 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rtutils.dll adsldpc.dll 5.00.2195.2842 127.27 KB (130,320 bytes) 2/10/2003 5:29:47 PM Microsoft Corporation c:\winnt\system32\adsldpc.dll activeds.dll 5.00.2195.2778 174.77 KB (178,960 bytes) 2/10/2003 5:29:41 PM Microsoft Corporation c:\winnt\system32\activeds.dll mprapi.dll 5.00.2181.1 79.27 KB (81,168 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll iphlpapi.dll 5.00.2173.2 67.77 KB (69,392 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll iphlpapi.dll 5.00.2134.1 7.27 KB (7,440 bytes) 12/7/1999 7:00:00 AM	rsaenh.dll 5.00.2195.2228 130.77 KB (133,904 bytes) 2/10/2003 5:31:00 PM Microsoft Corporation c:\winnt\system32\rsaenh.dll mscat32.dll 5.131.2134.1 7.77 KB (7,952 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rsaenhscat32.dll ole32.dll 5.00.2195.2887 969.77 KB (993,040 bytes) 2/10/2003 5:30:09 PM Microsoft Corporation c:\winnt\system32\ole32.dll imagehlp.dll 5.00.2195.2778 125.77 KB (128,784 bytes) 5/4/2001 1:05:02 PM Microsoft Corporation c:\winnt\system32\imagehlp.dll msasn1.dll 5.00.2134.1 51.27 KB (52,496 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\imagehlp.dll crypt32.dll 5.131.2195.2833 451.27 KB (462,096 bytes) 2/10/2003 5:29:51 PM Microsoft Corporation c:\winnt\system32\crypt32.dll wintrust.dll 5.131.2195.2779 162.27 KB (166,160 bytes)
rasapi32.dll 5.00.2195.2671 189.77 KB (194,320 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rasapi32.dll rtutils.dll 5.00.2168.1 43.77 KB (44,816 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rtutils.dll adsldpc.dll 5.00.2195.2842 127.27 KB (130,320 bytes) 2/10/2003 5:29:47 PM Microsoft Corporation c:\winnt\system32\adsldpc.dll activeds.dll 5.00.2195.2778 174.77 KB (178,960 bytes) 2/10/2003 5:29:41 PM Microsoft Corporation c:\winnt\system32\activeds.dll mprapi.dll 5.00.2181.1 79.27 KB (81,168 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll iphlpapi.dll 5.00.2173.2 67.77 KB (69,392 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll icmp.dll 5.00.2134.1 7.27 KB (7,440 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll icmp.dll 5.00.2134.1 7.27 KB (7,440 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation	rsaenh.dll 5.00.2195.2228 130.77 KB (133,904 bytes) 2/10/2003 5:31:00 PM Microsoft Corporation c:\winnt\system32\rsaenh.dll mscat32.dll 5.131.2134.1 7.77 KB (7,952 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rsaents22.dll ole32.dll 5.00.2195.2887 969.77 KB (993,040 bytes) 2/10/2003 5:30:09 PM Microsoft Corporation c:\winnt\system32\ole32.dll imagehlp.dll 5.00.2195.2778 125.77 KB (128,784 bytes) 5/4/2001 1:05:02 PM Microsoft Corporation c:\winnt\system32\ole32.dll msasn1.dll 5.00.2134.1 51.27 KB (52,496 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\msasn1.dll crypt32.dll 5.131.2195.2833 451.27 KB (462,096 bytes) 2/10/2003 5:29:51 PM Microsoft Corporation c:\winnt\system32\rsaents2\climagehlp.dll wintrust.dll 5.131.2195.2779 162.27 KB (166,160 bytes) 2/10/2003 5:30:17 PM Microsoft Corporation
rasapi32.dll 5.00.2195.2671 189.77 KB (194,320 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rasapi32.dll rtutils.dll 5.00.2168.1 43.77 KB (44,816 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rtutils.dll adsldpc.dll 5.00.2195.2842 127.27 KB (130,320 bytes) 2/10/2003 5:29:47 PM Microsoft Corporation c:\winnt\system32\adsldpc.dll activeds.dll 5.00.2195.2778 174.77 KB (178,960 bytes) 2/10/2003 5:29:41 PM Microsoft Corporation c:\winnt\system32\activeds.dll mprapi.dll 5.00.2181.1 79.27 KB (81,168 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll iphlapi.dll 5.00.2173.2 67.77 KB (69,392 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll icmp.dll 5.00.2134.1 7.27 KB (7,440 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll icmp.dll 5.00.2134.1 7.27 KB (7,440 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll icmp.dll 5.00.2134.1 7.27 KB (90,896 bytes)	rsaenh.dll 5.00.2195.2228 130.77 KB (133,904 bytes) 2/10/2003 5:31:00 PM Microsoft Corporation c:\winnt\system32\rsaenh.dll mscat32.dll 5.131.2134.1 7.77 KB (7,952 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mscat32.dll ole32.dll 5.00.2195.2887 969.77 KB (993,040 bytes) 2/10/2003 5:30:09 PM Microsoft Corporation c:\winnt\system32\classed 2.dll imagehlp.dll 5.00.2195.2778 125.77 KB (128,784 bytes) 5/4/2001 1:05:02 PM Microsoft Corporation c:\winnt\system32\magehlp.dll msasn1.dll 5.00.2134.1 51.27 KB (52,496 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\msasn1.dll crypt32.dll 5.131.2195.2833 451.27 KB (462,096 bytes) 2/10/2003 5:29:51 PM Microsoft Corporation c:\winnt\system32\crypt32.dll wintrust.dll 5.131.2195.2779 162.27 KB (166,160 bytes) 2/10/2003 5:30:17 PM Microsoft Corporation c:\winnt\system32\wintrust.dll
rasapi32.dll 5.00.2195.2671 189.77 KB (194,320 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rasapi32.dll rtutils.dll 5.00.2168.1 43.77 KB (44,816 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rtutils.dll adsldpc.dll 5.00.2195.2842 127.27 KB (130,320 bytes) 2/10/2003 5:29:47 PM Microsoft Corporation c:\winnt\system32\adsldpc.dll activeds.dll 5.00.2195.2778 174.77 KB (178,960 bytes) 2/10/2003 5:29:41 PM Microsoft Corporation c:\winnt\system32\activeds.dll mprapi.dll 5.00.2181.1 79.27 KB (81,168 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll iphlpapi.dll 5.00.2173.2 67.77 KB (69,392 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll icmp.dll 5.00.2134.1 7.27 KB (7,440 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll icmp.dll 5.00.2134.1 7.27 KB (7,440 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll icmp.dll 5.00.2195.2778 88.77 KB (90,896 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll c:\winnt\system32\mprapi.dll c:\winnt\system32\mprapi.dll c:\winnt\system32\mprapi.dll c:\winnt\system32\mprapi.dll icmp.dll 5.00.2134.1 7.27 KB (7,440 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll c:\winnt\system32\mprapi.dll c:\winnt\system32\mprapi.dll c:\winnt\system32\mprapi.dll	rsaenh.dll 5.00.2195.2228 130.77 KB (133,904 bytes) 2/10/2003 5:31:00 PM Microsoft Corporation c:\winnt\system32\rsaenh.dll mscat32.dll 5.131.2134.1 7.77 KB (7,952 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mscat32.dll ole32.dll 5.00.2195.2887 969.77 KB (993,040 bytes) 2/10/2003 5:30:09 PM Microsoft Corporation c:\winnt\system32\ole32.dll imagehlp.dll 5.00.2195.2778 125.77 KB (128,784 bytes) 5/4/2001 1:05:02 PM Microsoft Corporation c:\winnt\system32\ole32\simagehlp.dll msasn1.dll 5.00.2134.1 51.27 KB (52,496 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\ole32\simagehlp.dll crypt32.dll 5.131.2195.2833 451.27 KB (462,096 bytes) 2/10/2003 5:29:51 PM Microsoft Corporation c:\winnt\system32\crypt32.dll wintrust.dll 5.131.2195.2779 162.27 KB (166,160 bytes) 2/10/2003 5:30:17 PM Microsoft Corporation c:\winnt\system32\crypt32.wintrust.dll setupapi.dll 5.00.2195.2663 555.77 KB (569,104 bytes)
rasapi32.dll 5.00.2195.2671 189.77 KB (194,320 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rasapi32.dll rtutils.dll 5.00.2168.1 43.77 KB (44,816 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rtutils.dll adsldpc.dll 5.00.2195.2842 127.27 KB (130,320 bytes) 2/10/2003 5:29:47 PM Microsoft Corporation c:\winnt\system32\adsldpc.dll activeds.dll 5.00.2195.2778 174.77 KB (178,960 bytes) 2/10/2003 5:29:41 PM Microsoft Corporation c:\winnt\system32\activeds.dll mprapi.dll 5.00.2181.1 79.27 KB (81,168 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll iphlpapi.dll 5.00.2173.2 67.77 KB (69,392 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll icmp.dll 5.00.2134.1 7.27 KB (7,440 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll dhcpesvc.dll 5.00.2195.2778 88.77 KB (90,896 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll dhcpesvc.dll 5.00.2195.2778 88.77 KB (90,896 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll dhcpesvc.dll 5.00.2195.2778 88.77 KB (90,896 bytes)	rsaenh.dll 5.00.2195.2228 130.77 KB (133,904 bytes) 2/10/2003 5:31:00 PM Microsoft Corporation c:\winnt\system32\rsaenh.dll mscat32.dll 5.131.2134.1 7.77 KB (7,952 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mscat32.dll ole32.dll 5.00.2195.2887 969.77 KB (993,040 bytes) 2/10/2003 5:30:09 PM Microsoft Corporation c:\winnt\system32\ole32.dll imagehlp.dll 5.00.2195.2778 125.77 KB (128,784 bytes) 5/4/2001 1:05:02 PM Microsoft Corporation c:\winnt\system32\ole32\ole32.dll msasn1.dll 5.00.2134.1 51.27 KB (52,496 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\ole32\ole32.dlls.131.2195.2833 451.27 KB (462,096 bytes) 2/10/2003 5:29:51 PM Microsoft Corporation c:\winnt\system32\crypt32.dll wintrust.dll 5.131.2195.2779 162.27 KB (166,160 bytes) 2/10/2003 5:30:17 PM Microsoft Corporation c:\winnt\system32\crypt32.dll setupapi.dll 5.00.2195.2663 555.77 KB (569,104 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation
rasapi32.dll 5.00.2195.2671 189.77 KB (194,320 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rasapi32.dll rtutils.dll 5.00.2168.1 43.77 KB (44,816 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rtutils.dll adsldpc.dll 5.00.2195.2842 127.27 KB (130,320 bytes) 2/10/2003 5:29:47 PM Microsoft Corporation c:\winnt\system32\adsldpc.dll activeds.dll 5.00.2195.2778 174.77 KB (178,960 bytes) 2/10/2003 5:29:41 PM Microsoft Corporation c:\winnt\system32\activeds.dll mprapi.dll 5.00.2181.1 79.27 KB (81,168 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll iphlpapi.dll 5.00.2173.2 67.77 KB (69,392 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll icmp.dll 5.00.2134.1 7.27 KB (7,440 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll icmp.dll 5.00.2134.1 7.27 KB (7,440 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll icmp.dll 5.00.2195.2778 88.77 KB (90,896 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll c:\winnt\system32\mprapi.dll c:\winnt\system32\mprapi.dll c:\winnt\system32\mprapi.dll c:\winnt\system32\mprapi.dll icmp.dll 5.00.2134.1 7.27 KB (7,440 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll c:\winnt\system32\mprapi.dll c:\winnt\system32\mprapi.dll c:\winnt\system32\mprapi.dll	rsaenh.dll 5.00.2195.2228 130.77 KB (133,904 bytes) 2/10/2003 5:31:00 PM Microsoft Corporation c:\winnt\system32\rsaenh.dll mscat32.dll 5.131.2134.1 7.77 KB (7,952 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mscat32.dll ole32.dll 5.00.2195.2887 969.77 KB (993,040 bytes) 2/10/2003 5:30:09 PM Microsoft Corporation c:\winnt\system32\ole32.dll imagehlp.dll 5.00.2195.278 125.77 KB (128,784 bytes) 5/4/2001 1:05:02 PM Microsoft Corporation c:\winnt\system32\winagehlp.dll msasn1.dll 5.00.2134.1 51.27 KB (52,496 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\winagehlp.dll crypt32.dll 5.131.2195.2833 451.27 KB (462,096 bytes) 2/10/2003 5:29:51 PM Microsoft Corporation c:\winnt\system32\winagehlp.dll 5.131.2195.2779 162.27 KB (166,160 bytes) 2/10/2003 5:30:17 PM Microsoft Corporation c:\winnt\system32\wintrust.dll setupapi.dll 5.00.2195.2663 555.77 KB (59,104 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\wintrust.dll setupapi.dll 5.00.2195.2663 555.77 KB (569,104 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\setupapi.dll wintrust.dll setupapi.dll 5.00.2161.1 184.77 KB (189,200 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation
rasapi32.dll 5.00.2195.2671 189.77 KB (194,320 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rasapi32.dll rtutils.dll 5.00.2168.1 43.77 KB (44,816 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rtutils.dll adsldpc.dll 5.00.2195.2842 127.27 KB (130,320 bytes) 2/10/2003 5:29:47 PM Microsoft Corporation c:\winnt\system32\adsldpc.dll activeds.dll 5.00.2195.2778 174.77 KB (178,960 bytes) 2/10/2003 5:29:41 PM Microsoft Corporation c:\winnt\system32\activeds.dll mprapi.dll 5.00.2181.1 79.27 KB (81,168 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll iphlpapi.dll 5.00.2173.2 67.77 KB (69,392 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll icmp.dll 5.00.2134.1 7.27 KB (7,440 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll icmp.dll 5.00.2134.1 7.27 KB (7,440 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll icmp.dll 5.00.2195.2778 88.77 KB (90,896 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll eventlog.dll 5.00.2178.1 43.77 KB (44,816 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation	rsaenh.dll 5.00.2195.2228 130.77 KB (133,904 bytes) 2/10/2003 5:31:00 PM Microsoft Corporation c:\winnt\system32\rsaenh.dll mscat32.dll 5.131.2134.1 7.77 KB (7,952 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mscat32.dll ole32.dll 5.00.2195.2887 969.77 KB (993,040 bytes) 2/10/2003 5:30:09 PM Microsoft Corporation c:\winnt\system32\ole32.dll imagehlp.dll 5.00.2195.2778 125.77 KB (128,784 bytes) 5/4/2001 1:05:02 PM Microsoft Corporation c:\winnt\system32\ole32\magehlp.dll msasn1.dll 5.00.2134.1 51.27 KB (52,496 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\msasn1.dll crypt32.dll 5.131.2195.2833 451.27 KB (462,096 bytes) 2/10/2003 5:29:51 PM Microsoft Corporation c:\winnt\system32\crypt32.dll wintrust.dll 5.131.2195.2779 162.27 KB (166,160 bytes) 2/10/2003 5:30:17 PM Microsoft Corporation c:\winnt\system32\wintrust.dll setupapi.dll 5.00.2195.2663 555.77 KB (569,104 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\system32\crypt32.dll wintrust.dll 5.00.2195.2663 555.77 KB (569,104 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\setupapi.dll winnt\system32\setupapi.dll winnt\system32\setupapi.dll winntm\system32\setupapi.dll winntm\system32\setupapi.dll winntm\system32\setupapi.dll winntm.dll 5.00.2161.1 184.77 KB (189,200 bytes) 12/7/1999
rasapi32.dll 5.00.2195.2671 189.77 KB (194,320 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rasapi32.dll rtutils.dll 5.00.2168.1 43.77 KB (44,816 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rtutils.dll adsldpc.dll 5.00.2195.2842 127.27 KB (130,320 bytes) 2/10/2003 5:29:47 PM Microsoft Corporation c:\winnt\system32\adsldpc.dll activeds.dll 5.00.2195.2778 174.77 KB (178,960 bytes) 2/10/2003 5:29:41 PM Microsoft Corporation c:\winnt\system32\activeds.dll mprapi.dll 5.00.2181.1 79.27 KB (81,168 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll iphlpapi.dll 5.00.2173.2 67.77 KB (69,392 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll icmp.dll 5.00.2134.1 7.27 KB (7,440 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll icmp.dll 5.00.2134.1 7.27 KB (7,440 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll icmp.dll 5.00.2195.2778 88.77 KB (90,896 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll eventlog.dll 5.00.2178.1 43.77 KB (44,816 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\eventlog.dll	rsaenh.dll 5.00.2195.2228 130.77 KB (133,904 bytes) 2/10/2003 5:31:00 PM Microsoft Corporation c:\winnt\system32\rsaenh.dll mscat32.dll 5.131.2134.1 7.77 KB (7,952 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mscat32.dll ole32.dll 5.00.2195.2887 969.77 KB (993,040 bytes) 2/10/2003 5:30:09 PM Microsoft Corporation c:\winnt\system32\ole32.dll imagehlp.dll 5.00.2195.278 125.77 KB (128,784 bytes) 5/4/2001 1:05:02 PM Microsoft Corporation c:\winnt\system32\winagehlp.dll msasn1.dll 5.00.2134.1 51.27 KB (52,496 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\winagehlp.dll crypt32.dll 5.131.2195.2833 451.27 KB (462,096 bytes) 2/10/2003 5:29:51 PM Microsoft Corporation c:\winnt\system32\winagehlp.dll 5.131.2195.2779 162.27 KB (166,160 bytes) 2/10/2003 5:30:17 PM Microsoft Corporation c:\winnt\system32\wintrust.dll setupapi.dll 5.00.2195.2663 555.77 KB (59,104 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\wintrust.dll setupapi.dll 5.00.2195.2663 555.77 KB (569,104 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\setupapi.dll wintrust.dll setupapi.dll 5.00.2161.1 184.77 KB (189,200 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation
rasapi32.dll 5.00.2195.2671 189.77 KB (194,320 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rasapi32.dll rtutils.dll 5.00.2168.1 43.77 KB (44,816 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rtutils.dll adsldpc.dll 5.00.2195.2842 127.27 KB (130,320 bytes) 2/10/2003 5:29:47 PM Microsoft Corporation c:\winnt\system32\adsldpc.dll activeds.dll 5.00.2195.2778 174.77 KB (178,960 bytes) 2/10/2003 5:29:41 PM Microsoft Corporation c:\winnt\system32\activeds.dll mprapi.dll 5.00.2181.1 79.27 KB (81,168 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll iphlpapi.dll 5.00.2173.2 67.77 KB (69,392 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll icmp.dll 5.00.2134.1 7.27 KB (7,440 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll icmp.dll 5.00.2134.1 7.27 KB (7,440 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll icmp.dll 5.00.2134.1 7.27 KB (90,896 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mprapi.dll eventlog.dll 5.00.2178.1 43.77 KB (44,816 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\eventlog.dll ntdsapi.dll 5.00.2195.2661 55.77 KB (57,104 bytes) 2/10/2003	rsaenh.dll 5.00.2195.2228 130.77 KB (133,904 bytes) 2/10/2003 5:31:00 PM Microsoft Corporation c:\winnt\system32\rsaenh.dll mscat32.dll 5.131.2134.1 7.77 KB (7,952 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\mscat32.dll ole32.dll 5.00.2195.2887 969.77 KB (993,040 bytes) 2/10/2003 5:30:09 PM Microsoft Corporation c:\winnt\system32\ole32.dll imagehlp.dll 5.00.2195.2778 125.77 KB (128,784 bytes) 5/4/2001 1:05:02 PM Microsoft Corporation c:\winnt\system32\winagehlp.dll msasn1.dll 5.00.2134.1 51.27 KB (52,496 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\msasn1.dll crypt32.dll 5.131.2195.2833 451.27 KB (462,096 bytes) 2/10/2003 5:29:51 PM Microsoft Corporation c:\winnt\system32\crypt32.dll wintrust.dll 5.131.2195.2779 162.27 KB (166,160 bytes) 2/10/2003 5:30:17 PM Microsoft Corporation c:\winnt\system32\wintrust.dll setupapi.dll 5.00.2195.2663 555.77 KB (569,104 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\crypt32.dll wintrysystem32\crypt32.dll wintrysystem32\crypt32.dll setupapi.dll 5.00.2195.2663 555.77 KB (569,104 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\crypt32.dll winnm.dll 5.00.2161.1 184.77 KB (189,200 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation

comctl32.dll 5.81 537.77 KB (550,672 bytes) 12/7/1999	msvcrt.dll 6.10.8924.0 284.05 KB (290,869 bytes) 5/4/2001
7:00:00 AM Microsoft Corporation	1:05:02 PM Microsoft Corporation
c:\winnt\system32\comctl32.dll	c:\winnt\system32\msvcrt.dll
shlwapi.dll5.00.3315.1000 282.77 KB (289,552 bytes) 2/10/2003	winlogon.exe 5.00.2195.2953 173.77 KB (177,936 bytes)
5:30:13 PM Microsoft Corporation	12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\shlwapi.dll	c:\winnt\system32\winlogon.exe
shell32.dll 5.00.3315.2902 2.25 MB (2,359,056 bytes) 2/10/2003	sfcfiles.dll 5.00.2195.2967 948.27 KB (971,024 bytes) 2/10/2003
5:30:13 PM Microsoft Corporation	5:30:12 PM Microsoft Corporation
c:\winnt\system32\shell32.dll	c:\winnt\system32\sfcfiles.dll
msgina.dll 5.00.2195.2779 324.27 KB (332,048 bytes) 12/7/1999	ntdll.dll 5.00.2195.2779 478.77 KB (490,256 bytes) 5/4/2001
7:00:00 AM Microsoft Corporation	1:05:02 PM Microsoft Corporation
c:\winnt\system32\msgina.dll	c:\winnt\system32\ntdll.dll
wsock32.dll 5.00.2195.2871 21.27 KB (21,776 bytes)	smss.exe 5.00.2195.2901 44.27 KB (45,328 bytes) 12/7/1999
2/10/2003 5:30:17 PM Microsoft Corporation	7:00:00 AM Microsoft Corporation
c:\winnt\system32\wsock32.dll	c:\winnt\system32\smss.exe
dnsapi.dll 5.00.2195.2785 130.77 KB (133,904 bytes) 2/10/2003 5:29:53 PM Microsoft Corporation 2/10/2003	[Carvinga]
5:29:53 PM Microsoft Corporation c:\winnt\system32\dnsapi.dll	[Services]
wldap32.dll 5.00.2195.2797 125.27 KB (128,272 bytes)	Display Name Name State Start ModeService Type
2/10/2003 5:30:17 PM Microsoft Corporation	Path Error Control Start NameTag ID
c:\winnt\system32\wldap32.dll	Alerter Alerter Running Auto Share Process
ws2help.dll 5.00.2134.1 17.77 KB (18,192 bytes)	c:\winnt\system32\services.exe Normal LocalSystem 0
12/7/1999 7:00:00 AM Microsoft Corporation	Application Management AppMgmt Stopped Manual Share
c:\winnt\system32\ws2help.dll	Process c:\winnt\system32\services.exe Normal LocalSystem 0
ws2 32.dll5.00.2195.2780 67.77 KB (69,392 bytes) 2/10/2003	Computer Browser Browser Running Auto Share Process
5:30:17 PM Microsoft Corporation	c:\winnt\system32\services.exe Normal LocalSystem 0
c:\winnt\system32\ws2 32.dll	Indexing Service cisve Stopped Manual Share Process
samlib.dll 5.00.2195.2780 49.77 KB (50,960 bytes) 12/7/1999	c:\winnt\system32\cisvc.exe Normal LocalSystem 0
7:00:00 AM Microsoft Corporation	ClipBook ClipSrv Stopped Manual Own Process
c:\winnt\system32\samlib.dll	c:\winnt\system32\clipsrv.exe Normal LocalSystem 0
netrap.dll 5.00.2134.1 11.27 KB (11,536 bytes) 12/7/1999	Distributed File System Dfs Running Auto Own
7:00:00 AM Microsoft Corporation	Process c:\winnt\system32\dfssvc.exe Normal LocalSystem 0
c:\winnt\system32\netrap.dll	DHCP Client Dhcp Running Auto Share Process
netapi32.dll 5.00.2195.2808 303.77 KB (311,056 bytes)	c:\winnt\system32\services.exe Normal LocalSystem 0
1	
2/10/2003 5:30:06 PM Microsoft Corporation	Logical Disk Manager Administrative Service dmadmin Stopped
2/10/2003 5:30:06 PM Microsoft Corporation c:\winnt\system32\netapi32.dll	Logical Disk Manager Administrative Service dmadmin Stopped Manual Share Process c:\winnt\system32\dmadmin.exe /com
2/10/2003 5:30:06 PM Microsoft Corporation c:\winnt\system32\netapi32.dll profmap.dll 5.00.2181.1 29.27 KB (29,968 bytes)	
c:\winnt\system32\netapi32.dll	Manual Share Process c:\winnt\system32\dmadmin.exe /com
c:\winnt\system32\netapi32.dll profmap.dll 5.00.2181.1 29.27 KB (29,968 bytes)	Manual Share Process c:\winnt\system32\dmadmin.exe /com Normal LocalSystem 0
c:\winnt\system32\netapi32.dll profmap.dll 5.00.2181.1 29.27 KB (29,968 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation	Manual Share Process c:\winnt\system32\dmadmin.exe /com Normal LocalSystem 0 Logical Disk Managerdmserver Running Auto Share Process
c:\winnt\system32\netapi32.dll profmap.dll 5.00.2181.1 29.27 KB (29,968 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\profmap.dll secur32.dll5.00.2195.2862 46.77 KB (47,888 bytes) 2/10/2003 5:30:12 PM Microsoft Corporation	Manual Share Process c:\winnt\system32\dmadmin.exe /com Normal LocalSystem 0 Logical Disk Managerdmserver Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0
c:\winnt\system32\netapi32.dll profmap.dll 5.00.2181.1 29.27 KB (29,968 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\profmap.dll secur32.dll5.00.2195.2862 46.77 KB (47,888 bytes) 2/10/2003	Manual Share Process c:\winnt\system32\dmadmin.exe /com Normal LocalSystem 0 Logical Disk Managerdmserver Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 DNS Client Dnscache Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 Event Log Eventlog Running Auto Share Process
c:\winnt\system32\netapi32.dll profmap.dll 5.00.2181.1 29.27 KB (29,968 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\profmap.dll secur32.dll5.00.2195.2862 46.77 KB (47,888 bytes) 2/10/2003 5:30:12 PM Microsoft Corporation c:\winnt\system32\secur32.dll sfc.dll 5.00.2195.2896 92.11 KB (94,320 bytes) 2/10/2003	Manual Share Process c:\winnt\system32\dmadmin.exe /com Normal LocalSystem 0 Logical Disk Managerdmserver Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 DNS Client Dnscache Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 Event Log Eventlog Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 Share Process c:\winnt\system32\services.exe Normal LocalSystem 0
c:\winnt\system32\netapi32.dll profmap.dll	Manual Share Process c:\winnt\system32\dmadmin.exe /com Normal LocalSystem 0 Logical Disk Managerdmserver Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 DNS Client Dnscache Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 Event Log Eventlog Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 COM+ Event System EventSystem Running Manual Share
c:\winnt\system32\netapi32.dll profmap.dll	Manual Share Process c:\winnt\system32\dmadmin.exe /com Normal LocalSystem 0 Logical Disk Managerdmserver Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 DNS Client Dnscache Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 Event Log Eventlog Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 COM+ Event System EventSystem Running Manual Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal
c:\winnt\system32\netapi32.dll profmap.dll 5.00.2181.1 29.27 KB (29,968 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\profmap.dll secur32.dll5.00.2195.2862 46.77 KB (47,888 bytes) 2/10/2003 5:30:12 PM Microsoft Corporation c:\winnt\system32\secur32.dll sfc.dll 5.00.2195.2896 92.11 KB (94,320 bytes) 2/10/2003 5:30:12 PM Microsoft Corporation c:\winnt\system32\secur32.dll sfc.dll 5.00.2195.2896 92.11 KB (94,320 bytes) 12/7/1999 12/7/1999	Manual Share Process c:\winnt\system32\dmadmin.exe /com Normal LocalSystem 0 Logical Disk Managerdmserver Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 DNS Client Dnscache Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 Event Log Eventlog Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 COM+ Event System EventSystem Running Manual Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0
c:\winnt\system32\netapi32.dll profmap.dll 5.00.2181.1 29.27 KB (29,968 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\profmap.dll secur32.dll5.00.2195.2862 46.77 KB (47,888 bytes) 2/10/2003 5:30:12 PM Microsoft Corporation c:\winnt\system32\secur32.dll 5:00.2195.2896 92.11 KB (94,320 bytes) 2/10/2003 5:30:12 PM Microsoft Corporation c:\winnt\system32\secur32.dll sfc.dll 5.00.2195.2896 92.11 KB (94,320 bytes) 2/10/2003 5:30:12 PM Microsoft Corporation c:\winnt\system32\scdll nddeapi.dll5.00.2137.1 15.27 KB (15,632 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation	Manual Share Process c:\winnt\system32\dmadmin.exe /com Normal LocalSystem 0 Logical Disk Managerdmserver c:\winnt\system32\services.exe Normal LocalSystem 0 DNS Client Dnscache Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 Event Log Eventlog Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 Event Log Eventlog Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 COM+ Event System EventSystem Running Manual Share Process c:\winnt\system32\services.exe -k netsvcs Normal LocalSystem 0 Fax Service Fax Stopped Manual Own Process
c:\winnt\system32\netapi32.dll profmap.dll 5.00.2181.1 29.27 KB (29,968 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\profmap.dll secur32.dll5.00.2195.2862 46.77 KB (47,888 bytes) 2/10/2003 5:30:12 PM Microsoft Corporation c:\winnt\system32\secur32.dll sfc.dll 5.00.2195.2896 92.11 KB (94,320 bytes) 2/10/2003 5:30:12 PM Microsoft Corporation c:\winnt\system32\secur32.dll sfc.dll 5.00.2195.2896 92.11 KB (94,320 bytes) 2/10/2003 5:30:12 PM Microsoft Corporation c:\winnt\system32\sfc.dll nddeapi.dll5.00.2137.1 15.27 KB (15,632 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\ndeapi.dll	Manual Share Process c:\winnt\system32\dmadmin.exe /com Normal LocalSystem 0 Logical Disk Managerdmserver c:\winnt\system32\services.exe Normal LocalSystem 0 DNS Client Dnscache Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 Event Log Eventlog Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 Event Log Eventlog Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 COM+ Event System EventSystem Running Manual Share Process c:\winnt\system32\services.exe \ Normal LocalSystem O Fax Service Fax Stopped Manual Own Process c:\winnt\system32\faxsvc.exe Normal LocalSystem 0
c:\winnt\system32\netapi32.dll profmap.dll 5.00.2181.1 29.27 KB (29,968 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\profmap.dll secur32.dll5.00.2195.2862 46.77 KB (47,888 bytes) 2/10/2003 5:30:12 PM Microsoft Corporation c:\winnt\system32\secur32.dll sfc.dll 5.00.2195.2896 92.11 KB (94,320 bytes) 2/10/2003 5:30:12 PM Microsoft Corporation c:\winnt\system32\secur32.dll sfc.dll 5.00.2195.2896 92.11 KB (94,320 bytes) 2/10/2003 5:30:12 PM Microsoft Corporation c:\winnt\system32\sfc.dll nddeapi.dll5.00.2137.1 15.27 KB (15,632 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\nddeapi.dll userenv.dll5.00.2195.2780 361.77 KB (370,448 bytes) 12/7/1999	Manual Share Process c:\winnt\system32\dmadmin.exe /com Normal LocalSystem 0 Logical Disk Managerdmserver c:\winnt\system32\services.exe Normal LocalSystem 0 DNS Client Dnscache Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 Event Log Eventlog Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 Event Log Eventlog Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 COM+ Event System EventSystem Running Manual Share Process c:\winnt\system32\services.exe \ Normal LocalSystem 0 Fax Service Fax Stopped Manual Own Process c:\winnt\system32\faxsvc.exe Normal LocalSystem 0 Internet Authentication Service IAS Running Auto Share
c:\winnt\system32\netapi32.dll profmap.dll 5.00.2181.1 29.27 KB (29,968 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\profmap.dll secur32.dll5.00.2195.2862 46.77 KB (47,888 bytes) 2/10/2003 5:30:12 PM Microsoft Corporation c:\winnt\system32\secur32.dll sfc.dll 5.00.2195.2896 92.11 KB (94,320 bytes) 2/10/2003 5:30:12 PM Microsoft Corporation c:\winnt\system32\sfc.dll nddeapi.dll5.00.2137.1 15.27 KB (15,632 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\shot Corporation c:\winnt\system32\shot Corporation c:\winnt\system32\shot Corporation c:\winnt\system32\shot Corporation c:\winnt\system32\nddeapi.dll userenv.dll5.00.2195.2780 361.77 KB (370,448 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation	Manual Share Process c:\winnt\system32\dmadmin.exe /com Normal LocalSystem 0 Logical Disk Managerdmserver c:\winnt\system32\services.exe Normal LocalSystem 0 DNS Client Dnscache Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 Event Log Eventlog Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 Event Event System Systems2\services.exe Normal LocalSystem 0 COM+ Event System EventSystem Running Manual Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 Fax Service Fax Stopped Manual Own Process c:\winnt\system32\faxsvc.exe Normal LocalSystem 0 Internet Authentication Service IAS Running Auto Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal
c:\winnt\system32\netapi32.dll profmap.dll	Manual Share Process c:\winnt\system32\dmadmin.exe /com Normal LocalSystem 0 Logical Disk Managerdmserver c:\winnt\system32\services.exe Normal LocalSystem 0 DNS Client Dnscache Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 Event Log Eventlog Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 Event Log Eventlog Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 COM+ Event System EventSystem Running Manual Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Fax Service Fax Stopped Manual Own Process c:\winnt\system32\faxsvc.exe Normal LocalSystem 0 Internet Authentication Service IAS Running Auto Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Internet Authentication Service IAS Running Auto Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0
c:\winnt\system32\netapi32.dll profmap.dll	Manual Share Process c:\winnt\system32\dmadmin.exe /com Normal LocalSystem 0 Logical Disk Managerdmserver c:\winnt\system32\services.exe Normal LocalSystem 0 DNS Client Dnscache Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 Event Log Eventlog Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 Event Lystem32\services.exe Normal LocalSystem 0 COM+ Event System EventSystem Running Manual Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 Fax Service Fax Stopped Manual Own Process c:\winnt\system32\faxsvc.exe Normal LocalSystem 0 Internet Authentication Service IAS Running Auto Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 IIS Admin Service IISADMIN Running Auto Share
c:\winnt\system32\netapi32.dll profmap.dll	Manual Share Process c:\winnt\system32\dmadmin.exe /com Normal LocalSystem 0 Logical Disk Managerdmserver c:\winnt\system32\services.exe Normal LocalSystem 0 DNS Client Dnscache Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 Event Log Eventlog Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 Event System EventSystem Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 COM+ Event System EventSystem Running Manual Share Process c:\winnt\system32\services.exe -k netsvcs Normal LocalSystem 0 Fax Service Fax Stopped Manual Own Process c:\winnt\system32\faxsvc.exe Normal LocalSystem 0 Internet Authentication Service IAS Running Auto Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 IIS Admin Service IISADMIN Running Auto Share Process c:\winnt\system32\inetsrv\inetinfo.exe Normal
c:\winnt\system32\netapi32.dll profmap.dll	Manual Share Process C:\winnt\system32\dmadmin.exe /com
c:\winnt\system32\netapi32.dll profmap.dll	Manual Share Process C:\winnt\system32\dmadmin.exe /com
c:\winnt\system32\netapi32.dll profmap.dll	Manual Share Process c:\winnt\system32\dmadmin.exe /com Normal LocalSystem 0 Logical Disk Managerdmserver Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 DNS Client Dnscache Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 Event Log Eventlog Running Auto Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 COM+ Event System EventSystem Running Manual Share Process c:\winnt\system32\services.exe Normal LocalSystem 0 Fax Service Fax Stopped Manual Own Process c:\winnt\system32\faxsvc.exe Normal LocalSystem 0 Internet Authentication Service IAS Running Auto Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 IIS Admin Service IISADMIN Running Auto Share Process c:\winnt\system32\interstr\int
c:\winnt\system32\netapi32.dll profmap.dll	Manual Share Process C:\winnt\system32\dmadmin.exe /com
c:\winnt\system32\netapi32.dll profmap.dll	Manual Share Process C:\winnt\system32\dmadmin.exe /com
c:\winnt\system32\netapi32.dll profmap.dll	Manual Share Process C:\winnt\system32\dmadmin.exe /com
c:\winnt\system32\netapi32.dll profmap.dll	Manual Share Process C:\winnt\system32\dmadmin.exe /com
c:\winnt\system32\netapi32.dll profmap.dll	Manual Share Process C:\winnt\system32\dmadmin.exe /com
c:\winnt\system32\netapi32.dll profmap.dll	Manual Share Process Normal LocalSystem O
c:\winnt\system32\netapi32.dll profmap.dll	Manual Share Process Normal LocalSystem O
c:\winnt\system32\netapi32.dll profmap.dll	Manual Share Process Normal LocalSystem O
c:\winnt\system32\netapi32.dll profmap.dll	Manual Share Process C:\winnt\system32\dmadmin.exe /com Normal LocalSystem 0
c:\winnt\system32\netapi32.dll profmap.dll	Manual Share Process C:\winnt\system32\dmadmin.exe /com Normal LocalSystem 0
c:\winnt\system32\netapi32.dll profmap.dll	Manual Share Process C:\winnt\system32\dmadmin.exe /com Normal LocalSystem 0

	Ir. and a series of the series of the series
Messenger Messenger Running Auto Share Process	Internet Connection Sharing SharedAccess Stopped Manual
c:\winnt\system32\services.exe Normal LocalSystem 0	Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal
NetMeeting Remote Desktop Sharing mnmsrvc Stopped Manual Own Process c:\winnt\system32\mnmsrvc.exe Normal	LocalSystem 0 Simple TCP/IP Services SimpTcp Running Auto Share
Own Process c:\winnt\system32\mnmsrvc.exe Normal LocalSystem 0	Simple TCP/IP Services SimpTcp Running Auto Share Process c:\winnt\system32\tcpsvcs.exe Normal LocalSystem 0
Distributed Transaction Coordinator MSDTC Running Auto	Print Spooler Spooler Running Auto Own Process
Own Process c:\winnt\system32\msdtc.exe Normal	c:\winnt\system32\spoolsv.exe Normal LocalSystem 0
LocalSystem 0	Performance Logs and Alerts SysmonLog Stopped Manual
Windows Installer MSIServer Stopped Manual Share Process	Own Process c:\winnt\system32\smlogsvc.exe Normal
c:\winnt\system32\msiexec.exe /v Normal LocalSystem 0	, ,
Network DDE NetDDE Stopped Manual Share Process	Telephony TapiSrv Running Manual Share Process
c:\winnt\system32\netdde.exe Normal LocalSystem 0	c:\winnt\system32\svchost.exe -k tapisrv
Network DDE DSDMNetDDEdsdm Stopped Manual Share	Terminal Services TermService Stopped Disabled Own
Process c:\winnt\system32\netdde.exe Normal LocalSystem 0	11
Net Logon Netlogon Stopped Manual Share Process	Telnet TlntSvr Stopped Manual Own Process
c:\winnt\system32\lsass.exe Normal LocalSystem 0	c:\winnt\system32\tlntsvr.exe Normal LocalSystem 0
Network Connections Netman Running Manual Share Process	Distributed Link Tracking Server TrkSvr Stopped Manual Share
c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0	
File Replication NtFrs Stopped Manual Own Process	Distributed Link Tracking Client TrkWks Running Auto Share
c:\winnt\system32\ntfrs.exe Ignore LocalSystem 0	Process c:\winnt\system32\services.exe Normal LocalSystem 0
NT LM Security Support Provider NtLmSsp Stopped Manual	Uninterruptible Power Supply UPS Stopped Manual Own
Share Process c:\winnt\system32\lsass.exe Normal	Process c:\winnt\system32\ups.exe Normal LocalSystem 0
LocalSystem 0	Utility Manager UtilMan Stopped Manual Own Process
Removable Storage NtmsSvc Running Auto Share Process	c:\winnt\system32\utilman.exe Normal LocalSystem 0
c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0	
Plug and Play PlugPlay Running Auto Share Process	c:\winnt\system32\services.exe Normal LocalSystem 0
c:\winnt\system32\services.exe Normal LocalSystem 0	World Wide Web Publishing Service W3SVC Running Auto
IPSEC Policy Agent Policy Agent Running Auto Share	Share Process c:\winnt\system32\inetsrv\inetinfo.exe Normal
Process c:\winnt\system32\lsass.exe Normal LocalSystem 0	LocalSystem 0
Protected Storage Protected Storage Running Auto Share	Windows Management Instrumentation WinMgmt Running Auto
Process c:\winnt\system32\services.exe Normal LocalSystem 0	Own Process c:\winnt\system32\wbem\winmgmt.exe Ignore
Remote Access Auto Connection Manager RasAuto Stopped Manual	LocalSystem 0
Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal	Windows Management Instrumentation Driver Extensions Wmi
LocalSystem 0	Running Manual Share Process c:\winnt\system32\services.exe
Remote Access Connection Manager RasMan Stopped Manual	N 1 7 10 4
Remote Access Connection Manager Rasiman Stopped Manual	Normal LocalSystem 0
Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal	Normal LocalSystem 0
	Normal LocalSystem 0 [Program Groups]
Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Routing and Remote Access RemoteAccess Stopped Disabled	[Program Groups]
Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0	
Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Routing and Remote Access RemoteAccess Stopped Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0	[Program Groups] Group Name Name User Name Accessories Default User: Accessories Default User
Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Routing and Remote Access RemoteAccess Stopped Disabled Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Remote Registry Service RemoteRegistry Running Auto	[Program Groups] Group Name Name User Name Accessories Default User:Accessories Default User Accessories\Accessibility Default User:Accessories\Accessibility
Share Process c:\winnt\system32\svchost.exe -k netsvcs LocalSystem 0 Routing and Remote Access RemoteAccess Stopped Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Remote Registry Service RemoteRegistry Running Own Process c:\winnt\system32\regsvc.exe Normal	[Program Groups] Group Name Name User Name Accessories Default User:Accessories Default User Accessories\Accessibility Default User:Accessories\Accessibility Default User
Share Process c:\winnt\system32\svchost.exe -k netsvcs LocalSystem 0 Routing and Remote Access RemoteAccess Stopped Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Remote Registry Service RemoteRegistry Running Own Process c:\winnt\system32\regsvc.exe Normal LocalSystem 0	[Program Groups] Group Name Name User Name Accessories Default User:Accessories Default User Accessories\Accessibility Default User:Accessories\Accessibility Default User Accessories\Entertainment Default User:Accessories\Entertainment
Share Process c:\winnt\system32\svchost.exe -k netsvcs LocalSystem 0 Routing and Remote Access RemoteAccess Stopped Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Remote Registry Service RemoteRegistry Running Own Process c:\winnt\system32\regsvc.exe Normal LocalSystem 0 Remote Command Service RMSYS Running Auto Own	[Program Groups] Group Name Name User Name Accessories Default User:Accessories Default User Accessories\Accessibility Default User:Accessories\Accessibility Default User Accessories\Entertainment Default User:Accessories\Entertainment Default User
Share Process c:\winnt\system32\svchost.exe -k netsvcs LocalSystem 0 Routing and Remote Access RemoteAccess Stopped Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Remote Registry Service RemoteRegistry Running Own Process c:\winnt\system32\regsvc.exe Normal LocalSystem 0 Remote Command Service RMSYS Running Auto Own Process c:\program files\benchcraft\rsys.exe Normal	[Program Groups] Group Name Name User Name Accessories Default User:Accessories Default User Accessories\Accessibility Default User:Accessories\Accessibility Default User Accessories\Entertainment Default User:Accessories\Entertainment Default User Accessories\System Tools Default User:Accessories\System Tools
Share Process c:\winnt\system32\svchost.exe -k netsvcs LocalSystem 0 Routing and Remote Access RemoteAccess Stopped Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Remote Registry Service RemoteRegistry Running Own Process c:\winnt\system32\regsvc.exe Normal LocalSystem 0 Remote Command Service RMSYS Running Auto Own Process c:\program files\benchcraft\rsys.exe Normal \.\Administrator 0	[Program Groups] Group Name Name User Name Accessories Default User:Accessories Default User Accessories\Accessibility Default User:Accessories\Accessibility Default User Accessories\Entertainment Default User Accessories\System Tools Default User:Accessories\System Tools Default User
Share Process c:\winnt\system32\svchost.exe -k netsvcs LocalSystem 0 Routing and Remote Access RemoteAccess Stopped Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Remote Registry Service RemoteRegistry Running Own Process c:\winnt\system32\regsvc.exe Normal LocalSystem 0 Remote Command Service RMSYS Running Auto Own Process c:\program files\benchcraft\rsys.exe Normal \.\Administrator 0 Remote Procedure Call (RPC) Locator RpcLocator Stopped	[Program Groups] Group Name Name User Name Accessories Default User:Accessories Default User Accessories\Accessibility Default User:Accessories\Accessibility Default User Accessories\Entertainment Default User Accessories\System Tools Default User:Accessories\System Tools Default User Startup Default User:Startup Default User
Share Process c:\winnt\system32\svchost.exe -k netsvcs LocalSystem 0 Routing and Remote Access RemoteAccess Stopped Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Remote Registry Service RemoteRegistry Running Own Process c:\winnt\system32\regsvc.exe Normal LocalSystem 0 Remote Command Service RMSYS Running Auto Own Process c:\program files\benchcraft\rsys.exe Normal \.\Administrator 0 Remote Procedure Call (RPC) Locator RpcLocator Stopped Manual Own Process c:\winnt\system32\locator.exe Normal	[Program Groups] Group Name Name User Name Accessories Default User:Accessories Default User Accessories\Accessibility Default User:Accessories\Accessibility Default User Accessories\Entertainment Default User:Accessories\Entertainment Default User Accessories\System Tools Default User:Accessories\System Tools Default User Startup Default User:Startup Default User Accessories All Users:Accessories All Users
Share Process c:\winnt\system32\svchost.exe -k netsvcs LocalSystem 0 Routing and Remote Access RemoteAccess Stopped Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Remote Registry Service RemoteRegistry Running Own Process c:\winnt\system32\regsvc.exe Normal LocalSystem 0 Remote Command Service RMSYS Running Auto Own Process c:\program files\benchcraft\rsys.exe Normal \text{.Administrator} 0 Remote Procedure Call (RPC) Locator RpcLocator Stopped Manual Own Process c:\winnt\system32\locator.exe Normal \text{.Administrator} 0 Remote Procedure Call (RPC) Locator RpcLocator Stopped Manual Own Process c:\winnt\system32\locator.exe Normal \text{.Administrator} 0	[Program Groups] Group Name Name User Name Accessories Default User:Accessories Default User Accessories\Accessibility Default User:Accessories\Accessibility Default User Accessories\Entertainment Default User:Accessories\Entertainment Default User Accessories\System Tools Default User:Accessories\System Tools Default User Startup Default User:Startup Default User Accessories All Users:Accessories All Users Accessories\Accessibility All Users:Accessories\Accessibility
Share Process c:\winnt\system32\svchost.exe -k netsvcs	[Program Groups] Group Name Name User Name Accessories Default User:Accessories Default User Accessories\Accessibility Default User:Accessories\Accessibility Default User Accessories\Entertainment Default User:Accessories\Entertainment Default User Accessories\System Tools Default User:Accessories\System Tools Default User Startup Default User:Startup Default User Accessories All Users:Accessories All Users Accessories All Users:Accessories\Accessibility All Users
Share Process c:\winnt\system32\svchost.exe -k netsvcs	[Program Groups] Group Name Name User Name Accessories Default User:Accessories Default User Accessories\Accessibility Default User:Accessories\Accessibility Default User Accessories\Entertainment Default User:Accessories\Entertainment Default User Accessories\System Tools Default User:Accessories\System Tools Default User Startup Default User:Startup Default User Accessories All Users:Accessories All Users Accessories\Accessibility All Users Accessories\Communications All Users:Accessories\Communications
Share Process c:\winnt\system32\svchost.exe -k netsvcs LocalSystem 0 Routing and Remote Access RemoteAccess Stopped Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Remote Registry Service RemoteRegistry Running Own Process c:\winnt\system32\regsvc.exe Normal LocalSystem 0 Remote Command Service RMSYS Running Auto Own Process c:\program files\benchcraft\rsys.exe Normal \.\Administrator 0 Remote Procedure Call (RPC) Locator RpcLocator Stopped Manual Own Process c:\winnt\system32\regsvc.exe Normal LocalSystem 0 Remote Procedure Call (RPC) RpcSs Running Auto Share Process c:\winnt\system32\svchost -k rpcss Normal LocalSystem 0	[Program Groups] Group Name Name User Name Accessories Default User:Accessories Default User Accessories\Accessibility Default User:Accessories\Accessibility Default User Accessories\Entertainment Default User:Accessories\Entertainment Default User Accessories\System Tools Default User:Accessories\System Tools Default User Startup Default User:Startup Default User Accessories All Users:Accessories All Users Accessories\Accessibility All Users Accessories\Communications All Users:Accessories\Communications All Users
Share Process c:\winnt\system32\svchost.exe -k netsvcs LocalSystem 0 Routing and Remote Access RemoteAccess Stopped Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Remote Registry Service RemoteRegistry Running Own Process c:\winnt\system32\regsvc.exe Normal LocalSystem 0 Remote Command Service RMSYS Running Auto Own Process c:\program files\benchcraft\rsys.exe Normal \.\Administrator 0 Remote Procedure Call (RPC) Locator RpcLocator Stopped Manual Own Process c:\winnt\system32\rsystem32\locator.exe Normal LocalSystem 0 Remote Procedure Call (RPC) RpcSs Running Auto Share Process c:\winnt\system32\svchost -k rpcss Normal LocalSystem 0 QoS Admission Control (RSVP) RSVP Running Auto Own	[Program Groups] Group Name Name User Name Accessories Default User:Accessories Default User Accessories\Accessibility Default User:Accessories\Accessibility Default User Accessories\Entertainment Default User:Accessories\Entertainment Default User Accessories\System Tools Default User:Accessories\System Tools Default User Startup Default User:Startup Default User Accessories All Users:Accessories All Users Accessories\Accessibility All Users Accessories\Communications All Users:Accessories\Communications All Users Accessories\Entertainment All Users:Accessories\Entertainment
Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Routing and Remote Access RemoteAccess Stopped Disabled Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Remote Registry Service RemoteRegistry Running Own Process c:\winnt\system32\regsvc.exe Normal LocalSystem 0 Remote Command Service RMSYS Running Auto Own Process c:\program files\benchcraft\rsys.exe Normal \.\Administrator 0 Remote Procedure Call (RPC) Locator RpcLocator Stopped Manual Own Process c:\winnt\system32\regsvc.exe Normal LocalSystem 0 Remote Procedure Call (RPC) RpcSs Running Auto Share Process c:\winnt\system32\svchost -k rpcss Normal LocalSystem 0 QoS Admission Control (RSVP) RSVP Running Auto Own Process c:\winnt\system32\rsvp.exe -s Normal LocalSystem 0	[Program Groups] Group Name Name User Name Accessories Default User:Accessories Default User Accessories\Accessibility Default User:Accessories\Accessibility Default User Accessories\Entertainment Default User:Accessories\Entertainment Default User Accessories\System Tools Default User:Accessories\System Tools Default User Startup Default User:Startup Default User Accessories All Users:Accessories All Users Accessories\Accessibility All Users Accessories\Communications All Users:Accessories\Communications All Users Accessories\Entertainment All Users:Accessories\Entertainment All Users
Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Routing and Remote Access RemoteAccess Stopped Disabled Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Remote Registry Service RemoteRegistry Running Own Process c:\winnt\system32\regsvc.exe Normal LocalSystem 0 Remote Command Service RMSYS Running Auto Own Process c:\program files\benchcraft\rsys.exe Normal \.\Administrator 0 Remote Procedure Call (RPC) Locator RpcLocator Stopped Manual Own Process c:\winnt\system32\regsvc.exe Normal LocalSystem 0 Remote Procedure Call (RPC) RpcSs Running Auto Share Process c:\winnt\system32\svchost -k rpcss Normal LocalSystem 0 QoS Admission Control (RSVP) RSVP Running Auto Own Process c:\winnt\system32\rsvp.exe -s Normal LocalSystem 0 QoS Admission Control (RSVP) RSVP Running Auto Own Process c:\winnt\system32\rsvp.exe -s Normal LocalSystem 0 Security Accounts Manager SamSs Running Auto Share	[Program Groups] Group Name Name User Name Accessories Default User:Accessories Default User Accessories\Accessibility Default User:Accessories\Accessibility Default User Accessories\Entertainment Default User:Accessories\Entertainment Default User Accessories\System Tools Default User:Accessories\System Tools Default User Startup Default User:Startup Default User Accessories All Users:Accessories All Users Accessories\Accessibility All Users Accessories\Communications All Users:Accessories\Communications All Users Accessories\Entertainment All Users:Accessories\Entertainment All Users Accessories\Games All Users:Accessories\Entertainment All Users Accessories\Games All Users:Accessories\Entertainment All Users Accessories\Games All Users:Accessories\Games All Users
Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Routing and Remote Access RemoteAccess Stopped Disabled Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Remote Registry Service RemoteRegistry Running Auto Own Process c:\winnt\system32\regsvc.exe Normal LocalSystem 0 Remote Command Service RMSYS Running Auto Own Process c:\program files\benchcraft\rsys.exe Normal \text{.\Administrator} 0 Remote Procedure Call (RPC) Locator RpcLocator Stopped Manual Own Process c:\winnt\system32\svchost -k rpcss Normal LocalSystem 0 Remote Procedure Call (RPC) RpcSs Running Auto Share Process c:\winnt\system32\svchost -k rpcss Normal LocalSystem 0 QoS Admission Control (RSVP) RSVP Running Auto Own Process c:\winnt\system32\rsvp.exe -s Normal LocalSystem 0 Security Accounts Manager SamSs Running Auto Share Process c:\winnt\system32\specked -s Normal LocalSystem 0 Security Accounts Manager SamSs Running Auto Share Process c:\winnt\system32\specked -s Normal LocalSystem 0 Routing and Remote Access RemoteAccess Stopped Disabled Disable	[Program Groups] Group Name Name User Name Accessories Default User:Accessories Default User Accessories\Accessibility Default User:Accessories\Accessibility Default User Accessories\Entertainment Default User:Accessories\Entertainment Default User Accessories\System Tools Default User:Accessories\System Tools Default User Startup Default User:Startup Default User Accessories All Users:Accessories All Users Accessories\Accessibility All Users Accessories\Communications All Users:Accessories\Communications All Users Accessories\Entertainment All Users:Accessories\Entertainment All Users Accessories\Games All Users:Accessories\Games All Users Accessories\Games All Users:Accessories\Communications All Users Accessories\Games All Users:Accessories\Games All Users Accessories\Games All Users Accessories\Microsoft Script Debugger All Users:Accessories\Microsoft
Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Routing and Remote Access RemoteAccess Stopped Disabled Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Remote Registry Service RemoteRegistry Running Auto Own Process c:\winnt\system32\regsvc.exe Normal LocalSystem 0 Remote Command Service RMSYS Running Auto Own Process c:\program files\benchcraft\rsys.exe Normal \tag{Normal} Normal	[Program Groups] Group Name Name User Name Accessories Default User:Accessories Default User Accessories\Accessibility Default User:Accessories\Accessibility Default User Accessories\Entertainment Default User:Accessories\Entertainment Default User Accessories\System Tools Default User:Accessories\System Tools Default User Startup Default User:Startup Default User Accessories All Users:Accessories All Users Accessories\Accessibility All Users Accessories\Accessibility All Users:Accessories\Accessibility All Users Accessories\Communications All Users:Accessories\Entertainment All Users Accessories\Games All Users:Accessories\Entertainment All Users Accessories\Games All Users:Accessories\Games All Users Accessories\Microsoft Script Debugger All Users:Accessories\Microsoft Script Debugger All Users
Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Routing and Remote Access RemoteAccess Stopped Disabled Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Remote Registry Service RemoteRegistry Running Auto Own Process c:\winnt\system32\regsvc.exe Normal LocalSystem 0 Remote Command Service RMSYS Running Auto Own Process c:\program files\benchcraft\rsys.exe Normal \text{.Administrator} 0 Remote Procedure Call (RPC) Locator RpcLocator Stopped Manual Own Process c:\winnt\system32\svchost-k rpcss Normal LocalSystem 0 Remote Procedure Call (RPC) RpcSs Running Auto Share Process c:\winnt\system32\svchost-k rpcss Normal LocalSystem 0 QoS Admission Control (RSVP) RSVP Running Auto Own Process c:\winnt\system32\rsvp.exe -s Normal LocalSystem 0 Security Accounts Manager SamSs Running Auto Share Process c:\winnt\system32\rsvp.exe -s Normal LocalSystem 0 Security Accounts Manager SamSs Running Auto Share Process c:\winnt\system32\rsvp.exe -s Normal LocalSystem 0 Smart Card Helper SCardDrv Stopped Manual Share Process c:\winnt\system32\scardsvr.exe Ignore LocalSystem 0	[Program Groups] Group Name Name User Name Accessories Default User:Accessories Default User Accessories\Accessibility Default User:Accessories\Accessibility Default User Accessories\Entertainment Default User:Accessories\Entertainment Default User Accessories\System Tools Default User:Accessories\System Tools Default User Startup Default User:Startup Default User Accessories All Users:Accessories All Users Accessories\Accessibility All Users Accessories\Communications All Users:Accessories\Communications All Users Accessories\Entertainment All Users:Accessories\Entertainment All Users Accessories\Games All Users:Accessories\Games All Users Accessories\Games All Users:Accessories\Games All Users Accessories\Microsoft Script Debugger All Users:Accessories\System Tools All Users:Accessories\System Tools
Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Routing and Remote Access RemoteAccess Stopped Disabled Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Remote Registry Service RemoteRegistry Running Auto Own Process c:\winnt\system32\regsvc.exe Normal LocalSystem 0 Remote Command Service RMSYS Running Auto Own Process c:\program files\benchcraft\rsys.exe Normal \text{.Administrator} 0 Remote Procedure Call (RPC) Locator RpcLocator Stopped Manual Own Process c:\winnt\system32\rsystem32\locator.exe Normal LocalSystem 0 Remote Procedure Call (RPC) RpcSs Running Auto Share Process c:\winnt\system32\svchost -k rpcss Normal LocalSystem 0 QoS Admission Control (RSVP) RSVP Running Auto Own Process c:\winnt\system32\rsvp.exe -s Normal LocalSystem 0 QoS Admission Control (RSVP) RSVP Running Auto Share Process c:\winnt\system32\rsvp.exe -s Normal LocalSystem 0 Security Accounts Manager SamSs Running Auto Share Process c:\winnt\system32\lsass.exe Normal LocalSystem 0 Smart Card Helper SCardDrv Stopped Manual Share Process c:\winnt\system32\scardsvr.exe Ignore LocalSystem 0 Smart Card SCardSvr Stopped Manual Share Process	[Program Groups] Group Name Name User Name Accessories Default User:Accessories Default User Accessories\Accessibility Default User:Accessories\Accessibility Default User Accessories\Entertainment Default User:Accessories\Entertainment Default User Accessories\System Tools Default User:Accessories\System Tools Default User Startup Default User:Startup Default User Accessories All Users:Accessories All Users Accessories\Accessibility All Users:Accessories\Accessibility All Users Accessories\Communications All Users:Accessories\Communications All Users Accessories\Entertainment All Users:Accessories\Entertainment All Users Accessories\Games All Users:Accessories\Games All Users Accessories\Microsoft Script Debugger All Users:Accessories\Microsoft Script Debugger All Users Accessories\System Tools All Users:Accessories\System Tools All Users
Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Routing and Remote Access RemoteAccess Stopped Disabled Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Remote Registry Service RemoteRegistry Running Auto Own Process c:\winnt\system32\regsvc.exe Normal LocalSystem 0 Remote Command Service RMSYS Running Auto Own Process c:\program files\benchcraft\rsys.exe Normal LocalSystem 0 Remote Procedure Call (RPC) Locator RpcLocator Stopped Manual Own Process c:\winnt\system32\locator.exe Normal LocalSystem 0 Remote Procedure Call (RPC) RpcSs Running Auto Share Process c:\winnt\system32\svchost -k rpcss Normal LocalSystem 0 QoS Admission Control (RSVP) RSVP Running Auto Own Process c:\winnt\system32\rsvp.exe -s Normal LocalSystem 0 Security Accounts Manager SamSs Running Auto Share Process c:\winnt\system32\lsass.exe Normal LocalSystem 0 Smart Card Helper SCardDrv Stopped Manual Share Process c:\winnt\system32\scardsvr.exe Ignore LocalSystem 0 Smart Card SCardSvr Stopped Manual Share Process c:\winnt\system32\scardsvr.exe Ignore LocalSystem 0	[Program Groups] Group Name Name User Name Accessories Default User:Accessories Default User Accessories\Accessibility Default User:Accessories\Accessibility Default User Accessories\Entertainment Default User:Accessories\Entertainment Default User Accessories\System Tools Default User:Accessories\System Tools Default User Startup Default User:Startup Default User Accessories All Users:Accessories All Users Accessories\Accessibility All Users:Accessories\Accessibility All Users Accessories\Communications All Users:Accessories\Communications All Users Accessories\Entertainment All Users:Accessories\Entertainment All Users Accessories\Games All Users:Accessories\Games All Users Accessories\Microsoft Script Debugger All Users Accessories\System Tools All Users:Accessories\System Tools All Users Accessories\System Tools All Users:Accessories\System Tools All Users Administrative Tools All Users:Administrative Tools All Users
Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Routing and Remote Access RemoteAccess Stopped Disabled Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Remote Registry Service RemoteRegistry Running Auto Own Process c:\winnt\system32\regsvc.exe Normal LocalSystem 0 Remote Command Service RMSYS Running Auto Own Process c:\program files\benchcraft\rsys.exe Normal LocalSystem 0 Remote Procedure Call (RPC) Locator RpcLocator Stopped Manual Own Process c:\winnt\system32\locator.exe Normal LocalSystem 0 Remote Procedure Call (RPC) RpcSs Running Auto Share Process c:\winnt\system32\svchost -k rpcss Normal LocalSystem 0 QoS Admission Control (RSVP) RSVP Running Auto Own Process c:\winnt\system32\rsvp.exe -s Normal LocalSystem 0 Security Accounts Manager SamSs Running Auto Share Process c:\winnt\system32\rsvp.exe -s Normal LocalSystem 0 Smart Card Helper SCardDrv Stopped Manual Share Process c:\winnt\system32\scardsvr.exe Ignore LocalSystem 0 Smart Card SCardSvr Stopped Manual Share Process c:\winnt\system32\scardsvr.exe Ignore LocalSystem 0 Task Scheduler Schedule Running Auto Share Process c:\winnt\system32\scardsvr.exe Ignore LocalSystem 0 Task Scheduler Schedule Running Auto Share Process c:\winnt\system32\scardsvr.exe Ignore LocalSystem 0	[Program Groups] Group Name Name User Name Accessories Default User:Accessories Default User Accessories\Accessibility Default User:Accessories\Accessibility Default User Accessories\Entertainment Default User:Accessories\Entertainment Default User Accessories\System Tools Default User:Accessories\System Tools Default User Startup Default User:Startup Default User Accessories All Users:Accessories All Users Accessories\Accessibility All Users Accessories\Communications All Users:Accessories\Communications All Users Accessories\Entertainment All Users:Accessories\Entertainment All Users Accessories\Games All Users:Accessories\Games All Users Accessories\Microsoft Script Debugger All Users:Accessories\Microsoft Script Debugger All Users Accessories\System Tools All Users:Accessories\System Tools All Users Administrative Tools All Users:Administrative Tools All Users Microsoft SQL Server All Users:Microsoft SQL Server All Users
Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Routing and Remote Access RemoteAccess Stopped Disabled Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Remote Registry Service RemoteRegistry Running Auto Own Process c:\winnt\system32\regsvc.exe Normal LocalSystem 0 Remote Command Service RMSYS Running Auto Own Process c:\program files\benchcraft\rsys.exe Normal \text{.Administrator} 0 Remote Procedure Call (RPC) Locator RpcLocator Stopped Manual Own Process c:\winnt\system32\svchost -k rpcss Normal LocalSystem 0 Remote Procedure Call (RPC) RpcSs Running Auto Share Process c:\winnt\system32\svchost -k rpcss Normal LocalSystem 0 QoS Admission Control (RSVP) RSVP Running Auto Own Process c:\winnt\system32\svchost -k rpcss Normal LocalSystem 0 Security Accounts Manager SamSs Running Auto Share Process c:\winnt\system32\ssass.exe Normal LocalSystem 0 Smart Card Helper SCardDrv Stopped Manual Share Process c:\winnt\system32\scardsvr.exe Ignore LocalSystem 0 Smart Card SCardSvr Stopped Manual Share Process c:\winnt\system32\scardsvr.exe Ignore LocalSystem 0 Task Scheduler Schedule Running Auto Share Process c:\winnt\system32\scardsvr.exe Ignore LocalSystem 0 Task Scheduler Schedule Running Auto Share Process c:\winnt\system32\scardsvr.exe Ignore LocalSystem 0	[Program Groups] Group Name Name User Name Accessories Default User:Accessories Default User Accessories\Accessibility Default User:Accessories\Accessibility Default User Accessories\Entertainment Default User:Accessories\Entertainment Default User Accessories\System Tools Default User:Accessories\System Tools Default User Startup Default User:Startup Default User Accessories All Users:Accessories All Users Accessories\Accessibility All Users Accessories\Accessibility All Users:Accessories\Communications All Users Accessories\Entertainment All Users:Accessories\Entertainment All Users Accessories\Games All Users:Accessories\Games All Users Accessories\Microsoft Script Debugger All Users:Accessories\Microsoft Script Debugger All Users Accessories\System Tools All Users Administrative Tools All Users:Administrative Tools All Users Microsoft SQL Server All Users Startup All Users:Startup All Users
Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Routing and Remote Access RemoteAccess Stopped Disabled Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Remote Registry Service RemoteRegistry Running Auto Own Process c:\winnt\system32\regsvc.exe Normal LocalSystem 0 Remote Command Service RMSYS Running Auto Own Process c:\program files\benchcraft\rsys.exe Normal \text{Normal} Normalo	[Program Groups] Group Name Name User Name Accessories Default User:Accessories Default User Accessories\Accessibility Default User:Accessories\Accessibility Default User Accessories\Entertainment Default User:Accessories\Entertainment Default User Accessories\System Tools Default User:Accessories\System Tools Default User Startup Default User:Startup Default User Accessories All Users:Accessories All Users Accessories\Accessibility All Users Accessories\Accessibility All Users:Accessories\Communications All Users Accessories\Communications All Users:Accessories\Communications All Users Accessories\Games All Users:Accessories\Games All Users Accessories\Games All Users Accessories\Microsoft Script Debugger All Users:Accessories\Microsoft Script Debugger All Users Accessories\System Tools All Users Administrative Tools All Users:Administrative Tools All Users Microsoft SQL Server All Users Startup All Users:Startup All Users Accessories CLIENT10\Administrator:Accessories
Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Routing and Remote Access RemoteAccess Stopped Disabled Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Remote Registry Service RemoteRegistry Running Auto Own Process c:\winnt\system32\regsvc.exe Normal LocalSystem 0 Remote Command Service RMSYS Running Auto Own Process c:\program files\benchcraft\rsys.exe Normal \text{Normal} Normalo	[Program Groups] Group Name Name User Name Accessories Default User:Accessories Default User Accessories\Accessibility Default User:Accessories\Accessibility Default User Accessories\Entertainment Default User:Accessories\Entertainment Default User Accessories\System Tools Default User:Accessories\System Tools Default User Startup Default User:Startup Default User Accessories All Users:Accessories All Users Accessories\Accessibility All Users:Accessories\Accessibility All Users Accessories\Communications All Users:Accessories\Communications All Users Accessories\Games All Users:Accessories\Entertainment All Users Accessories\Microsoft Script Debugger All Users:Accessories\Microsoft Script Debugger All Users Accessories\System Tools All Users:Accessories\System Tools All Users Accessories\System Tools All Users:Administrative Tools All Users Microsoft SQL Server All Users:Microsoft SQL Server All Users Accessories CLIENT10\Administrator:Accessories CLIENT10\Administrator
Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Routing and Remote Access RemoteAccess Stopped Disabled Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Remote Registry Service RemoteRegistry Running Auto Own Process c:\winnt\system32\regsvc.exe Normal LocalSystem 0 Remote Command Service RMSYS Running Auto Own Process c:\program files\benchcraft\rsys.exe Normal LocalSystem 0 Remote Procedure Call (RPC) Locator RpcLocator Stopped Manual Own Process c:\winnt\system32\svchost -k rpcss Normal LocalSystem 0 Remote Procedure Call (RPC) RpcSs Running Auto Share Process c:\winnt\system32\svchost -k rpcss Normal LocalSystem 0 QoS Admission Control (RSVP) RSVP Running Auto Own Process c:\winnt\system32\rsvp.exe -s Normal LocalSystem 0 Security Accounts Manager SamSs Running Auto Share Process c:\winnt\system32\rsvp.exe -s Normal LocalSystem 0 Security Accounts Manager SamSs Running Auto Share Process c:\winnt\system32\rsvp.exe -s Normal LocalSystem 0 Smart Card Helper SCardDrv Stopped Manual Share Process c:\winnt\system32\scardsvr.exe Ignore LocalSystem 0 Smart Card SCardSvr Stopped Manual Share Process c:\winnt\system32\scardsvr.exe Ignore LocalSystem 0 Task Scheduler Schedule Running Auto Share Process c:\winnt\system32\scardsvr.exe Ignore LocalSystem 0 RunAs Service seclogon Running Auto Share Process c:\winnt\system32\scardsvr.exe Ignore LocalSystem 0 System Event Notification SENS Running Auto Share	[Program Groups] Group Name Name User Name Accessories Default User:Accessories Default User Accessories\Accessibility Default User:Accessories\Accessibility Default User Accessories\Entertainment Default User:Accessories\Entertainment Default User Accessories\System Tools Default User:Accessories\System Tools Default User Startup Default User:Startup Default User Accessories All Users:Accessories All Users Accessories\Accessibility All Users Accessories\Accessibility All Users:Accessories\Communications All Users Accessories\Communications All Users:Accessories\Entertainment All Users Accessories\Games All Users:Accessories\Games All Users Accessories\Games All Users:Accessories\Games All Users Accessories\Microsoft Script Debugger All Users:Accessories\Microsoft Script Debugger All Users Accessories\System Tools All Users:Accessories\System Tools All Users Accessories\System Tools All Users:Administrative Tools All Users Microsoft SQL Server All Users:Microsoft SQL Server All Users Startup All Users:Startup All Users Accessories CLIENT10\Administrator:Accessories CLIENT10\Administrator Accessories\Accessories\Accessories\Clientication Accessories\Accessories\Accessories\Data Accessories\Clientication Accessories\Accessories\Data Accessories\Clientication Accessories\Accessories\Data Accessories
Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Routing and Remote Access RemoteAccess Stopped Disabled Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Remote Registry Service RemoteRegistry Running Auto Own Process c:\winnt\system32\regsvc.exe Normal LocalSystem 0 Remote Command Service RMSYS Running Auto Own Process c:\program files\benchcraft\rsys.exe Normal LocalSystem 0 Remote Procedure Call (RPC) Locator RpcLocator Stopped Manual Own Process c:\winnt\system32\svchost.exe Normal LocalSystem 0 Remote Procedure Call (RPC) RpcSs Running Auto Share Process c:\winnt\system32\svchost.exe rpcss Normal LocalSystem 0 QoS Admission Control (RSVP) RSVP Running Auto Own Process c:\winnt\system32\rsvp.exe -s Normal LocalSystem 0 Security Accounts Manager SamSs Running Auto Share Process c:\winnt\system32\svchost.exe Normal LocalSystem 0 Security Accounts Manager SamSs Running Auto Share Process c:\winnt\system32\svchost.exe Ignore LocalSystem 0 Smart Card Helper SCardDrv Stopped Manual Share Process c:\winnt\system32\scardsvr.exe Ignore LocalSystem 0 Task Scheduler Schedule Running Auto Share Process c:\winnt\system32\scardsvr.exe Ignore LocalSystem 0 RunAs Service seclogon Running Auto Share Process c:\winnt\system32\scardsvr.exe Ignore LocalSystem 0 System Event Notification SENS Running Auto Share Process c:\winnt\system32\scardsvr.exe Ignore LocalSystem 0 System Event Notification SENS Running Auto Share Process c:\winnt\system32\scardsvr.exe -k netsvcs Normal	[Program Groups] Group Name Name User Name Accessories Default User: Accessories Default User Accessories\Accessibility Default User: Accessories\Accessibility Default User Accessories\Entertainment Default User: Accessories\Entertainment Default User Accessories\System Tools Default User: Accessories\System Tools Default User Startup Default User: Startup Default User Accessories All Users: Accessories All Users Accessories\Accessibility All Users: Accessories\Accessibility All Users Accessories\Communications All Users: Accessories\Communications All Users Accessories\Entertainment All Users Accessories\Games All Users: Accessories\Games All Users Accessories\Microsoft Script Debugger All Users: Accessories\Microsoft Script Debugger All Users Accessories\System Tools All Users Administrative Tools All Users: Administrative Tools All Users Microsoft SQL Server All Users: Microsoft SQL Server All Users Startup All Users: Startup All Users Accessories CLIENT10\Administrator: Accessories Accessories\Accessories\Accessories CLIENT10\Administrator: Accessories\Accessibility CLIENT10\Administrator: Accessories\Accessibility
Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Routing and Remote Access RemoteAccess Stopped Disabled Share Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0 Remote Registry Service RemoteRegistry Running Auto Own Process c:\winnt\system32\regsvc.exe Normal LocalSystem 0 Remote Command Service RMSYS Running Auto Own Process c:\program files\benchcraft\rsys.exe Normal LocalSystem 0 Remote Procedure Call (RPC) Locator RpcLocator Stopped Manual Own Process c:\winnt\system32\svchost -k rpcss Normal LocalSystem 0 Remote Procedure Call (RPC) RpcSs Running Auto Share Process c:\winnt\system32\svchost -k rpcss Normal LocalSystem 0 QoS Admission Control (RSVP) RSVP Running Auto Own Process c:\winnt\system32\rsvp.exe -s Normal LocalSystem 0 Security Accounts Manager SamSs Running Auto Share Process c:\winnt\system32\rsvp.exe -s Normal LocalSystem 0 Security Accounts Manager SamSs Running Auto Share Process c:\winnt\system32\rsvp.exe -s Normal LocalSystem 0 Smart Card Helper SCardDrv Stopped Manual Share Process c:\winnt\system32\scardsvr.exe Ignore LocalSystem 0 Smart Card SCardSvr Stopped Manual Share Process c:\winnt\system32\scardsvr.exe Ignore LocalSystem 0 Task Scheduler Schedule Running Auto Share Process c:\winnt\system32\scardsvr.exe Ignore LocalSystem 0 RunAs Service seclogon Running Auto Share Process c:\winnt\system32\scardsvr.exe Ignore LocalSystem 0 System Event Notification SENS Running Auto Share	[Program Groups] Group Name Name User Name Accessories Default User:Accessories Default User Accessories\Accessibility Default User:Accessories\Accessibility Default User Accessories\Entertainment Default User:Accessories\Entertainment Default User Accessories\System Tools Default User:Accessories\System Tools Default User Startup Default User:Startup Default User Accessories All Users:Accessories All Users Accessories\Accessibility All Users Accessories\Accessibility All Users:Accessories\Communications All Users Accessories\Communications All Users:Accessories\Entertainment All Users Accessories\Games All Users:Accessories\Games All Users Accessories\Games All Users:Accessories\Games All Users Accessories\Microsoft Script Debugger All Users:Accessories\Microsoft Script Debugger All Users Accessories\System Tools All Users:Accessories\System Tools All Users Accessories\System Tools All Users:Administrative Tools All Users Microsoft SQL Server All Users:Microsoft SQL Server All Users Startup All Users:Startup All Users Accessories CLIENT10\Administrator:Accessories CLIENT10\Administrator Accessories\Accessories\Accessories\Clientication Accessories\Accessories\Accessories\Data Accessories\Clientication Accessories\Accessories\Data Accessories\Clientication Accessories\Accessories\Data Accessories

Accessories\Entertainment enhsig.dll <File Missing> Not Available Not Available CLIENT10\Administrator:Accessories\Entertainment Not Available Not Available CLIENT10\Administrator iemigrat.dll <File Missing> Not Available Not CLIENT10\Administrator:Accessories\System Accessories\System Tools Available Not Available Not Available CLIENT10\Administrator iesetup.dll 5.0.3103.1000 57 KB 5/4/2001 1:05:02 PM Tools Administrative Tools CLIENT10\Administrator: Administrative Tools C:\WINNT\system32 Microsoft Corporation CLIENT10\Administrator 5.0.2920.0 59 KB 12/7/1999 8:00:00 AM iexplore.exe Benchcraft CLIENT10\Administrator:Benchcraft C:\Program Files\Internet Explorer Microsoft Corporation CLIENT10\Administrator 5.0.2195.2778 126 KB 5/4/2001 1:05:02 PM imagehlp.dll C:\WINNT\system32 Microsoft Corporation CLIENT10\Administrator:Startup Startup CLIENT10\Administrator imghelp.dll <File Missing> Not Available Not Available Not Available Not Available inseng.dll 5.0.3103.1000 [Startup Programs] 72 KB 5/4/2001 1:05:02 PM C:\WINNT\system32 Microsoft Corporation Program Command User NameLocation jobexec.dll 5.0.0.1 47 KB 12/7/1999 8:00:00 AM No startup program information C:\WINNT\system32 Microsoft Corporation jscript.dll 5.1.0.5907 476 KB 5/4/2001 1:05:02 PM C:\WINNT\system32 [OLE Registration] Microsoft Corporation jsproxy.dll 5.0.2920.0 13 KB 12/7/1999 8:00:00 AM Object Local Server C:\WINNT\system32 Microsoft Corporation Sound (OLE2) sndrec32.exe <File Missing> Not Available msaahtml.dll Not Media Clip mplay32.exe Available Not Available Not Available Video Clipmplay32.exe /avi 2290 KB 5/4/2001 1:05:02 PM mshtml.dll 5.0.3315.2870 MIDI Sequence mplay32.exe/mid C:\WINNT\system32 Microsoft Corporation Sound Not Available msjava.dll 5.0.3802.0 923 KB 5/4/2001 1:05:02 PM C:\WINNT\system32 Media Clip Microsoft Corporation Not Available Image Document "C:\Program Files\Windows msoss.dll <File Missing> Not Available Not Available NT\Accessories\ImageVue\KodakImg.exe" Not Available Not Available WordPad Document "%ProgramFiles%\Windows msxml.dll 8.0.5718.1 493 KB 5/4/2001 1:05:02 PM C:\WINNT\system32 NT\Accessories\WORDPAD.EXE" Microsoft Corporation Windows Media Services DRM Storage object Not Available occache.dll 5.0.3103.1000 86 KB 5/4/2001 1:05:02 PM C:\WINNT\system32 Microsoft Corporation Bitmap Image mspaint.exe ole32.dll 5.0.2195.2887 970 KB 5/4/2001 1:05:02 PM [Internet Explorer 5] C:\WINNT\system32 Microsoft Corporation oleaut32.dll 2.40.4517.0 612 KB 5/4/2001 1:05:02 PM [Following are sub-categories of this main category] C:\WINNT\system32 Microsoft Corporation olepro32.dll 5.0.4517.0 160 KB 5/4/2001 1:05:02 PM C:\WINNT\system32 Microsoft Corporation [Summary] rsabase.dll 5.0.2195.2228 128 KB 5/4/2001 1:05:02 PM Value C:\WINNT\system32 Microsoft Corporation Item rsaenh.dll 5.0.2195.2228 Version 5.00.3315.1000 131 KB 5/4/2001 1:05:02 PM 53315.1000 C:\WINNT\system32 Microsoft Corporation Build Product ID 51876-270-8956491-05798 rsapi32.dll <File Missing> Not Available Not Available C:\Program Files\Internet Explorer Not Available Not Available Application Path Language English (United States) rsasig.dll <File Missing> Not Available Not Available Not Available Active Printer Not Available Not Available schannel.dll 5.1.2195.0 138 KB 5/4/2001 1:05:02 PM Cipher Strength 168-bit C:\WINNT\system32 Microsoft Corporation Disabled shdoc401.dll <File Missing> Content Advisor Not Available Not **IEAK Install** Available Not Available Not Available No shdocvw.dll 5.0.3315.2879 1078 KB 5/4/2001 1:05:02 PM [File Versions] C:\WINNT\system32 Microsoft Corporation shell32.dll 5.0.3315.2902 2304 KB 5/4/2001 1:05:02 PM C:\WINNT\system32 Microsoft Corporation File Version Size Date Path Company 5.0.2195.2867 advapi32.dll 5/4/2001 1:05:02 PM shlwapi.dl15.0.3315.1000 5/4/2001 1:05:02 PM 352 KB 283 KB C:\WINNT\system32 Microsoft Corporation C:\WINNT\system32 Microsoft Corporation advpack.dll 5.0.3103.1000 87 KB 5/4/2001 1:05:02 PM 5.0.2920.0 82 KB 12/7/1999 8:00:00 AM C:\WINNT\system32 Microsoft Corporation C:\WINNT\system32 Microsoft Corporation browselc.dll 5.0.3315.2846 5/4/2001 1:05:02 PM urlmon.dll 5.0.3315.1000 441 KB 5/4/2001 1:05:02 PM 35 KB C:\WINNT\system32 Microsoft Corporation C:\WINNT\system32 Microsoft Corporation browseui.dll 5.0.3315.2846 789 KB 5/4/2001 1:05:02 PM vbscript.dll 5.1.0.5907 428 KB 5/4/2001 1:05:02 PM C:\WINNT\system32 Microsoft Corporation C:\WINNT\system32 Microsoft Corporation ckcnv.exe 5.0.2189.1 9 KB 12/7/1999 8:00:00 AM webcheck.dll 5.0.3315.1000 252 KB 5/4/2001 1:05:02 PM C:\WINNT\system32 Microsoft Corporation C:\WINNT\system32 Microsoft Corporation 5.81.3103.1000 win.com 5.0.2134.1 24 KB comctl32.dll 538 KB 5/4/2001 1:05:02 PM 12/7/1999 8:00:00 AM C:\WINNT\system32 Microsoft Corporation C:\WINNT\system32 Microsoft Corporation crypt32.dll 5.131.2195.2833 451 KB 5/4/2001 1:05:02 PM 457 KB 5/4/2001 1:05:02 PM wininet.dll 5.0.3315.1000 C:\WINNT\system32 Microsoft Corporation C:\WINNT\system32 Microsoft Corporation

winsock.dll 3.10.0.103 3 KB 12/7/1999 8:00:00 AM

C:\WINNT\system32 Microsoft Corporation

wintrust.dll 5.131.2195.2779 162 KB 5/4/2001 1:05:02 PM

C:\WINNT\system32 Microsoft Corporation

wsock.vxd <File Missing> Not Available Not Available

Not Available Not Available

wsock32.dll 5.0.2195.2871 21 KB 5/4/2001 1:05:02 PM

C:\WINNT\system32 Microsoft Corporation

wsock32n.dll <File Missing> Not Available Not

Available Not Available Not Available

[Connectivity]

Item Value

Connection Preference Never dial

EnableHttp1.1 1 ProxyHttp1.1 0

LAN Settings

AutoConfigProxy wininet.dll

AutoProxyDetectMode Enabled

AutoConfigURL
Proxy Disabled
ProxyServer
ProxyOverride

[Cache]

[Following are sub-categories of this main category]

[Summary]

Item Value

Page Refresh Type Automatic

Temporary Internet Files Folder C:\Documents and

Settings\Administrator\Local Settings\Temporary Internet Files

Total Disk Space 17351 MB Available Disk Space 13323 MB Maximum Cache Size 542 MB Available Cache Size 542 MB

[List of Objects]

Program File Status CodeBase No cached object information available

[Content]

[Following are sub-categories of this main category]

[Summary]

Item Value

Content Advisor Disabled

[Personal Certificates]

Issued To Issued By Validity Signature Algorithm

Administrator Administrator 2/10/2003 to 1/17/2103

sha1RSA

[Other People Certificates]

Issued To Issued By Validity Signature Algorithm No other people certificate information available

[Publishers]

Name

No publisher information available

[Security]

Zone Security Level

Local intranet Medium-low

Trusted sites Low

Internet Medium

Restricted sites High

RTE Input Parameters

Profile: 1530wh_15rte

File Path: C:\Program Files\BenchCraft\1530wh_15rte.pro

Version: 3

Number of Engines: 15

Name: rte111 Description: rte111

Directory: c:\rtelogs\rte111.log

Machine: rtes2

Parameter Set: PARAM2

Index: 0 Seed: 25744

Configured Users: 1020

Pipe Name: DRIVER11571562

Connect Rate: 400 Start Rate: 0 Max. Concurrency: -1

Concurrency Rate: 10 CLIENT NURAND: 233

CPU: 0

Name: rte112 Description: rte112

Directory: c:\rtelogs\rte112.log

Machine: rtes2 Parameter Set: PARAM2

Index: 3000000

Seed: 25744

Configured Users: 1020

Pipe Name: DRIVER25384421

Connect Rate: 400 Start Rate: 0 Max. Concurrency: -1 Concurrency Rate: 10 CLIENT_NURAND: 233

CPU: 1

Name: rte113 Description: rte113

Directory: c:\rtelogs\rte113.log

Machine: rtes2

Parameter Set: PARAM2

Index: 6000000 Seed: 25744

Configured Users: 1020 Pipe Name: DRIVER35434750

Connect Rate: 400 Start Rate: 0 Max. Concurrency: -1 Concurrency Rate: 10 CLIENT_NURAND: 233

CPU: 2

Name: rte114 Description: rte114

Directory: c:\rtelogs\rte114.log

Machine: rtes2

Parameter Set: PARAM2

Index: 9000000 Seed: 25744

Configured Users: 1020

Pipe Name: DRIVER45470859

Connect Rate: 400 Start Rate: 0 Max. Concurrency: -1 Concurrency Rate: 10 CLIENT NURAND: 233

CPU: 3

Name: rte121 Description: rte121

Directory: c:\rtelogs\rte121.log

Machine: rtes2

Parameter Set: PARAM2 Index: 15000000 Seed: 25744

Configured Users: 1020

Pipe Name: DRIVER55546265

Connect Rate: 400 Start Rate: 0 Max. Concurrency: -1 Concurrency Rate: 10 CLIENT NURAND: 233

CPU: 0

Name: rte122 Description: rte122

Directory: c:\rtelogs\rte122.log

Machine: rtes2

Parameter Set: PARAM2 Index: 18000000

Seed: 25744

Configured Users: 1020 Pipe Name: DRIVER65583640

Connect Rate: 400 Start Rate: 0 Max. Concurrency: -1 Concurrency Rate: 10 CLIENT NURAND: 233

CPU: 1

Name: rte123 Description: rte123

Directory: c:\rtelogs\rte123.log

Machine: rtes2

Parameter Set: PARAM2

Index: 21000000 Seed: 25744

Configured Users: 1020

Pipe Name: DRIVER75623609

Connect Rate: 400 Start Rate: 0 Max. Concurrency: -1 Concurrency Rate: 10 CLIENT_NURAND: 233

CPU: 2

Name: rte124 Description: rte124

Directory: c:\rtelogs\rte124.log

Machine: rtes2

Parameter Set: PARAM2

Index: 24000000 Seed: 25744

Configured Users: 1020

Pipe Name: DRIVER85670968

Connect Rate: 400 Start Rate: 0 Max. Concurrency: -1 Concurrency Rate: 10 CLIENT_NURAND: 233

CPU: 3

Name: rte132

Description: rte132

Directory: c:\rtelogs\rte132.log

Machine: frte90 Parameter Set: PARAM2 Index: 27000000 Seed: 25744

Configured Users: 1020

Pipe Name: DRIVER95713843

Connect Rate: 400 Start Rate: 0 Max. Concurrency: -1 Concurrency Rate: 10 CLIENT NURAND: 233

CPU: 0

Name: rte133 Description: rte133

Directory: c:\rtelogs\rte133.log

Machine: frte90

Parameter Set: PARAM2

Index: 30000000 Seed: 25744

Configured Users: 1020

Pipe Name: DRIVER105754984

Connect Rate: 400 Start Rate: 0 Max. Concurrency: -1 Concurrency Rate: 10 CLIENT NURAND: 233

CPU: 1

Name: rte134

Description: rte134

Directory: c:\rtelogs\rte134.log

Machine: frte90 Parameter Set: PARAM2

Index: 33000000 Seed: 25744

Configured Users: 1020

Pipe Name: DRIVER115821125

Connect Rate: 400 Start Rate: 0 Max. Concurrency: -1 Concurrency Rate: 10 CLIENT NURAND: 233

CPU: 0

Name: rte135 Description: rte135

Directory: c:\rtelogs\rte135.log

Machine: frte90 Parameter Set: PARAM2 Index: 72000000

Seed: 25744

Configured Users: 1020

Pipe Name: DRIVER2568868468

Connect Rate: 400 Start Rate: 0 Max. Concurrency: -1

Concurrency Rate: 10 CLIENT NURAND: 233

CPU: 1

Name: rte141 Description: rte141

Directory: c:\rtelogs\rte141.log

Machine: frte90

Parameter Set: PARAM2

Index: 75000000 Seed: 25744

Configured Users: 1020

Pipe Name: DRIVER2668925937

Connect Rate: 400 Start Rate: 0 Max. Concurrency: -1 Concurrency Rate: 10 CLIENT_NURAND: 233

CPU: 1

Name: rte142 Description: rte142

Directory: c:\rtelogs\rte142.log

Machine: frte90

Parameter Set: PARAM2

Index: 78000000 Seed: 25744

Configured Users: 1020

Pipe Name: DRIVER2768987671

Connect Rate: 400 Start Rate: 0 Max. Concurrency: -1 Concurrency Rate: 10 CLIENT_NURAND: 233

CPU: 0

Name: rte143 Description: rte143

Directory: c:\rtelogs\rte143.log

Machine: frte90 Parameter Set: PARAM2

Index: 81000000 Seed: 25744

Configured Users: 1020

Pipe Name: DRIVER2869074843

Connect Rate: 400 Start Rate: 0 Max. Concurrency: -1 Concurrency Rate: 10 CLIENT_NURAND: 233 CPU: 1

Number of User groups: 15

Driver Engine: rte111 IIS Server: client11 SQL Server: ibmserver3 Database: tpcc User: sa Protocol: HTML w id Range: 1 - 102 w_id Min Warehouse: 1 w id Max Warehouse: 1530 Scale: Normal User Count: 1020 District id: 1 Scale Down: No

Driver Engine: rte112

IIS Server: client11 SQL Server: ibmserver3 Database: tpcc User: sa Protocol: HTML w id Range: 103 - 204 w id Min Warehouse: 1 w_id Max Warehouse: 1530

Scale: Normal User Count: 1020 District id: 1 Scale Down: No

Driver Engine: rte113 IIS Server: client11 SQL Server: ibmserver3 Database: tpcc User: sa Protocol: HTML w id Range: 205 - 306 w id Min Warehouse: 1 w id Max Warehouse: 1530 Scale: Normal User Count: 1020 District id: 1

Driver Engine: rte114 IIS Server: client11 SQL Server: ibmserver3 Database: tpcc User: sa Protocol: HTML w id Range: 307 - 408

w id Min Warehouse: 1 w id Max Warehouse: 1530

Scale Down: No

Scale: Normal User Count: 1020 District id: 1 Scale Down: No

Driver Engine: rte121 IIS Server: client12 SQL Server: ibmserver3 Database: tpcc User: sa Protocol: HTML w_id Range: 409 - 510 w id Min Warehouse: 1 w id Max Warehouse: 1530 Scale: Normal

User Count: 1020 District id: 1 Scale Down: No

IIS Server: client12 SQL Server: ibmserver3 Database: tpcc User: sa Protocol: HTML w_id Range: 511 - 612

Driver Engine: rte122

w_id Min Warehouse: 1 w_id Max Warehouse: 1530 Scale: Normal User Count: 1020

District id: 1 Scale Down: No

Driver Engine: rte123 IIS Server: client12 SQL Server: ibmserver3

Database: tpcc User: sa Protocol: HTML w_id Range: 613 - 714 w_id Min Warehouse: 1 w id Max Warehouse: 1530

Scale: Normal User Count: 1020 District id: 1 Scale Down: No

Driver Engine: rte124 IIS Server: client12

SQL Server: ibmserver3
Database: tpcc

User: sa Protocol: HTML w_id Range: 715 - 816 w_id Min Warehouse: 1 w_id Max Warehouse: 1530

Scale: Normal User Count: 1020 District id: 1 Scale Down: No

Driver Engine: rte132 IIS Server: client13 SQL Server: ibmserver3

Database: tpcc User: sa Protocol: HTML w_id Range: 817 - 918 w_id Min Warehouse: 1 w id Max Warehouse: 1530

Scale: Normal User Count: 1020 District id: 1 Scale Down: No

Driver Engine: rte133 IIS Server: client13 SQL Server: ibmserver3 Database: tpcc

User: sa Protocol: HTML w_id Range: 919 - 1020 w_id Min Warehouse: 1 w_id Max Warehouse: 1530

Scale: Normal User Count: 1020 District id: 1 Scale Down: No

Driver Engine: rte134 IIS Server: client13 SQL Server: ibmserver3

Database: tpcc User: sa Protocol: HTML w_id Range: 1021 - 1122 w_id Min Warehouse: 1 w_id Max Warehouse: 1530 Scale: Normal User Count: 1020 District id: 1

Scale Down: No

Driver Engine: rte135 IIS Server: client13 SQL Server: ibmserver3 Database: tpcc User: sa Protocol: HTML w_id Range: 1123 - 1224 w_id Min Warehouse: 1 w_id Max Warehouse: 1530 Scale: Normal

Scale Down: No
Driver Engine: rte141
IIS Server: client14
SQL Server: ibmserver3

User Count: 1020 District id: 1

Database: tpcc User: sa Protocol: HTML w_id Range: 1225 - 1326 w_id Min Warehouse: 1 w_id Max Warehouse: 1530

Scale: Normal User Count: 1020 District id: 1 Scale Down: No

Driver Engine: rte142
IIS Server: client14
SQL Server: ibmserver3
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 1327 - 1428
w_id Min Warehouse: 1
w_id Max Warehouse: 1530
Scale: Normal
User Count: 1020
District id: 1
Scale Down: No

Driver Engine: rte143
IIS Server: client14
SQL Server: ibmserver3
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 1429 - 1530
w_id Min Warehouse: 1
w_id Max Warehouse: 1530
Scale: Normal
User Count: 1020
District id: 1
Scale Down: No

Number of Parameter Sets: 2

~Default Default Parameter Set						
		Txn		Vov	рт рт	Manu
					RT RT	
				e Time		nce Delay
		New Order	10.00	12.05	18.01	0.10
5.00	0.10	D	10.00	12.05	2.01	0.10
5.00	0.10	Payment	10.00	12.05	3.01	0.10
5.00	0.10	Delivery	1.00	5.05	2.01	0.10
5.00	0.10	Delivery	1.00	3.03	2.01	0.10
3.00	0.10	Stock Level	1.00	5.05	2.01	0.10
20.00	0.10	Stock Level	1.00	3.03	2.01	0.10
20.00	0.10	Order Status	1.00	10.05	2.01	0.10
5.00	0.10	Order Status	1.00	10.03	2.01	0.10
		PARAM2				
		Performance p	aramete	rs		
		Txn		Key	RT RT	Menu
		Weight Time Time Delay Fence Dela				
		New Order	_	12.05	18.01	0.10
5.00	0.10	Tiem order	10.00	12.00	10.01	0.10
		Payment	9.61	12.05	3.01	0.10
5.00	0.10	,				
		Delivery	0.90	5.05	2.01	0.10
5.00	0.10	,				
		Stock Level	0.90	5.05	2.01	0.10
20.00	0.10					
		Order Status	0.90	10.05	2.01	0.10
5.00	0.10					

Appendix D: 60-Day Space

		TPC-C 60-Day	/ Space Requiren	nents		
Warehouses	1,530				tpmC	18,936.05
						T
Table	Rows	Data KB	Index KB	Extra 5% KB	8HR Space	Total Space KB
Warehouse	1,530	168	32	10.00		210.00
District	15,300	1,704	32	86.80		1,822.80
Item	100,000	9,528	48	478.80		10,054.80
New-Order	13,770,000	217,712	512		122,400.00	
History	45,900,000	2,550,008	24		504,967.67	
Orders	45,900,000	1,406,904	639,784		405,293.45	
Customer	45,900,000	33,381,824	1,990,552	1,768,618.80		37,140,994.80
Order-Line	459,001,971	28,687,624	60,736		5,692,866.74	
Stock	153,000,000	48,960,000	91,480	2,452,574.00		51,504,054.00
Totals		115,215,472	2,783,200	4,221,768.40	6,725,527.87	128,945,968.27
Segment	LogDev Cnt.	Segment Size	Needed	Overhead		Not Needed
misc	3	43,008,000	40,300,919	403,009		2,304,071.34
big	3	89,702,400	88,645,049	•		170,900.71
master, msdb,model	1	13,312	13,312			-
tpcc_root	1	8,192	8,192			-
tempdb	1	8,704	8,704			-
Totals		132,740,608.00	128,976,176.27	1,289,459.68		2,474,972.05
		, ,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,	-,,	,,	I	, , , ,
Dynamic Space			der, Order-Line and H			
Static Space			Space + Overhead - D			
Free Space			- Dynamic Space - S	tatic Space - Not ne	eded	
Daily Growth		(Dynamic Space/W				
Daily Spread	(2,940,869.09)	Free Space - 1.5 * D	oaily Growth (Zero If N	Negative)		
60-Day Space (KB)	478 729 562 31	Static Space + 60 (I	Daily Growth + Daily	Spread)		
60-Day Space (GB)			3 (Excludes OS,Pagin		;)	
Available (GB)			gured and available f			configuration
Log File Storage Requ						
		T				
Log Size (MB)		Total Size of Log Fi				
% Log Used		% of Log File Used		during Entire Dur		
Total N-O Txn			-Order Transactions	uuring Entire Kun		
Log / N-O Txn 8 Hour Log (GB)		KB of Log per New	-Order Transaction BB (Excluding Space	for Dodundanas		
Log Configured (GB)		o Hours of Log In C	DE (Excluding Space)	ioi Reduildancy)		1
Disk Capacity	67.72 MB	GB				
18.2GB 10K RPM	17,736	16.93				
73.4GB 10K RPM	70,944	67.72				
Space Usage	GB Needed		Disks Priced	Disk Size	GB Priced	GB Usable
60-Day (RAID-0)	456.55		42	18.2GB	711.06	711.06
,	_				Total DB	711.06
8hr Log (RAID-1)	40.09		2	73.4GB	135.44	
					Total Log	67.72
OS, SQL Server	4.00		1	18.2GB	16.93	16.93
Total Space			45			
TOTAL SPACE	500.65		45		863.43	795.71

Appendix E: Third-Party Quotations	

Microsoft Corporation One Microsoft Way Redmond, WA 98052-6399 Tel 425 882 8080 Fax 425 936 7329 http://www.microsoft.com/

Microsoft

July 9, 2003

IBM Corporation Chris King 3039 Cornwallis Road Research Triangle Park, NC 27709

Ms. King:

Here is the information you requested regarding pricing for several Microsoft products to be used in conjunction with your TPC-C benchmark testing.

All pricing shown is in US Dollars (\$).

Part Number	Description	Unit Price	Quantity	Price
228-01079	SQL Server 2000 Standard Edition Per processor licensing No discounts applied	\$4,999	1	\$4,999
C11-00821	Windows 2000 Server Server license only - No CALs Discount Schedule: Open Program - No Level Unit Price reflects a 8% discount from the retail unit price of \$799.	\$738	1	\$738
P73-00295	Windows Server 2003, Standard Server Server license only - No CALs Discount Schedule: Open Program - No Level Unit Price reflects a 26% discount from the retail unit price of \$999.	\$738	1	\$738
254-00170	Visual C++ Standard No discounts applied	\$109	1	\$109
PRO-PRORS-16U-01	Database Server Support Package 1 Year Term	\$1,950	3	\$5,850

Some products may not be currently orderable but will be available through Microsoft's normal distribution channels by April 2, 2003.

This quote is valid for the next 90 days.

If we can be of any further assistance, please contact Jamie Reding at (425) 703-0510 or jamiere@microsoft.com.

Computer Giants Page 1 of 2



Crossover Page 1 of 1

Home **Network Cards** Network Cables & MISC Cat5e Crossover Cables Print Servers **Barcode Readers Extension Cables** Miscellaneous **WE ARE ANTI SPAM** Blacklisted Brands gaming Cables - Misc SCSI Cables & devices 6ft 4 wire black molded As low as 34 cents each network patch cable supports 10 / 100 mbps networks *Order quantities over 5 ONLY* Barcode Cables Fiber Optic Networking **Network Cables** Networking Power Print serverss Printing SCSI Software Storage

Privacy Policy
Info &
Shipping Notes
& Ways to delay
Processing of order
Search
Index

Switch Boxes

Show Order

Y? SHOPPING

LanAdapters.com

Crossover

7Ft Category 5 cross over Cable RJ45/RJ45 PC To PC Cat5
 Hooks 2 PCs Together W/O Hub 10bt or 100bt Supports Fast Ethernet RJ-45 tw
 crossover

CBLC5C7 Regular price: \$2.00 Sale price: \$1.00 Order

25ft Category 5 Cross over cable TESTED RJ45/RJ45Lan C
 Cat 5 Hooks up 2 PCs together W/O hub molded 10bt or 100bt Supports Fast E twisted pair 8 colors

CBLC5C25 Regular price: \$8.00 Sale price: \$3.00 Order