

**TPC Benchmark™ C**  
**Full Disclosure Report**  
**for**  
**IBM @server xSeries 370**  
**using**  
**Microsoft SQL Server 2000**  
**and**  
**Microsoft Windows 2000 Datacenter Server**

**Submitted for Review**  
**March 23, 2001**



## **First Edition - March 2001**

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 2001. 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 is a registered trademark, and the e-business logo, xSeries, Netfinity and ServeRAID are trademarks of International Business Machines Corporation.

The following terms used in this publication are trademarks of other companies as follows: TPC Benchmark, TPC-C and tpmC are trademark of Transaction Processing Performance Council; Intel, Pentium and Xeon are registered trademarks of Intel Corporation; Microsoft, Windows 2000, 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***

<sup>1</sup> MHz only measures microprocessor internal clock speed, not application performance. Many factors affect application performance.

<sup>2</sup> In the context of the TPC-C benchmark reported in this document, 1GB equals 1024\*1024MB. The reason for calculating GB in this way is to maintain compatibility with the method Windows' logical disk manager uses to report storage.

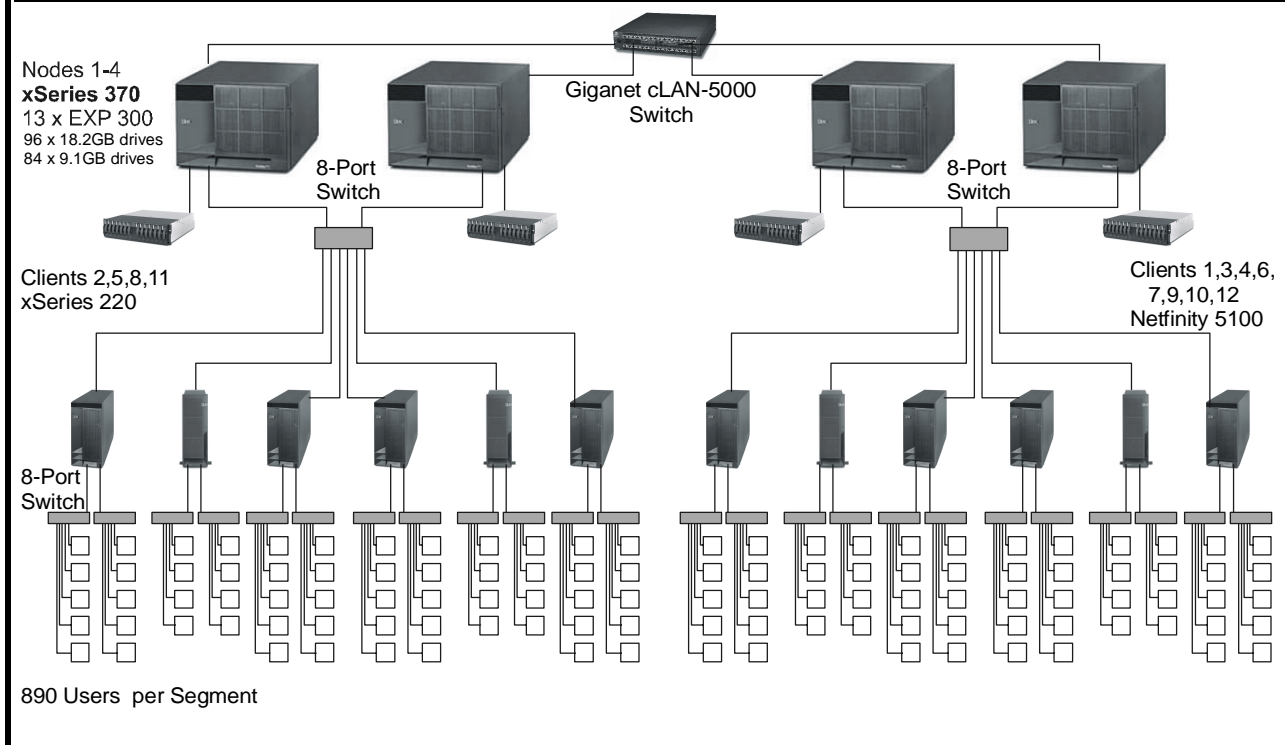


**IBM @server xSeries 370  
with SQL Server 2000**

TPC-C Rev. 3.5

Report Date: Mar. 23, 2001

Total System Cost		TPC-C Throughput	Price/Performance	Availability Date
<b>\$3,090,250</b>		<b>121,319.23 tpmC</b>	<b>\$25.47 / tpmC</b>	<b>May 31, 2001</b>
Processors	Database Manager	Operating System	Other Software	Number of Users
<b>32 Intel<sup>(R)</sup> Pentium<sup>(R)</sup> III Xeon™ 900MHz 2MB L2 Cache</b>	<b>Microsoft<sup>(R)</sup> SQL Server 2000</b>	<b>Microsoft Windows<sup>(R)</sup> 2000 Datacenter Server</b>	<b>Microsoft COM+ Microsoft Visual C++ 6.0</b>	<b>99,680</b>



System Component	Qty	Each of 4 Server Nodes	Qty	Each of 8 Clients:
Processors	8	900MHz Pentium III Xeon	2	733MHz Pentium III
Cache		w/2MB L2 Cache		w/256KB L2 Cache
Memory	32	1GB ECC SDRAM	4	128MB
Disk Controllers	7	ServeRAID-4H Ultra160 SCSI Adapter	1	Ultra160 SCSI Onboard
Disk Drives	84	9.1GB (15000 rpm)	1	9.1GB (10000 rpm)
	96	18.2GB (15000 rpm)		
	2	9.1GB (10000 rpm)		
Total Storage	1	9.2TB (8.77TB online)	2	Each of 4 Clients:
Other			1	933MHz/256KB Pentium III
Tape Drive	1	3502-R14 DLT Tape Library	1	128MB ECC SDRAM
		Giganet cLAN-1000 Host Adapter	2	256MB ECC SDRAM
Interconnect	1	Giganet cLAN-5000 Switch	1	Ultra160 SCSI Onboard
			1	9.1GB (10000 rpm)



@server xSeries 370 c/s  
with Microsoft SQL Server  
2000

TPC-C Revision 3.5

Report Date: March 23, 2001

Description	Order Number	Third-Party Brand	Pricing	Unit Price	Qty	Ext. Price	5-Yr. Maint.*
<b>Server Hardware</b>							
xSeries 370 900MHz/2MB Pentium III Xeon*	8681-3RX	IBM	1	\$19,114	4	\$76,456	\$42,740
900MHz/2MB L2 Cache Processor Upgrade	19K4637	IBM	1	6,599	28	184,772	0
8500R Memory Expansion Card	28L4454	IBM	1	625	4	2,500	0
8500 >4X Accelerator Filter	10K2335	IBM	1	1,249	4	4,996	0
1GB SDRAM RDIMM Memory Kit	33L3056	IBM	1	2,596	128	332,288	0
9.1GB 10K Ultra160 SCSI Drive	37L7204	IBM	1	299	8	2,392	0
ServeRAID-4H Ultra160 SCSI Adapter	37L6889	IBM	1	2,439	28	68,292	0
Ultra2 SCSI 4m Cable	03K9311	IBM	1	105	104	10,920	0
10/100 Ethernet Server Adapter with IPsec	06P3601	IBM	1	99	4	396	0
E54 15" (13.8" Viewable) Color Monitor*	6331N2N	IBM	1	173	4	692	1,120
3502-R14 DLT Tape Autoloader*	3502R14	IBM	1	13,779	1	13,779	3,844
IBM Smart-UPS Model 5000RMB	37L6861	IBM	1	3,299	4	13,196	0
Netfinity Rack*	9306900	IBM	1	1,725	6	10,350	7,848
Side Panel Kit	94G6669	IBM	1	195	1	195	0
Hardware installation and prep fees		IBM	1			21,180	0
Giganet cLAN-1000 Host Adapter (incl. 10%)	cLAN-1000	Giganet	3	795	6	4,770	75,000
Giganet cLAN5000 8-Port Switch (incl. 10%)	cLAN5000	Giganet	3	6,250	3	18,750	incl. above
Giganet cLAN-A1011 10M Cable (incl. 10%)	cLAN-A1011	Giganet	3	135	6	810	incl. above
<b>Storage Hardware</b>							
Netfinity EXP300 Rack Storage Enclosure*	35311RU	IBM	1	3,179	52	165,308	31,512
9.1GB 15K Ultra160 SCSI Drive	19K0655	IBM	1	601	336	201,936	0
18.2GB 15K Ultra160 SCSI Drive	19K0656	IBM	1	789	384	302,976	0
<b>Subtotal</b>						<b>\$1,436,954</b>	<b>\$162,064</b>
<b>Server Software</b>							
Datacenter OS Preload Kit/Ship Group/Warr.	22P4745	IBM	1			\$702,000	0
Microsoft SQL Server 2000	11K7641	Microsoft	2	16,541	32	539,312	0
5-Year Maintenance for Software		Microsoft	2	2,095	20		\$41,900
<b>Subtotal</b>						<b>\$1,231,312</b>	<b>\$41,900</b>
<b>Client Hardware</b>							
Netfinity 5100 / 733MHz/256KB Pentium III*	8658-2RY	IBM	1	2,052	8	\$16,416	\$47,240
733MHz/256KB Pentium III Upgrade	00N7943	IBM	1	399	8	3,192	0
128MB DIMMs	33L3123	IBM	1	185	24	4,440	0
xSeries 220 / 933MHz/256KB Pentium III*	8645-41X	IBM	1	1,593	4	6,372	12,592
933MHz/256KB Pentium III Upgrade	10K3819	IBM	1	999	4	3,996	0
256MB ECC SDRAM RDIMM	33L3144	IBM	1	365	8	2,920	0
9.1GB 10K Ultra160 SCSI Drive	37L7204	IBM	1	299	12	3,588	0
Intel Pro/100+ Dual-Port Ethernet Adapter**	8472	Intel	1	219	12	2,628	0
E54 15" (13.8" Viewable) Color Monitor*	6331N2N	IBM	1	173	12	2,076	3,360
<b>Subtotal</b>						<b>\$45,628</b>	<b>\$63,192</b>
<b>Client Software</b>							
Microsoft Windows 2000 Server with COM+	C11-00821	Microsoft	2	738	12	8,856	incl. above
Microsoft Visual C++ Professional 6.0	048-00317	Microsoft	2	549	1	549	incl. above
<b>Subtotal</b>						<b>\$9,405</b>	<b>\$0</b>
<b>User Connectivity</b>							
8-Port 10/100Mbps Hub** (10% spares***)	DEH2924	Generic	4	23	13,799	\$317,377	\$0
8-Port 10/100 Nway Fast Ethernet Switch (+!0% )	NX-DSS8		4	25	30	725	0
<b>Subtotal</b>						<b>\$318,102</b>	<b>\$0</b>
<b>Total</b>						<b>\$3,041,401</b>	<b>\$267,156</b>
<b>Discount</b>						<b>(218,307)</b>	<b>\$0</b>
<b>Grand Total</b>						<b>\$2,823,094</b>	<b>\$267,156</b>

Large volume discount; prices will vary if purchase individually.

Notes: \* The standard 3-year warranty and the extended warranty on IBM hardware is for 7x24, 4-hour response. \*\* Five-year warranty. \*\*\* 10% or minimum 2 spares are added in place of on-site service (products have a 5-year return-to-vendor warranty)

Pricing: 1 - IBM; 2 - Microsoft; 3 - Giganet; 4 - Software House International

Audited by Brad Askins of InfoSizing, Inc.

**Five-Year Cost of Ownership:** \$3,090,250

**tpmC Rating:** 121,319.23

**\$/tpmC:** \$25.47

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.

<b>Numerical Quantities Summary</b>			
MQTh, Computed Maximum Qualified Throughput: % throughput difference, reported and reproducibility runs:			121,319.23 tpmC 0.7 %
<b>Response Times (in seconds)</b>	<b>90%</b>	<b>Average</b>	<b>Maximum</b>
New-Order	1.30	0.78	24.82
Payment	1.30	0.77	26.10
Delivery (Interactive)	0.83	0.33	2.19
Stock-Level	1.10	0.57	6.11
Order Status	1.03	0.51	6.42
Delivery (Deferred)	0.46	0.34	6.14
Menu	0.85	0.34	3.88
<b>Transaction Mix (in percent of total transactions)</b>		<b>Total Occurrences</b>	<b>Percent</b>
New-Order		3,639,577	44.83
Payment		3,493,410	43.03
Delivery		328,099	4.04
Stock Level		328,584	4.05
Order Status		329,461	4.06
<b>Emulation Delay (in seconds)</b>	<b>Response Time</b>		<b>Menu</b>
New-Order	0.1		0.1
Payment	0.1		0.1
Order-Status	0.1		0.1
Delivery	0.1		0.1
Stock-Level	0.1		0.1
<b>Keying/Think Times (in seconds)</b>	<b>Minimum</b>	<b>Average</b>	<b>Maximum</b>
New-Order	18.00 / 0.00	18.01 / 12.04	18.07 / 120.51
Payment	3.00 / 0.00	3.01 / 12.05	3.07 / 120.51
Delivery	2.00 / 0.00	2.01 / 5.06	2.08 / 50.51
Stock Level	2.00 / 0.00	2.01 / 5.06	2.05 / 50.50
Order Status	2.00 / 0.00	2.01 / 10.07	2.05 / 100.51
<b>Test Duration</b>			
Ramp-up time			1 hour 6 minutes 21 seconds
Measurement interval			30 minutes
Number of transactions (all types) completed in measurement interval			8,447,190
Ramp-down time			3 minutes 1 seconds
Number of checkpoints in measurement interval			1
Checkpoint interval			30 minutes

## Table of Contents

<b>Preface</b> .....	10
<b>General Items</b> .....	14
Application Code and Definition Statements .....	14
Benchmark Sponsor .....	14
Parameter Settings .....	14
Configuration Diagrams .....	14
<i>Measured Configuration</i> .....	15
<i>Priced Configuration</i> .....	16
<b>Clause 1: Logical Database Design Related Items</b> .....	17
Table Definitions .....	17
Physical Organization of the Database .....	17
Insert and Delete Operations .....	17
Horizontal or Vertical Partitioning .....	17
Replication .....	17
Table Attributes .....	17
<b>Clause 2: Transaction and Terminal Profiles Related Items</b> .....	18
Random Number Generation .....	18
Screen Layout .....	18
Terminal Verification .....	18
Intelligent Terminals .....	18
Transaction Profiles .....	18
Deferred Delivery Mechanism .....	19
<b>Clause 3: Transaction and System Properties Related Items</b> .....	20
Atomicity Requirements .....	20
<i>Completed Transactions</i> .....	20
<i>Aborted Transactions</i> .....	20
Consistency Requirements .....	20
Isolation Requirements .....	21
<i>Loss of Data Test</i> .....	21
<i>Combined Loss of Log and Loss of System Test (Instantaneous Interruption and Loss of Memory)</i> .....	22
<i>Loss of Interconnect</i> .....	22
<b>Clause 4: Scaling and Database Population Related Items</b> .....	23
Cardinality of Tables .....	23
Distribution of Tables and Logs .....	23
Database Model Implemented .....	24
Partitions/Replications Mapping .....	25
180-Day Space Requirement .....	25
<b>Clause 5: Performance Metrics and Response Time Related Items</b> .....	26
Measured tpmC .....	26
Response Times .....	26
Keying/Think Times .....	26
Response Time Frequency Distribution Curves .....	27
Work Performed during Steady State .....	31
<i>Transaction Flow</i> .....	31
Checkpoints .....	31
Reproducibility Methodology .....	31
Measurement Interval .....	31
Transaction Mix .....	32
Percentage of Total Mix .....	32
<b>Clause 6: SUT, Driver and Communication Definition Related Items</b> .....	34
Description of RTE .....	34
Emulated Components .....	34

Benchmarked and Targeted System Configuration Diagrams .....	34
Network Configuration .....	34
Network Bandwidth .....	35
Operator Intervention .....	35
<b>Clause 7: Pricing Related Items</b> .....	36
Hardware and Software Components .....	36
Availability Date .....	36
Measured tpmC .....	36
Country-Specific Pricing .....	36
Usage Pricing .....	36
System Pricing .....	37
<b>Clause 9: Audit Related Items</b> .....	38
Auditor .....	38
Availability of the Full Disclosure Report .....	38
<i>Attestation letter</i> .....	39
<b>Appendix A: Source Code</b> .....	41
Web Client Source code .....	41
<i>dlldata.c</i> .....	41
<i>error.h</i> .....	41
<i>install.c</i> .....	43
<i>install.h</i> .....	52
<i>install.rc</i> .....	52
<i>install_com.cpp</i> .....	55
<i>install_resource.h</i> .....	59
<i>isapi_dll_resource.h</i> .....	59
<i>license.txt</i> .....	59
<i>methods.h</i> .....	61
<i>Readregistry.cpp</i> .....	63
<i>ReadRegistry.h</i> .....	64
<i>ReadWHouse.cpp</i> .....	64
<i>ReadWHouse.h</i> .....	67
<i>rtetime.h</i> .....	67
<i>SetAuditTime.cpp</i> .....	68
<i>spinlock.h</i> .....	70
<i>tpcc.cpp</i> .....	71
<i>tpc.def</i> .....	96
<i>tpcc.h</i> .....	96
<i>tpcc.rc</i> .....	98
<i>tpcc_com.cpp</i> .....	99
<i>tpcc_com.h</i> .....	101
<i>tpcc_com_all.cpp</i> .....	103
<i>tpcc_com_all.def</i> .....	108
<i>tpcc_com_all.h</i> .....	108
<i>tpcc_com_all.idl</i> .....	109
<i>tpcc_com_all.rc</i> .....	110
<i>tpcc_com_all.rgs</i> .....	111
<i>tpcc_com_all_i.c</i> .....	111
<i>tpcc_com_no.rgs</i> .....	112
<i>tpcc_com_os.rgs</i> .....	112
<i>tpcc_com_pay.rgs</i> .....	112
<i>tpcc_com_ps.def</i> .....	113
<i>tpcc_com_ps.h</i> .....	113
<i>tpcc_com_ps.idl</i> .....	115
<i>tpcc_com_ps_i.c</i> .....	115
<i>tpcc_com_ps_p.c</i> .....	116

<i>tpcc_com_rem.cpp</i>	127
<i>tpcc_com_rem.h</i>	128
<i>tpcc_com_remote.cpp</i>	130
<i>tpcc_com_remote.def</i>	135
<i>tpcc_com_remote.h</i>	135
<i>tpcc_com_remote.idl</i>	135
<i>tpcc_com_remote.rc</i>	136
<i>tpcc_com_remote.rgs</i>	137
<i>tpcc_com_remote_i.c</i>	137
<i>tpcc_com_remote_Methods.h</i>	137
<i>tpcc_com_remote_ps.def</i>	139
<i>tpcc_com_remote_ps.h</i>	139
<i>tpcc_com_remote_ps.idl</i>	141
<i>tpcc_com_remote_ps_i.c</i>	141
<i>tpcc_com_remote_ps_p.c</i>	142
<i>tpcc_com_remote_resource.h</i>	153
<i>tpcc_com_sl.rgs</i>	153
<i>tpcc_odbc.cpp</i>	153
<i>tpcc_odbc.h</i>	162
<i>trans.h</i>	164
<i>txn_base.h</i>	166
<i>txnlog.h</i>	166
Stored Procedures	169
<i>delivery.sql</i>	169
<i>neword.sql</i>	170
<i>ordstat.sql</i>	173
<i>payment.sql</i>	174
<i>stocklev.sql</i>	176
<b>Appendix B: Database Design</b>	178
Database Build	178
<i>4x2492</i>	178
<i>add_constraints_t1.sql</i>	178
<i>add_constraints_t2.sql</i>	178
<i>add_constraints_t3.sql</i>	179
<i>add_constraints_t4.sql</i>	179
<i>add_views1.sql</i>	180
<i>add_views2.sql</i>	181
<i>add_views3.sql</i>	181
<i>add_views4.sql</i>	183
<i>item_iot1.sql</i>	183
<i>item_iot2.sql</i>	184
<i>item_iot3.sql</i>	184
<i>item_iot4.sql</i>	184
<i>backup.sql</i>	185
<i>backupdev.sql</i>	185
<i>createdb.sql</i>	185
<i>dbopt1.sql</i>	186
<i>dbopt2.sql</i>	186
<i>idxcuscl.sql</i>	187
<i>idxcusnc.sql</i>	187
<i>idxdiscl.sql</i>	187
<i>idxitmcl.sql</i>	188
<i>idxnodcl.sql</i>	188
<i>idxodlcl.sql</i>	188
<i>idxordcl.sql</i>	188



<i>idxordnc.sql</i>	189
<i>idxstkcl.sql</i>	189
<i>idxwarcl.sql</i>	189
<i>tables.sql</i>	189
<i>restore.sql</i>	192
<i>removedb.sql</i>	192
<i>version.sql</i>	192
Load Source Code	193
<i>getargs.c</i>	193
<i>random.c</i>	195
<i>strings.c</i>	196
<i>time.c</i>	199
<i>tpcc.h</i>	199
<i>tpccldr.c</i>	200
<i>tpccldr.mak</i>	224
<b>Appendix C: Tunable Parameters</b>	229
Database Startup Parameters	229
Database Information	229
Software Update	230
Microsoft Windows 2000 Datacenter Configuration	230
<i>boot.ini</i>	230
<i>Enabling AWE</i>	230
<i>Enabling VIA</i>	230
<i>System Information Report for Server Nodes</i>	230
Client Configuration	256
<i>Transaction Monitor: COM+ Settings on Clients</i>	256
<i>Windows Registry Editor Version 5.00</i>	256
<i>System Information Report for Clients</i>	258
<i>Netfinity 5100</i>	258
RTE Input Parameters	297
<b>Appendix D: 180-Day Space</b>	330
<b>Appendix E: Third-Party Quotations</b>	331

---

## Preface

The TPC Benchmark™ 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 3.5, released October 25, 1999.

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 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.*

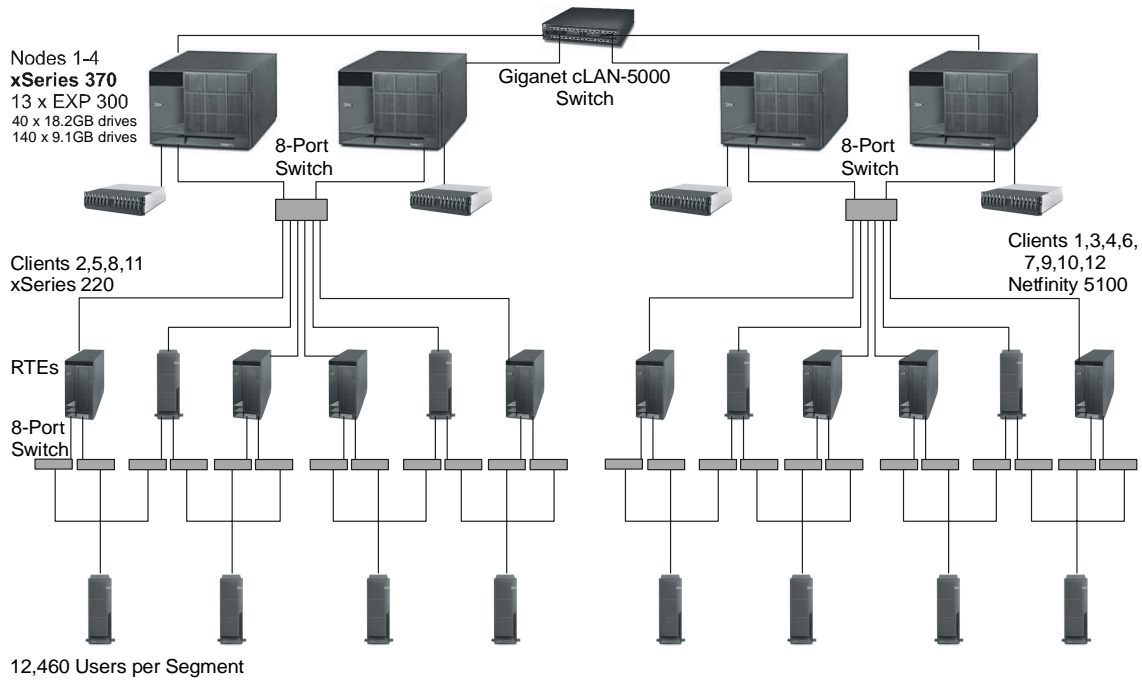
Appendix C contains the tunable parameters for the database, the operating system, and the transaction monitor.

### Configuration Diagrams

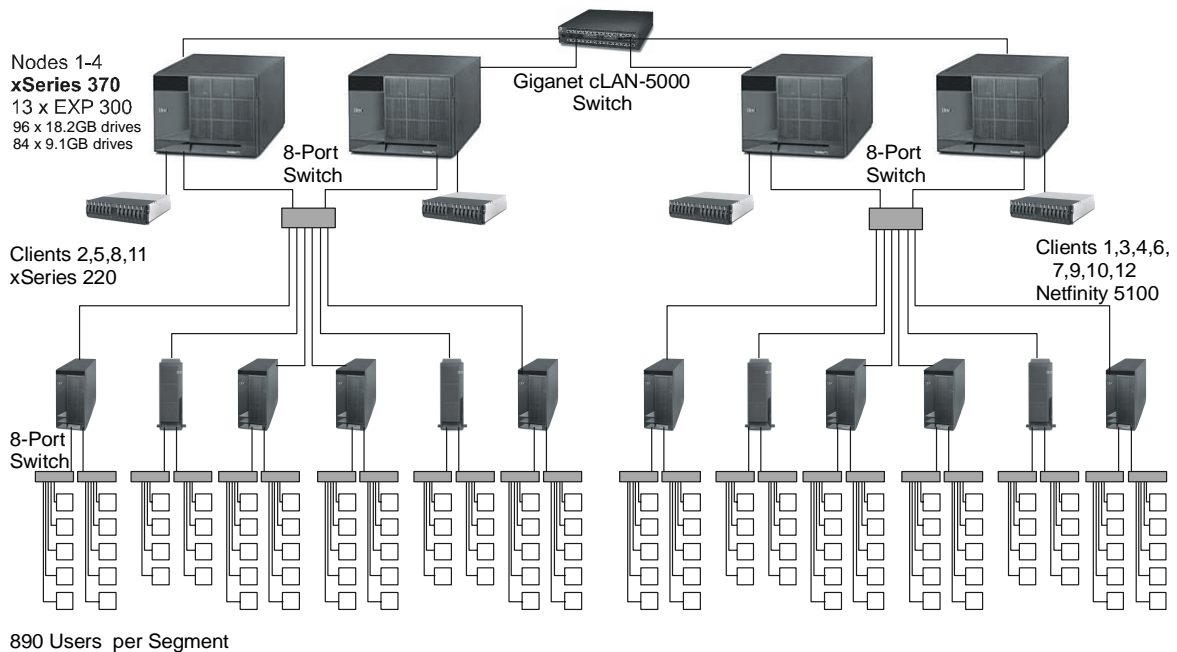
*Diagrams of both measured and priced configurations must be provided, accompanied by a description of the differences.*

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

## Measured Configuration



## Priced Configuration



## Disk Drive Substitution

The priced and measured configurations were identical with one exception. At each server node, the measured configuration used 140 9.1GB drives and 40 18.2GB drives. For the priced configuration, in order to satisfy the 180-day space requirement, we substituted 56 of the 9.1GB drives used in the measured configuration with 56 18.2GB drives. To verify equivalent performance of the drives, a regular TPC-C run was made with DISKPERF on to monitor the disk performance. The PERFMON data showed that the disk latency and the disk queuing depth for the 18.2GB drive is better or equivalent to the 9.1GB drive.

---

## 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. (8.1.2.1)*

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. (8.1.2.2)*

Physical space was allocated to SQL Server 2000 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. (8.1.2.3)*

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 SQL Server 2000 and priced as static space.

The insert and delete functions were verified by the auditor. In addition, the auditor verified that the primary key for each database table could be updated.

### 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. (8.1.2.4)*

The database tables, with one exception, were partitioned across the server nodes using range partitioning. The warehouse ID was used as the partition key. The specifics of the distribution of database partitions across the physical media can be found in Table 4-2. The exception is the ITEM table, which was replicated across all database server nodes.

### Replication

*Replication tables, if used, must be disclosed (see Clause 1.4.6). (8.1.2.5)*

The ITEM table was replicated across all database server nodes. Database triggers were used to ensure that updates to the ITEM table on one node were made to the ITEM table on the other nodes. Their consistency has been verified by the auditor.

### 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). (8.1.2.6)*

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 for each user were captured and verified by the auditor to be unique. In addition, the contents of the database were systematically searched and randomly sampled by the auditor for patterns that would indicate that the random number generator had effected any kind of discernible pattern; none was found.

### Screen Layout

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

All screen layouts followed the TPC Benchmark C Standard Specification exactly.

### 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 done by xSeries 220 and Netfinity 5100 clients. The screen I/O 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 370 server.

### Transaction Profiles

*The percentage of home and remote order-lines in the New-Order transactions must be disclosed. (8.1.3.5)*

*The percentage of New-Order transactions that were rolled back as a result of an unused item number must be disclosed. (8.1.3.6)*

*The number of items per orders entered by New-Order transactions must be disclosed. (8.1.3.7)*

*The percentage of home and remote Payment transactions must be disclosed. (8.1.3.8)*

*The percentage of Payment and Order-Status transactions that used non-primary key (C\_LAST) access to the database must be disclosed. (8.1.3.9)*

*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. (8.1.3.10)*

*The mix (i.e., percentages) of transaction types seen by the SUT must be disclosed. (8.1.3.11)*

**Table 2-1. Transaction Statistics**

<b>New Order</b>	<b>Value (%)</b>
Home warehouse order lines	99.00
Remote warehouse order lines	1.00
Rolled back transactions	1.00
Average number of items per order	10.00
<b>Payment</b>	
Home warehouse payment transactions	85.00
Remote warehouse payment transactions	15.00
<b>Non-Primary Key Access</b>	
Payment transactions using C_LAST	60.01
Order-Status transactions using C_LAST	60.01
<b>Delivery</b>	
Delivery transactions skipped	0
<b>Transaction Mix</b>	
New-Order	44.83
Payment	43.03
Delivery	4.04
Stock Level	4.05
Order Status	4.06

## **Deferred Delivery Mechanism**

*The queuing mechanism used to defer execution of the Delivery transaction must be disclosed. (8.1.3.12)*

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. (8.1.4.1)*

### 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 twice to verify the Atomicity of completed transactions. One time a warehouse ID that was local to the node that the test was being run on was used. The second time a warehouse ID that was remote to the node that the test was being run on was used.

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 twice to verify the Atomicity of the aborted Payment transaction. One time a warehouse ID that was local to the node that the test was being run on was used. The second time a warehouse ID that was remote to the node that the test was being run on was used.

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. The shell scripts were run twice, once on a node that the warehouse being used locally and then again on a node that did not have the warehouse being used locally. 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*
- *Instantaneous interruption (system crash/system hang) in processing that requires system reboot to recover*
- *Failure of all or part of memory (loss of contents)*

Recovery from all three of these failures was demonstrated as described in the following sections.

### **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 150 users submitting transactions over two server nodes.
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 giving SUM1.
2. A test was started under full load with all users submitting transactions.
3. One disk from the log array was removed. Since the disk was RAID-1 mirrored, SQL Server continued to process transactions without interruption.
4. The test continued under full load with all users submitting transactions. A checkpoint was issued, and the system continued to run for another 5 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.

### **Loss of Interconnect**

The following steps were performed:

1. The four Consistency conditions were tested.
2. The current count of the total number of orders, SUM1, was determined by obtaining the sum of the D\_NEXT\_O\_ID column for all rows in the District table.
3. A measurement was started under full load with all users submitting transactions. When all users were submitting transactions, a checkpoint was executed.
4. The checkpoint lasted 9 minutes. During that time the measurement throughput reached a minimum of 90 percent of the reported throughput and maintained that level until the run was paused.
5. After the checkpoint had completed, power was removed from the interconnect switch and Benchcraft was paused and then stopped.
6. After one minute, the power cords were reconnected to the Gigaset switches. There was a burst of I/O on the database drives as the fabric came back up.
7. The database was checkpointed. The database was stopped on all nodes.
8. The four Consistency conditions were again tested successfully.
9. Step 1 was repeated to obtain the current count of the total number of orders, SUM2.
10. The following relationship was verified:  
 $SUM2 \geq (SUM1 + \text{New Order completed} - \text{New Orders rolled back})$

---

## 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. (8.1.5.1)

The database was built with 9,968 warehouses, and the audited run used 9,968 warehouses.

**Table 4-1. Initial Cardinality of Tables**

Table Name	Rows
Warehouse	9,968
District	99,680
Item	100,000
New Order	89,712,000
History	299,040,000
Orders	299,040,000
Customer	299,040,000
Order Line	2,990,389,741
Stock	996,800,000
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. (8.1.5.2)

The following series of table depicts the database configuration of the tested system to meet the 8-hour steady state requirement. The configuration was the same on all 4 nodes. The big\_files file group contains the t\_customer and t\_stock tables. The misc\_file group contains the other TPC-C tables.

**Table 4-2. Distribution of Tables and Logs**

Controller	Disk	Drives	Partition	Size	Use
1	Disk 0 Disk 1	2 - 9.1GB RAID-1 12 - 18.2GB RAID-10	C: NTFS F: RAW	8.47GB 78.17GB 23.53GB	OS DB Log Unallocated
2	Disk 2	28 - 18.2GB RAID-0	1 RAW 2 RAW	23.29GB 11.77GB 447GB	G: H: Free
3	Disk 3	28 - 9.1GB RAID-0	1 RAW 2 RAW	23.29GB 11.77GB 202.23GB	I: J: Free
4	Disk 4	28 - 9.1GB RAID-0	1 RAW 2 RAW S: NTFS	23.29GB 11.77GB 117.19GB 85.03GB	K: L: backup1-1 Free
5	Disk 5	28 - 9.1GB RAID-0	1 RAW 2 RAW T: NTFS	23.29GB 11.77GB 117.19GB 85.03GB	M: N: backup1-2 Free
6	Disk 6	28 - 9.1GB RAID-0	1 RAW 2 RAW U: NTFS	23.29GB 11.77GB 117.19GB 85.03GB	O: P: backup2-1 Free
7	Disk 7	28 - 9.1GB RAID-0	1 RAW 2 RAW V: NTFS	23.29GB 11.77GB 117.19GB 85.03GB	Q: R: backup2-2 Free

The ServeRAID-4H controller offers a standard 32MB battery-backed cache. The write policy for the operating system logical drive and the database log logical drive was set to write-through. All database logical drives used a write-back policy.

### 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/I, 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.*

SQL Server 2000 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 ODBC interface.

## **Partitions/Replications Mapping**

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

The database tables, with one exception, were partitioned across the server nodes using range partitioning. The warehouse ID was used as the partition key. The specifics of the distribution of database partitions across the physical media can be found in Table 4-2. The exception is the ITEM table, which was replicated across all nodes.

## **180-Day Space Requirement**

*Details of the 180-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). (8.1.5.5)*

See Appendix D for details about how the 180-day space requirements were calculated.

---

## Clause 5: Performance Metrics and Response Time Related Items

### Measured tpmC

*Measured tpmC must be reported. (8.1.6.1)*

Measured tpmC: 121,319.23 tpmC

Price per tpmC: \$25.47 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. (8.1.6.2)*

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	90th %
New-Order	0.78	24.82	1.30
Payment	0.77	26.10	1.30
Delivery (interactive)	0.33	2.19	0.83
Stock Level	0.57	6.11	1.10
Order Status	0.51	6.42	1.03
Delivery (deferred)	0.34	6.14	0.46
Menu	0.34	3.88	0.85

### Keying/Think Times

*The minimum, the average, and the maximum keying and think times must be reported for each transaction type. (8.1.6.3)*

Table 5-2 lists the keying/think times for the measured system.

**Table 5-2. Keying/Think Times**

Transaction Type	Minimum	Average	Maximum
New-Order	18.00 / 0.00	18.01 / 12.04	18.07 / 120.51
Payment	3.00 / 0.00	3.01 / 12.05	3.07 / 120.51
Delivery	2.00 / 0.00	2.01 / 5.06	2.08 / 50.51
Stock Level	2.00 / 0.00	2.01 / 5.06	2.05 / 50.50
Order Status	2.00 / 0.00	2.01 / 10.07	2.05 / 100.51

## Response Time Frequency Distribution Curves

Response time frequency distribution curves (see Clause 5.6.1) must be reported for each transaction type. (8.1.6.4)

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

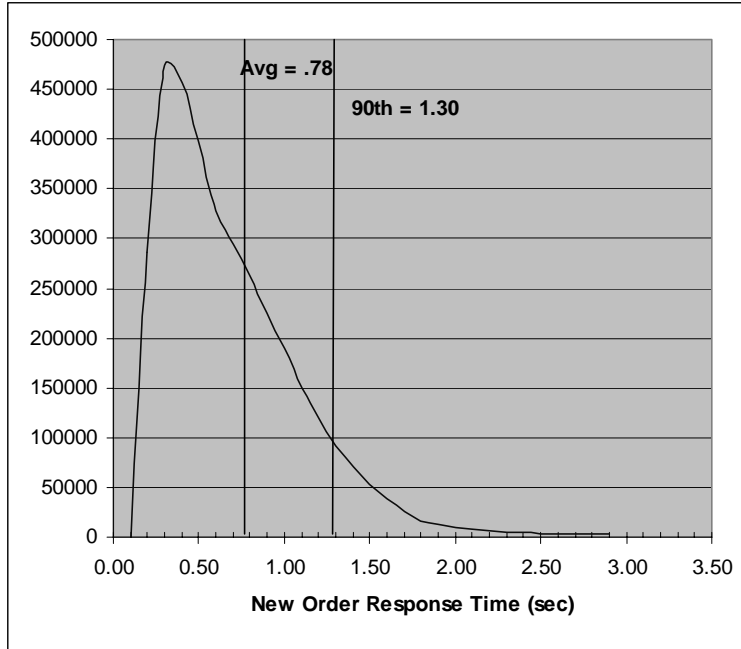
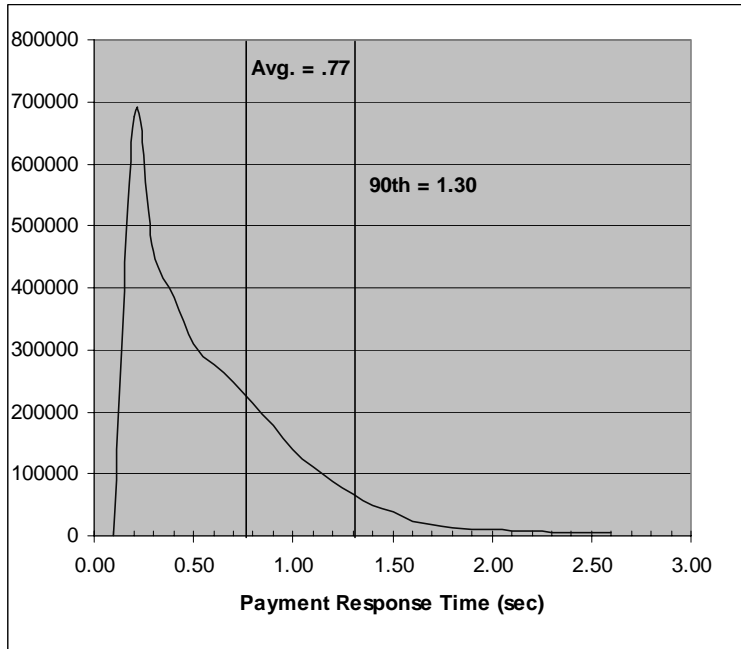
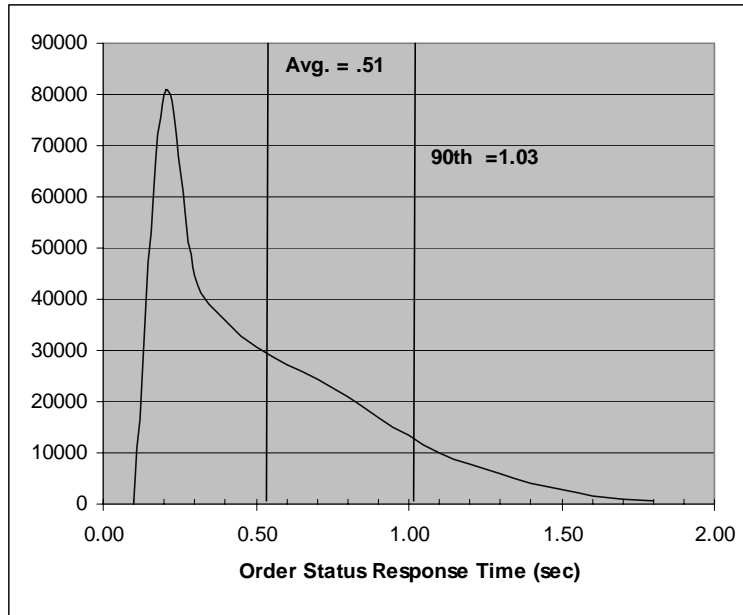


Figure 5-2. Payment Transaction - Response Time Frequency Distribution

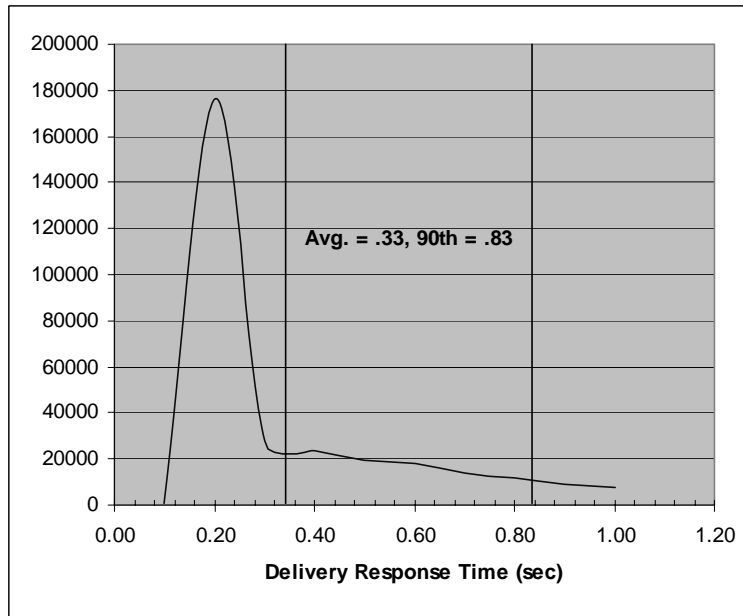




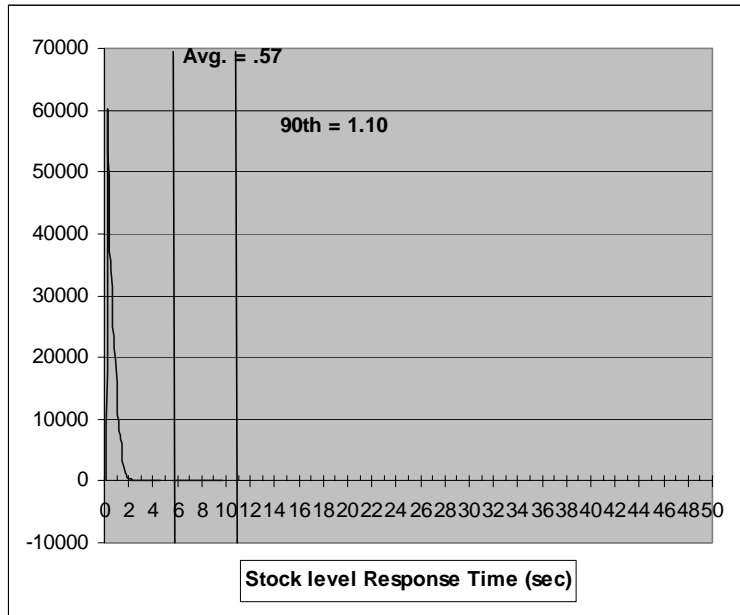
**Figure 5-3. Order-Status Transaction - Response Time Frequency Distribution**



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



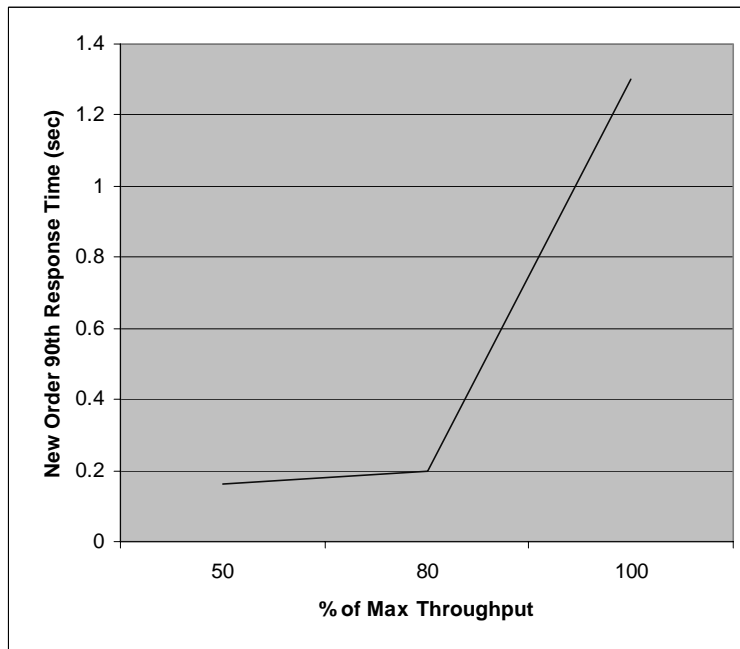
**Figure 5-5. Stock-Level Transaction - Response Time Frequency Distribution**



### Performance Curve for Response Time vs. Throughput

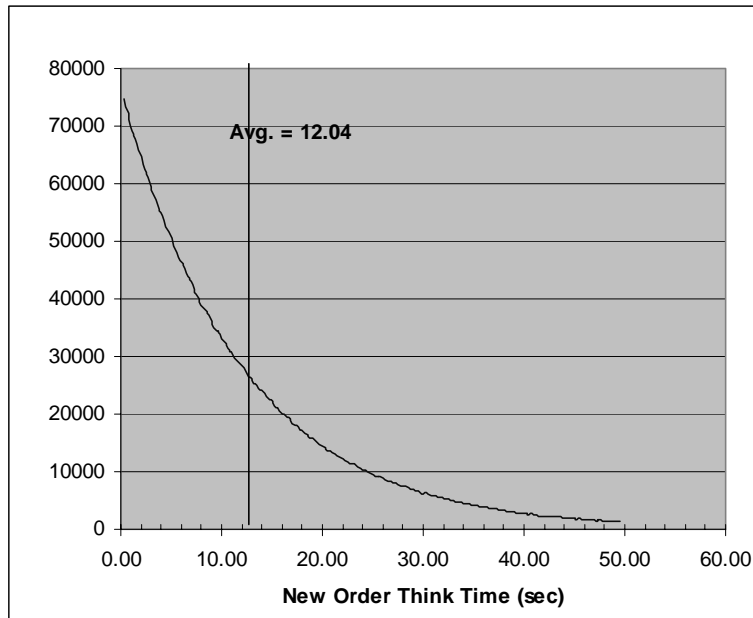
*The performance curve for response time vs. throughput (see Clause 5.6.2) must be reported for the New-Order transaction. (8.1.6.5)*

**Figure 5-6. New-Order Response Time vs. Throughput**

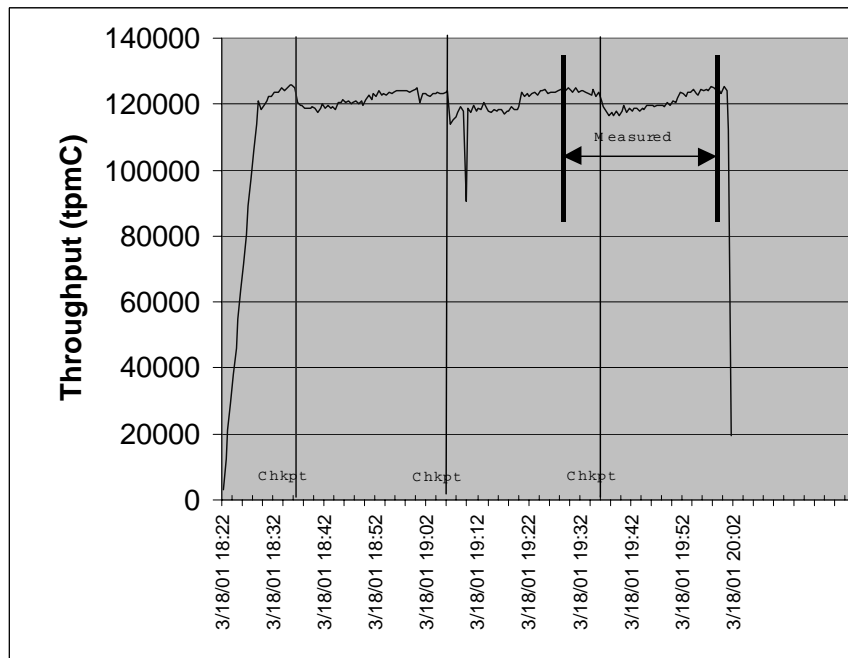


**Figure 5-7. New-Order Think Time Distribution Throughput vs. Elapsed Time**

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. (8.1.6.9)*

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. (8.1.6.10)*

### 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 SQL ODBC library and SQL Server remote stored procedure calls.

## Checkpoints

Checkpoints were executed on the server during the ramp-up phase and at ?-minute intervals. Each measured run contained one checkpoint. SQL Server was started with trace flag 3502, which causes 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.

## Reproducibility Methodology

*A description of the method used to determine the reproducibility of the measurement results must be reported. (8.1.6.11)*

A repeatability measurement was taken on the xSeries 370 server for the same length of time as the measured run. The repeatability measurement was 120,460.87 tpmC.

## Measurement Interval

*A statement of the duration of the measurement interval for the reported Maximum Qualified Throughput (tpmC) must be included. (8.1.6.12)*

The measurement interval was 30 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.

## **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**

<b>New Order</b>	<b>Value (%)</b>
Home warehouse order lines	99.00
Remote warehouse order lines	1.00
Rolled back transactions	1.00
Average number of items per order	10.00
<b>Payment</b>	
Home warehouse payment transactions	85.00
Remote warehouse payment transactions	15.00
<b>Non-Primary Key Access</b>	
Payment transactions using C_LAST	60.01
Order-Status transactions using C_LAST	60.01
<b>Delivery</b>	
Delivery transactions skipped	0
<b>Transaction Mix</b>	
New-Order	44.83
Payment	43.03
Delivery	4.04
Stock Level	4.05
Order Status	4.06

---

## 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. (8.1.7.1)*

The RTE used was Microsoft BenchCraft V2.0 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. (8.1.7.2)*

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). (8.1.7.4)*

The client-to-server LAN connections used 100Mbps switched Ethernet for both the measured and priced configurations. The server-to-server connections used a Gigaset cLAN-5000 Switch for both the measured and priced configurations. The server-to-server protocol was VIA.

In the measured configuration, the 24 LAN segment connections between clients and RTEs used the 8-port 10/100Mbps switch as shown in the measured configuration diagram. The priced configuration included 10Mbps hubs as shown in the priced configuration diagram.

## **Network Bandwidth**

*The bandwidth of the network(s) used in the tested/priced configuration must be disclosed. (8.1.7.5)*

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 clients 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. (8.1.7.6)*

The configuration did not require any operator intervention to sustain the reported throughput during the eight-hour period.



---

## 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. (8.1.8.1)*

*The total 5-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. (8.1.8.2)*

A detailed list of all hardware and software, including the 5-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. (8.1.8.3)*

The 900MHz model of the xSeries 370 server, along with the 900MHz processor option, will be generally available April 13, 2001. All other hardware is generally available now. Datacenter Server will be generally available May 31, 2001. All other software used in this benchmark is generally available now.

### Measured tpmC

*A statement of the measured tpmC, as well as the respective calculations for the 5-year pricing, price/performance (price/tpmC) and the availability date must be included. (8.1.8.4)*

- Maximum Qualified Throughput: 121,319.23 tpmC
- Price per tpmC: \$25.47 per tpmC
- Five-year cost of ownership: \$3,090,250

### 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. (8.1.8.6)

The component pricing based on usage is shown below:

- 4 copies of Microsoft Windows 2000 Datacenter Server
- 12 copies of Microsoft Windows 2000 Server

- 5-year support for all hardware components (spares were priced for hardware components for which 4-hour response time is not offered)

## **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). (8.1.8.7)*  
*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. (8.1.8.8)*

A detailed list of all hardware and software, including the 5-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. (8.1.9.1)*

This implementation of the TPC-C benchmark was audited by Brad 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<sup>TM</sup>C," the Full Disclosure Report must have been submitted to the TPC Administrator as well as written permission obtained to distribute same. (8.2)*

This implementation of the TPC Benchmark C was audited by Brad Askins of InfoSizing, Inc. Further information may be obtained from:

InfoSizing, Inc.  
1373 North Franklin Street  
Colorado Springs, CO 80903  
Phone: 719-473-7555  
Fax: 719-473-7554

Benchmark Sponsor: William D. Hall  
 Mgr., Server Systems Performance  
 IBM Server Group  
 3039 Cornwallis Road  
 Research Triangle Park, NC27709

March 20, 2001

I verified the TPC Benchmark™ C performance of the following Client/Server configuration:

Platform: **IBM @server xSeries 370**  
 Operating system: **Microsoft Windows 2000 Datacenter Server**  
 Database Manager: **Microsoft SQL Server 2000 Enterprise Edition**  
 Transaction Manager: **Microsoft COM+**

The results were:

CPU's Speed	Memory	Disks(total)	NewOrder 90% Response Time	tpmC
<b>4 Servers: IBM @server xSeries 370 each with:</b>				
8 x Pentium III Xeon (900 MHz)	32 GB Main (2MB L2 Cache per processor)	96 x 18.2 GB 86 x 9.1 GB	1.30 Seconds	<b>121319.23</b>
<b>8 Clients: IBM Netfinity 5100 each with:</b>				
2 x Pentium III (733 MHz)	512 MB Main (256KB L2 Cache per processor)	1 x 9.1 GB	n/a	
<b>4 Clients: IBM @server xSeries 220 each with:</b>				
2 x Pentium III (933 MHz)	640 MB Main (256KB L2 Cache per processor)	1 x 9.1 GB	n/a	

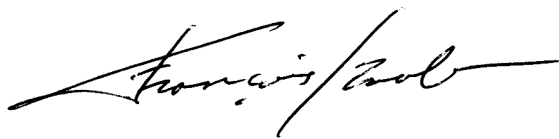
In my opinion, these performance results were produced in compliance with the TPC's requirements for the benchmark. The following verification items were given special attention:

- 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 30 minutes (1800 seconds)
- The reported measurement interval was representative of steady state conditions
- One checkpoint was taken during the reported measurement interval
- The repeatability of the measured performance was verified
- The 180 day storage requirement was correctly computed
- The ability to update primary keys with the same syntax as other columns was verified.
- The system pricing was verified for major components and maintenance

Additional Audit Notes:

The measured system included 224 9.1 GB 15K IBM model 19k0655 disks that were substituted by 224 18.2 GB IBM 19k0656 in the priced configuration. Based on the specifications of these disks and on additional performance data collected on these disks, it is my opinion that this substitution does not have a material effect on the reported performance.

Respectfully Yours,



François Raab, President



Bradley J. Askins, Auditor

# Appendix A: Source Code

## Web Client Source Code

### *dllldata.c*

```
*****
DllData file -- generated by MIDL compiler

DO NOT ALTER THIS FILE

This file is regenerated by MIDL on every IDL file compile.

To completely reconstruct this file, delete it and rerun MIDL
on all the IDL files in this DLL, specifying this file for the
/dllldata command line option

*****

#include <rpcproxy.h>

#ifdef __cplusplus
extern "C" {
#endif

EXTERN_PROXY_FILE(tpcc_com_remote_ps )

PROXYFILE_LIST_START
/* Start of list */
REFERENCE_PROXY_FILE(tpcc_com_remote_ps ),
/* End of list */
PROXYFILE_LIST_END

DLLDATA_ROUTINES(aProxyFileList, GET_DLL_CLSID)

#ifdef __cplusplus
} /*extern "C" */
#endif

/* end of generated dllldata file */
```

### *error.h*

```
/*
FILE:          ERROR.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 error exception
classes.
*
* Change history:
*              4.20.000 - updated rev number to match kit
```

```
*              4.21.000 - fixed bug: ~CBaseErr needed to be
declared virtual
*/

#pragma once

#ifndef _INC_STRING
#include <string.h>
#endif

const int m_szMsg_size = 512;
const int m_szApp_size = 64;
const int m_szLoc_size = 64;

//error message structure used in ErrorText routines
typedef struct _SERRORMSG
{
    int                iError;
//error id of message
    char               szMsg[256];
//message to sent to browser
} SERRORMSG;

typedef enum _ErrorLevel
{
    ERR_FATAL_LEVEL           = 1,
    ERR_WARNING_LEVEL        = 2,
    ERR_INFORMATION_LEVEL    = 3
} ErrorLevel;

#define ERR_TYPE_LOGIC
-1 //logic error in program;
internal error
#define ERR_SUCCESS
0 //success (a
non-error error)
#define ERR_BAD_ITEM_ID
1 //expected abort record in
txnRecord
#define ERR_TYPE_DELIVERY_POST
2 //expected delivery post failed
#define ERR_TYPE_WEBDLL
3 //tpcc web
generated error
#define ERR_TYPE_SQL
4 //sql server generated error
#define ERR_TYPE_DBLIB
5 //dblib generated error
#define ERR_TYPE_ODBC
6 //odbc generated error
#define ERR_TYPE_SOCKET
7 //error on communication
socket client rte only
#define ERR_TYPE_DEADLOCK
8 //dblib and odbc only
deadlock condition
#define ERR_TYPE_COM
9 //error from COM call
#define ERR_TYPE_TUXEDO
10 //tuxedo error
#define ERR_TYPE_OS
11 //operating
system error
#define ERR_TYPE_MEMORY
12 //memory allocation error
```

```

#define ERR_TYPE_TPCC_ODBC
    13 //error from tpcc odbc txn
module
#define ERR_TYPE_TPCC_DBLIB
    14 //error from tpcc dblib txn
module
#define ERR_TYPE_DELISRV
    15 //delivery server error
#define ERR_TYPE_TXNLOG
    16 //txn log error
#define ERR_TYPE_BCCONN
    17 //Benchcraft connection
class
#define ERR_TYPE_TPCC_CONN
    18 //Benchcraft connection
class
#define ERR_TYPE_ENCINA
    19 //Encina error
#define ERR_TYPE_COMPONENT
    20 //error from COM
component
#define ERR_TYPE_RTE
    21 //Benchcraft rte
#define ERR_TYPE_AUTOMATION
    22 //Benchcraft automation
errors
#define ERR_TYPE_DRIVER
    23 //Driver engine errors
#define ERR_TYPE_RTE_BASE
    24 //Framework errors

// TPC-W error types
#define ERR_TYPE_TPCW_CONN
    50 //Benchcraft connection
class
#define ERR_TYPE_TPCW_HTML
    51 //error from TpcwHtml dll
#define ERR_TYPE_TPCW_USER
    52 //error from TPC-W user
class
#define ERR_TYPE_TPCW_ENG_BASE
    53
#define ERR_TYPE_TPCW_ENG_OS
    54
#define ERR_TYPE_HTML_RESP
    55
#define ERR_TYPE_TPCW_ODBC
    56

#define ERR_INS_MEMORY "Insufficient
Memory to continue."
#define ERR_UNKNOWN "Unknown error."
#define ERR_MSG_BUF_SIZE 512
#define INV_ERROR_CODE -1

class CBaseErr
{
public:
    CBaseErr(LPCTSTR szLoc = NULL)
    {
        m_idMsg = INV_ERROR_CODE;

        if (szLoc)
        {
            m_szLoc = new char[m_szLoc_size];
            strcpy(m_szLoc, szLoc);
        }
    }

```

```

        else
            m_szLoc = NULL;

        m_szApp = new
char[m_szApp_size];
        GetModuleFileName(GetModuleHandle(NULL),
m_szApp, m_szApp_size);
    }

    CBaseErr(int idMsg, LPCTSTR szLoc = NULL)
    {
        m_idMsg = idMsg;

        if (szLoc)
        {
            m_szLoc = new char[m_szLoc_size];
            strcpy(m_szLoc, szLoc);
        }
        else
            m_szLoc = NULL;

        m_szApp = new
char[m_szApp_size];
        GetModuleFileName(GetModuleHandle(NULL),
m_szApp, m_szApp_size);
    }

    virtual ~CBaseErr(void)
    {
        if (m_szApp)
            delete [] m_szApp;
        if (m_szLoc)
            delete [] m_szLoc;
    };

    virtual void Draw(HWND hwnd, LPCTSTR szStr = NULL)
    {
        int j = 0;
        char szTmp[512];

        if (szStr)
            j = sprintf(szTmp, "%s\n", szStr);
        if (ErrorNum() != INV_ERROR_CODE)
            j += sprintf(szTmp+j, "Error =
%d\n", ErrorNum());
        if (m_szLoc)
            j += sprintf(szTmp+j, "Location =
%s\n", GetLocation());

        j += sprintf(szTmp+j, "%s\n", ErrorText());

        ::MessageBox(hwnd, szTmp, m_szApp,
MB_OK);
    }

    char *GetApp(void) { return m_szApp; }
    char *GetLocation(void) { return m_szLoc; }
    virtual int ErrorNum() { return m_idMsg; }
    virtual int ErrorType() = 0; // a value which
distinguishes the kind of error that occurred
    virtual char *ErrorText() = 0; // a string (i.e., human
readable) representation of the error

protected:
    char *m_szApp;
    char *m_szLoc; // code location where the error
occurred

    int m_idMsg;

```

```

};

class CSocketErr : public CBaseErr
{
public:
    enum Action
    {
        eNone = 0,
        eSend,
        eSocket,
        eBind,
        eConnect,
        eListen,
        eHost,
        eRecv,
        eGetHostByName,
        eWSACreateEvent,
        eWSASend,
        eWSAGetOverlappedResult,
        eWSARecv,
        eWSAWaitForMultipleEvents,
        eWSAStartup,
        eWSAResetEvent,
    };

    CSocketErr(Action eAction, LPCTSTR szLocation =
NULL);

    ~CSocketErr()
    {
        if (m_szErrorText != NULL)
            delete [] m_szErrorText;
    };

    Action    m_eAction;
    char     *m_szErrorText;

    int ErrorType() { return ERR_TYPE_SOCKET;};
    char *ErrorText(void);

};

class CSystemErr : public CBaseErr
{
public:
    enum Action
    {
        eNone = 0,
        eTransactNamedPipe,
        eWaitNamedPipe,
        eSetNamedPipeHandleState,
        eCreateFile,
        eCreateProcess,
        eCallNamedPipe,
        eCreateEvent,
        eCreateThread,
        eVirtualAlloc,
        eReadFile = 10,
        eWriteFile,
        eMapViewOfFile,
        eCreateFileMapping,
        eInitializeSecurityDescriptor,
        eSetSecurityDescriptorDacl,
        eCreateNamedPipe,
        eConnectNamedPipe,
        eWaitForSingleObject,
        eRegOpenKeyEx,
        eRegQueryValueEx = 20,
    };

```

```

        eBeginThread,
        eRegEnumValue,
        eRegSetValueEx,
        eRegCreateKeyEx,
        eWaitForMultipleObjects,
        eRegisterClassEx,
        eCreateWindow,
        eCreateSemaphore,
    };

    CSystemErr(Action eAction,
LPCTSTR szLocation);
    CSystemErr(int iError, Action
eAction, LPCTSTR szLocation);
    int ErrorType() { return
ERR_TYPE_OS;};
    char *ErrorText(void);
    void Draw(HWND hwnd, LPCTSTR szStr = NULL);

    Action    m_eAction;

private:
    char m_szMsg[ERR_MSG_BUF_SIZE];
};

class CMemoryErr : public CBaseErr
{
public:
    CMemoryErr();

    int ErrorType() {return ERR_TYPE_MEMORY;};
    char *ErrorText() {return ERR_INS_MEMORY;};
};

```

### **install.c**

```

/* FILE: INSTALL.C
 * Microsoft TPC-C Kit Ver.
4.20.000
 * Copyright Microsoft,
1999
 * All Rights Reserved
 *
 * not audited
 *
 * PURPOSE: Automated installation application
for TPC-C Web Kit
 * Contact: Charles Levine (clevine@microsoft.com)
 *
 * Change history:
 * 4.20.000 - added COM installation steps
 */

#include <windows.h>
#include <direct.h>
#include <io.h>
#include <stdlib.h>
#include <stdio.h>
#include <commctrl.h>
#include "..\..\common\src\ReadRegistry.h"

#include "resource.h"

#define WM_INITTEXT WM_USER+100

HICON hIcon;

```



```

HINSTANCE      hInst;

DWORD          versionExeMS;
DWORD          versionExeLS;
DWORD          versionExeMM;
DWORD          versionDIIMS;
DWORD          versionDIILS;

// TPC-C registry settings
TPCCREGISTRYDATA  Reg;

static int     iPoolThreadLimit;
static int     iThreadTimeout;
static int     iListenBackLog;
static int     iAcceptExOutstanding;

static int     iMaxPhysicalMemory;
//max physical memory in MB
static char    szLastFileName[64];          // last file we
worked on (for error reporting)

BOOL CALLBACK  LicenseDlgProc(HWND hwnd, UINT
uMsg, WPARAM wParam, LPARAM lParam);
BOOL CALLBACK  UpdatedDlgProc(HWND hwnd,
UINT uMsg, WPARAM wParam, LPARAM lParam);
BOOL CALLBACK  MainDlgProc(HWND hwnd, UINT
uMsg, WPARAM wParam, LPARAM lParam);
BOOL CALLBACK  CopyDlgProc(HWND hwnd, UINT
uMsg, WPARAM wParam, LPARAM lParam);
static void    ProcessOK(HWND hwnd, char
*szDllPath);
static void    ReadRegistrySettings(void);
static void    WriteRegistrySettings(char
*szDllPath);
static BOOL    RegisterDLL(char *szFileName);
static int     CopyFiles(HWND hDlg,
char *szDllPath);
static BOOL    GetInstallPath(char *szDllPath);
static void    GetVersionInfo(char *szDLLPath,
char *szExePath);
static BOOL    CheckWWWebService(void);
static BOOL    StartWWWebService(void);
static BOOL    StopWWWebService(void);
static void    UpdateDialog(HWND hDlg);

BOOL install_com(char *szDllPath);

#include "..\..\common\src\ReadRegistry.cpp"

int WINAPI WinMain( HINSTANCE hInstance, HINSTANCE
hPrevInstance, LPSTR lpCmdLine, int nCmdShow )
{
    int iRc;

    hInst = hInstance;

    InitCommonControls();

    hIcon = LoadIcon(hInstance,
MAKEINTRESOURCE(IDI_ICON1));

    iRc = DialogBox(hInstance,
MAKEINTRESOURCE(IDD_DIALOG4),GetDesktopWindow(),
LicenseDlgProc);
    if ( iRc )
    {

```

```

        iRc = DialogBox(hInstance,
MAKEINTRESOURCE(IDD_DIALOG1),GetDesktopWindow(),
MainDlgProc);
        if ( iRc )
        {
            DialogBoxParam(hInstance,
MAKEINTRESOURCE(IDD_DIALOG2),GetDesktopWindow(),
UpdatedDlgProc, (LPARAM)iRc);
        }

        DestroyIcon(hIcon);
        return 0;
    }

    BOOL CALLBACK LicenseDlgProc(HWND hwnd, UINT uMsg,
WPARAM wParam, LPARAM lParam)
    {
        HGLOBAL          hRes;
        HRSRC            hResInfo;
        BYTE             *pSrc, *pDst;
        DWORD            dwSize;
        static HFONT     hFont;

        switch(uMsg)
        {
            case WM_INITDIALOG:
                hFont = CreateFont(-12, 0, 0, 0, 400,
0, 0, 0, 0, 0, 0, 0, "Arial");
                SendMessage( GetDlgItem(hwnd,
IDR_LICENSE1), WM_SETFONT, (WPARAM)hFont,
MAKELPARAM(0, 0) );
                PostMessage(hwnd,
WM_INITTEXT, (WPARAM)0, (LPARAM)0);
                return TRUE;

            case WM_INITTEXT:
                hResInfo = FindResource(hInst,
MAKEINTRESOURCE(IDR_LICENSE1),"LICENSE");
                dwSize = SizeofResource(hInst,
hResInfo);
                hRes = LoadResource(hInst,
hResInfo );
                pSrc = (BYTE
*)LockResource(hRes);
                pDst = (unsigned char
*)malloc(dwSize+1);
                if ( pDst )
                {
                    memcpy(pDst, pSrc,
dwSize);
                    pDst[dwSize] = 0;
                    SetDlgItemText(hwnd,
IDC_LICENSE, (const char *)pDst);
                    free(pDst);
                }
                else
                    SetDlgItemText(hwnd,
IDC_LICENSE, (const char *)pSrc);
                return TRUE;

            case WM_DESTROY:
                DeleteObject(hFont);
                return TRUE;

            case WM_COMMAND:
                if ( wParam == IDOK )
                    EndDialog(hwnd, TRUE);
                if ( wParam == IDCANCEL )
                    EndDialog(hwnd,
FALSE);

```

<pre>                 default:                     break;             }             return FALSE;         }     }      BOOL CALLBACK UpdatedDlgProc(HWND hwnd, UINT uMsg,     WPARAM wParam, LPARAM lParam)     {         switch(uMsg)         {             case WM_INITDIALOG:                 switch(lParam)                 {                     case 1:                     case 2:                 }                 SetDlgItemText(hwnd, IDC_RESULTS, "TPC-C Web Client                 Installed");                 break;             }             return TRUE;         case WM_COMMAND:             if ( wParam == IDOK )                 EndDialog(hwnd, TRUE);             break;         default:             break;         }         return FALSE;     }      BOOL CALLBACK MainDlgProc(HWND hwnd, UINT uMsg,     WPARAM wParam, LPARAM lParam)     {         PAINTSTRUCT ps;         MEMORYSTATUS memoryStatus;         OSVERSIONINFO VI;         char szTmp[256];         static char szDllPath[256];         static char szExePath[256];          switch(uMsg)         {             case WM_INITDIALOG:                 GlobalMemoryStatus(&amp;memoryStatus);                 iMaxPhysicalMemory=                 (memoryStatus.dwTotalPhys/ 1048576);                  if ( GetInstallPath(szDllPath) )                 {                     MessageBox(hwnd, "Error                     internet service inetsrv is not installed.", NULL, MB_ICONSTOP                       MB_OK);                 }                 EndDialog(hwnd,                 FALSE);                 return TRUE;             }              // set default values             ZeroMemory( &amp;Reg, sizeof(Reg) );             ReadTPCCRegistrySettings( &amp;Reg );             ReadRegistrySettings();              Reg.dwNumberOfDeliveryThreads=             7;              Reg.dwMaxConnections = 10000; </pre>	<pre> 1000;  "tpcc"); "sa"); strcpy(Reg.szDbUser, strcpy(Reg.szDbPassword, ""); // iMaxPhysicalMemory * 2; iPoolThreadLimit = iPoolThreadLimit = 190; iThreadTimeout = 86400; iListenBackLog = 150; iAcceptExOutstanding = 40; GetModuleFileName(hInst, szExePath, sizeof(szExePath)); GetVersionInfo(szDllPath, szExePath); wsprintf(szTmp, "Version %d.%2.2d.%3.3d", versionExeMS, versionExeMM, versionExeLS); SetDlgItemText(hwnd, IDC_VERSION, szTmp); SetDlgItemText(hwnd, IDC_PATH, szDllPath); SetDlgItemText(hwnd, ED_DB_SERVER, Reg.szDbServer); SetDlgItemText(hwnd, ED_DB_USER_ID, Reg.szDbUser); SetDlgItemText(hwnd, ED_DB_PASSWORD, Reg.szDbPassword); SetDlgItemText(hwnd, ED_DB_NAME, Reg.szDbName); SetDlgItemInt(hwnd, ED_THREADS, Reg.dwNumberOfDeliveryThreads, FALSE); SetDlgItemInt(hwnd, ED_MAXCONNECTION, Reg.dwMaxConnections, FALSE); SetDlgItemInt(hwnd, ED_MAXDELIVERIES, Reg.dwMaxPendingDeliveries, FALSE); SetDlgItemInt(hwnd, ED_IIS_MAX_THREAD_POOL_LIMIT, iPoolThreadLimit, FALSE); SetDlgItemInt(hwnd, ED_IIS_THREAD_TIMEOUT, iThreadTimeout, FALSE); SetDlgItemInt(hwnd, ED_IIS_LISTEN_BACKLOG, iListenBackLog, FALSE); SetDlgItemInt(hwnd, ED_WEB_SERVICE_BACKLOG_QUEUE_SIZE, iAcceptExOutstanding, FALSE); CheckDlgButton(hwnd, IDC_DBLIB, 0); CheckDlgButton(hwnd, IDC_ODBC, 0); if ( Reg.eDB_Protocol == DBLIB )     CheckDlgButton(hwnd, IDC_DBLIB, 1); else if (Reg.eDB_Protocol == ODBC) </pre>
---	---

<pre> IDC_ODBC, 1);         CheckDlgButton(hwnd, // check OS version level for COM. Must be at least Windows 2000         VI.dwOSVersionInfoSize = sizeof(VI);         GetVersionEx( &amp;VI);         if (VI.dwMajorVersion &lt; 5)         {             HWND hDlg = GetDlgItem(hwnd, IDC_TM_MTS );             EnableWindow( hDlg, 0 ); // disable COM option             if (Reg.eTxnMon == COM)                 Reg.eTxnMon = None;         } IDC_TM_NONE, 0); IDC_TM_TUXEDO, 0); IDC_TM_MTS, 0); IDC_TM_ENCINA, 0);         switch (Reg.eTxnMon)         {         case None:             CheckDlgButton(hwnd, IDC_TM_NONE, 1);             break;         case TUXEDO:             CheckDlgButton(hwnd, IDC_TM_TUXEDO, 1);             break;         case ENCINA:             CheckDlgButton(hwnd, IDC_TM_ENCINA, 1);             break;         case COM:             CheckDlgButton(hwnd, IDC_TM_MTS, 1);             break;         }         return TRUE;         case WM_PAINT:             if ( IsIconic(hwnd) )             {                 BeginPaint(hwnd, &amp;ps);                 DrawIcon(ps.hdc, 0, 0, hIcon);                 EndPaint(hwnd, &amp;ps);                 return TRUE;             }             break;         case WM_COMMAND:             if ( HIWORD(wParam) == BN_CLICKED )             {                 switch( LOWORD(wParam))                 {                     case IDC_DBLIB: </pre>	<pre> return TRUE;         case IDC_ODBC:             return TRUE;         case IDOK:             ProcessOK(hwnd, szDllPath);             return TRUE;         case IDC_CANCEL:             EndDialog(hwnd, FALSE);             return TRUE;         default:             return FALSE;         }         }         default:             break;         }         }         return FALSE;     }     static void ProcessOK(HWND hwnd, char *szDllPath)     {         int          d;         HWND        hDlg;         int          rc;         char         szFullName[256];         char         szErrTxt[128];         // read settings from dialog         Reg.dwNumberOfDeliveryThreads = GetDlgItemInt(hwnd, ED_THREADS, &amp;d, FALSE);         Reg.dwMaxConnections = GetDlgItemInt(hwnd, ED_MAXCONNECTION, &amp;d, FALSE);         Reg.dwMaxPendingDeliveries = GetDlgItemInt(hwnd, ED_MAXDELIVERIES, &amp;d, FALSE);         GetDlgItemText(hwnd, ED_DB_SERVER, Reg.szDbServer, sizeof(Reg.szDbServer));         GetDlgItemText(hwnd, ED_DB_USER_ID, Reg.szDbUser, sizeof(Reg.szDbUser));         GetDlgItemText(hwnd, ED_DB_PASSWORD, Reg.szDbPassword, sizeof(Reg.szDbPassword));         GetDlgItemText(hwnd, ED_DB_NAME, Reg.szDbName, sizeof(Reg.szDbName));         if ( IsDlgButtonChecked(hwnd, IDC_DBLIB) )         {             Reg.eDB_Protocol = DBLIB;             rc = 1;         }         else if ( IsDlgButtonChecked(hwnd, IDC_ODBC) )         {             Reg.eDB_Protocol = ODBC;             rc = 2;         }         if ( IsDlgButtonChecked(hwnd, IDC_TM_NONE) ) </pre>
---	---

```

        Reg.eTxnMon = None;
    else if ( IsDlgButtonChecked(hwnd, IDC_TM_TUXEDO) )
        Reg.eTxnMon = TUXEDO;
    else if ( IsDlgButtonChecked(hwnd, IDC_TM_MTS) )
        Reg.eTxnMon = COM;
    else if ( IsDlgButtonChecked(hwnd, IDC_TM_ENCINA) )
        Reg.eTxnMon = ENCINA;

    iPoolThreadLimit = GetDlgItemInt(hwnd,
ED_IIS_MAX_THREAD_POOL_LIMIT,&d, FALSE);
    iThreadTimeout = GetDlgItemInt(hwnd,
ED_IIS_THREAD_TIMEOUT, &d, FALSE);
    iListenBackLog = GetDlgItemInt(hwnd,
ED_IIS_LISTEN_BACKLOG,&d, FALSE);
    iAcceptExOutstanding = GetDlgItemInt(hwnd,
ED_WEB_SERVICE_BACKLOG_QUEUE_SIZE,&d, FALSE);

    ShowWindow(hwnd, SW_HIDE);
    hDlg = CreateDialog(hInst,
MAKEINTRESOURCE(IDD_DIALOG3),hwnd, CopyDlgProc);
    ShowWindow(hDlg, SW_SHOWNA);
    UpdateDialog(hDlg);

    // write binaries to inetpub\wwwroot
    rc = CopyFiles(hDlg, szDllPath);
    if ( !rc )
    {
        ShowWindow(hwnd, SW_SHOWNA);
        DestroyWindow(hDlg);
        strcpy( szErrTxt, "Error(s) occurred when
creating " );
        strcat( szErrTxt, szLastFileName );
        MessageBox(hwnd, szErrTxt, NULL,
MB_ICONSTOP | MB_OK);
        EndDialog(hwnd, 0);
        return;
    }

    // update registry
    SetDlgItemText(hDlg, IDC_STATUS, "Updating
Registry.");
    SendDlgItemMessage(hDlg, IDC_PROGRESS1,
PBM_STEPIT, 0, 0);
    UpdateDialog(hDlg);
    WriteRegistrySettings(szDllPath);

    // register com proxy stub
    strcpy(szFullName, szDllPath);
    strcat(szFullName, "tpcc_com_ps.dll");
    if (!RegisterDLL(szFullName))
    {
        ShowWindow(hwnd, SW_SHOWNA);
        DestroyWindow(hDlg);
        strcpy( szErrTxt, "Error occurred when
registering " );
        strcat( szErrTxt, szFullName );
        MessageBox(hwnd, szErrTxt, NULL,
MB_ICONSTOP | MB_OK);
        EndDialog(hwnd, 0);
        return;
    }
    // register remote com proxy stub
    strcpy(szFullName, szDllPath);
    strcat(szFullName, "tpcc_com_remote_ps.dll");
    if (!RegisterDLL(szFullName))
    {

```

```

        ShowWindow(hwnd, SW_SHOWNA);
        DestroyWindow(hDlg);
        strcpy( szErrTxt, "Error occurred when
registering " );
        strcat( szErrTxt, szFullName );
        MessageBox(hwnd, szErrTxt, NULL,
MB_ICONSTOP | MB_OK);
        EndDialog(hwnd, 0);
        return;
    }
    // if using COM
    if (Reg.eTxnMon == COM)
    {
        SetDlgItemText(hDlg, IDC_STATUS,
"Configuring COM.");
        SendDlgItemMessage(hDlg, IDC_PROGRESS1,
PBM_STEPIT, 0, 0);
        UpdateDialog(hDlg);

        if (install_com(szDllPath))
        {
            ShowWindow(hwnd,
SW_SHOWNA);
            DestroyWindow(hDlg);
            strcpy( szErrTxt, "Error occurred
when configuring COM settings." );
            MessageBox(hwnd, szErrTxt, NULL,
MB_ICONSTOP | MB_OK);
            EndDialog(hwnd, 0);
            return;
        }
        Sleep(100);

        ShowWindow(hwnd, SW_SHOWNA);
        DestroyWindow(hDlg);

        EndDialog(hwnd, rc);
        return;
    }
static void ReadRegistrySettings(void)
{
    HKEY    hKey;
    DWORD   size;
    DWORD   type;

    if ( RegOpenKeyEx(HKEY_LOCAL_MACHINE,
"SYSTEM\\CurrentControlSet\\Services\\Inetinfo\\Parameters", 0,
KEY_READ, &hKey) == ERROR_SUCCESS )
    {
        size = sizeof(iPoolThreadLimit);
        if ( RegQueryValueEx(hKey,
"PoolThreadLimit", 0, &type, (char *)&iPoolThreadLimit, &size) ==
ERROR_SUCCESS )
            if ( !iPoolThreadLimit )
                iPoolThreadLimit =
iMaxPhysicalMemory * 2;

        size = sizeof(iThreadTimeout);
        if ( RegQueryValueEx(hKey, "ThreadTimeout",
0, &type, (char *)&iThreadTimeout, &size) == ERROR_SUCCESS )
            if ( !iThreadTimeout )
                iThreadTimeout = 86400;

```

```

        size = sizeof(iListenBackLog);
        if ( RegQueryValueEx(hKey, "ListenBackLog",
0, &type, (char *)&iListenBackLog, &size) == ERROR_SUCCESS )
            if ( !iListenBackLog )
                iListenBackLog = 150;

        RegCloseKey(hKey);
    }

    if ( RegOpenKeyEx(HKEY_LOCAL_MACHINE,
"SYSTEM\\CurrentControlSet\\Services\\W3SVC\\Parameters", 0,
KEY_READ, &hKey) == ERROR_SUCCESS )
    {
        size = sizeof(iAcceptExOutstanding);
        if ( RegQueryValueEx(hKey,
"AcceptExOutstanding", 0, &type, (char *)&iAcceptExOutstanding,
&size) == ERROR_SUCCESS )
            if ( !iAcceptExOutstanding )
                iAcceptExOutstanding =
40;

        RegCloseKey(hKey);
    }
}

static void WriteRegistrySettings(char *szDllPath)
{
    HKEY    hKey;
    DWORD   dwDisposition;
    char    szTmp[256];
    char    *ptr;
    int     iRc;

    if ( RegCreateKeyEx(HKEY_LOCAL_MACHINE,
"SOFTWARE\\Microsoft\\TPCC", 0, NULL,
REG_OPTION_NON_VOLATILE, KEY_ALL_ACCESS, NULL,
&hKey, &dwDisposition) == ERROR_SUCCESS )
    {
        strcpy(szTmp, szDllPath);
        ptr = strstr(szTmp, "tpcc");
        if ( ptr )
            *ptr = 0;

        RegSetValueEx(hKey, "Path", 0, REG_SZ,
szTmp, strlen(szTmp)+1);

        RegSetValueEx(hKey,
"NumberOfDeliveryThreads", 0, REG_DWORD, (char
*)&Reg.dwNumberOfDeliveryThreads,
sizeof(Reg.dwNumberOfDeliveryThreads));
        RegSetValueEx(hKey, "MaxConnections", 0,
REG_DWORD, (char *)&Reg.dwMaxConnections,
sizeof(Reg.dwMaxConnections));
        RegSetValueEx(hKey, "MaxPendingDeliveries",
0, REG_DWORD, (char *)&Reg.dwMaxPendingDeliveries,
sizeof(Reg.dwMaxPendingDeliveries));

        RegSetValueEx(hKey, "DB_Protocol", 0,
REG_SZ, szDBNames[Reg.eDB_Protocol],
strlen(szDBNames[Reg.eDB_Protocol])+1);
        RegSetValueEx(hKey, "TxnMonitor", 0,
REG_SZ, szTxnMonNames[Reg.eTxnMon],
strlen(szTxnMonNames[Reg.eTxnMon])+1);

        RegSetValueEx(hKey, "DbServer", 0, REG_SZ,
Reg.szDbServer, strlen(Reg.szDbServer)+1);
        RegSetValueEx(hKey, "DbName", 0, REG_SZ,
Reg.szDbName, strlen(Reg.szDbName)+1);

```

```

        RegSetValueEx(hKey, "DbUser", 0, REG_SZ,
Reg.szDbUser, strlen(Reg.szDbUser)+1);
        RegSetValueEx(hKey, "DbPassword", 0,
REG_SZ, Reg.szDbPassword, strlen(Reg.szDbPassword)+1);

        strcpy(szTmp, "YES");
        RegSetValueEx(hKey, "COM_SinglePool", 0,
REG_SZ, szTmp, strlen(szTmp)+1);

        RegFlushKey(hKey);
        RegCloseKey(hKey);
    }

    if ( (iRc=RegCreateKeyEx(HKEY_LOCAL_MACHINE,
"SYSTEM\\CurrentControlSet\\Services\\Inetinfo\\Parameters", 0,
NULL, REG_OPTION_NON_VOLATILE, KEY_ALL_ACCESS,
NULL, &hKey, &dwDisposition)) == ERROR_SUCCESS )
    {
        RegSetValueEx(hKey, "PoolThreadLimit", 0,
REG_DWORD, (char *)&iPoolThreadLimit,
sizeof(iPoolThreadLimit));
        RegSetValueEx(hKey, "ThreadTimeout", 0,
REG_DWORD, (char *)&iThreadTimeout, sizeof(iThreadTimeout));
        RegSetValueEx(hKey, "ListenBackLog", 0,
REG_DWORD, (char *)&iListenBackLog, sizeof(iListenBackLog));

        RegFlushKey(hKey);
        RegCloseKey(hKey);
    }

    if ( (iRc=RegCreateKeyEx(HKEY_LOCAL_MACHINE,
"SYSTEM\\CurrentControlSet\\Services\\W3SVC\\Parameters", 0,
NULL, REG_OPTION_NON_VOLATILE, KEY_ALL_ACCESS,
NULL, &hKey, &dwDisposition)) == ERROR_SUCCESS )
    {
        RegSetValueEx(hKey, "AcceptExOutstanding",
0, REG_DWORD, (char *)&iAcceptExOutstanding,
sizeof(iAcceptExOutstanding));

        RegFlushKey(hKey);
        RegCloseKey(hKey);
    }

    return;
}

BOOL CALLBACK CopyDlgProc(HWND hwnd, UINT uMsg,
WPARAM wParam, LPARAM lParam)
{
    if ( uMsg == WM_INITDIALOG )
    {
        SendDlgItemMessage(hwnd,
IDC_PROGRESS1, PBM_SETRANGE, 0, MAKELPARAM(0, 15));
        SendDlgItemMessage(hwnd,
IDC_PROGRESS1, PBM_SETSTEP, (WPARAM)1, 0);
        return TRUE;
    }
    return FALSE;
}

BOOL RegisterDLL(char *szFileName)
{
    HINSTANCE    hLib;
    FARPROC      lpDllEntryPoint;

    hLib = LoadLibrary(szFileName);
    if ( hLib == NULL )

```

```

        return FALSE;
// Find the entry point.
lpDllEntryPoint = GetProcAddress(hLib,
"DllRegisterServer");
if (lpDllEntryPoint != NULL)
{
    return ((*lpDllEntryPoint)() == S_OK);
}
else
    return FALSE; //unable to locate entry
point
}

BOOL FileFromResource( char *szResourceName, int iResourceId,
char *szDllPath, char *szFileName )
{
    HGLOBAL          hDLL;
    HRSRC            hResInfo;
    HANDLE           hFile;
    DWORD            dwSize;
    BYTE             *pSrc;
    DWORD            d;
    char             szFullName[256];

    hResInfo = FindResource(hInst,
MAKEINTRESOURCE(iResourceId), szResourceName);

    strcpy(szFullName, szDllPath);
    strcat(szFullName, szFileName);

    dwSize = SizeofResource(hInst, hResInfo);
    hDLL = LoadResource(hInst, hResInfo);
    pSrc = (BYTE *)LockResource(hDLL);
    remove(szFullName);

    if ( !(hFile = CreateFile(szFullName, GENERIC_WRITE, 0,
NULL, CREATE_ALWAYS, FILE_ATTRIBUTE_NORMAL,
NULL)) )
        return FALSE;

    if ( !WriteFile(hFile, pSrc, dwSize, &d, NULL) )
        return FALSE;

    CloseHandle(hFile);

    UnlockResource(hDLL);
    FreeResource(hDLL);
    return TRUE;
}

static int CopyFiles(HWND hDlg, char *szDllPath)
{
    BOOL            bSvcRunning;

    bSvcRunning = CheckWWWService();
    if ( bSvcRunning )
    {
        SetDlgItemText(hDlg, IDC_STATUS, "Stopping
Web Service.");
        SendDlgItemMessage(hDlg, IDC_PROGRESS1,
PBM_STEPIT, 0, 0);
        UpdateDialog(hDlg);

        StopWWWService();
        SendDlgItemMessage(hDlg, IDC_PROGRESS1,
PBM_STEPIT, 0, 0);
        UpdateDialog(hDlg);
    }
}

```

```

        SetDlgItemText(hDlg, IDC_STATUS, "Copying Files...");
        SendDlgItemMessage(hDlg, IDC_PROGRESS1,
PBM_STEPIT, 0, 0);
        UpdateDialog(hDlg);

        // install TPCC.DLL
        strcpy( szLastFileName, "tpcc.dll" );
        if (!FileFromResource( "TPCCDLL", IDR_TPCCDLL,
szDllPath, szLastFileName ))
            return 0;
        SendDlgItemMessage(hDlg, IDC_PROGRESS1,
PBM_STEPIT, 0, 0);
        UpdateDialog(hDlg);

        // install tpcc_dblib.dll
        strcpy( szLastFileName, "tpcc_dblib.dll" );
        if (!FileFromResource( "DBLIB_DLL", IDR_DBLIB_DLL,
szDllPath, szLastFileName ))
            return 0;
        SendDlgItemMessage(hDlg, IDC_PROGRESS1,
PBM_STEPIT, 0, 0);
        UpdateDialog(hDlg);

        // install tpcc_odbc.dll
        strcpy( szLastFileName, "tpcc_odbc.dll" );
        if (!FileFromResource( "ODBC_DLL", IDR_ODBC_DLL,
szDllPath, szLastFileName ))
            return 0;
        SendDlgItemMessage(hDlg, IDC_PROGRESS1,
PBM_STEPIT, 0, 0);
        UpdateDialog(hDlg);

        /*
        // install tuxapp.exe
        strcpy( szLastFileName, "tuxapp.exe" );
        if (!FileFromResource( "TUXEDO_APP",
IDR_TUXEDO_APP, szDllPath, szLastFileName ))
            return 0;
        SendDlgItemMessage(hDlg, IDC_PROGRESS1,
PBM_STEPIT, 0, 0);
        UpdateDialog(hDlg);

        // install tpcc_tuxedo.dll
        strcpy( szLastFileName, "tpcc_tuxedo.dll" );
        if (!FileFromResource( "TUXEDO_DLL",
IDR_TUXEDO_DLL, szDllPath, szLastFileName ))
            return 0;
        SendDlgItemMessage(hDlg, IDC_PROGRESS1,
PBM_STEPIT, 0, 0);
        UpdateDialog(hDlg);

        /*
        // install tpcc_com.dll
        strcpy( szLastFileName, "tpcc_com.dll" );
        if (!FileFromResource( "COM_DLL", IDR_COM_DLL,
szDllPath, szLastFileName ))
            return 0;
        SendDlgItemMessage(hDlg, IDC_PROGRESS1,
PBM_STEPIT, 0, 0);
        UpdateDialog(hDlg);

        // install tpcc_com_all.tlb
        strcpy( szLastFileName, "tpcc_com_all.tlb" );
        if (!FileFromResource( "COM_TYPLIB",
IDR_COMTYPLIB_DLL, szDllPath, szLastFileName ))
            return 0;
        SendDlgItemMessage(hDlg, IDC_PROGRESS1,
PBM_STEPIT, 0, 0);
        UpdateDialog(hDlg);

```

```

// install tpcc_com_ps.dll
strcpy( szLastFileName, "tpcc_com_ps.dll" );
if (!FileFromResource("COM_PS_DLL",
IDR_COMPS_DLL, szDllPath, szLastFileName))
    return 0;
SendDlgItemMessage(hDlg, IDC_PROGRESS1,
PBM_STEPIT, 0, 0);
UpdateDialog(hDlg);

// install tpcc_com_all.dll
strcpy( szLastFileName, "tpcc_com_all.dll" );
if (!FileFromResource("COM_ALL_DLL",
IDR_COMALL_DLL, szDllPath, szLastFileName))
    return 0;
SendDlgItemMessage(hDlg, IDC_PROGRESS1,
PBM_STEPIT, 0, 0);
UpdateDialog(hDlg);

// install tpcc_com_remote.tlb
strcpy( szLastFileName, "tpcc_com_remote.tlb" );
if (!FileFromResource("COM_REMTYPLIB",
IDR_COMREMTYPLIB_DLL, szDllPath, szLastFileName))
    return 0;
SendDlgItemMessage(hDlg, IDC_PROGRESS1,
PBM_STEPIT, 0, 0);
UpdateDialog(hDlg);

// install tpcc_com_remote_ps.dll
strcpy( szLastFileName, "tpcc_com_remote_ps.dll" );
if (!FileFromResource("COM_REMPS_DLL",
IDR_COMREMPS_DLL, szDllPath, szLastFileName))
    return 0;
SendDlgItemMessage(hDlg, IDC_PROGRESS1,
PBM_STEPIT, 0, 0);
UpdateDialog(hDlg);

// install tpcc_com_remote.dll
strcpy( szLastFileName, "tpcc_com_remote.dll" );
if (!FileFromResource("COM_REMOTE_DLL",
IDR_REMOTE_DLL, szDllPath, szLastFileName))
    return 0;
SendDlgItemMessage(hDlg, IDC_PROGRESS1,
PBM_STEPIT, 0, 0);
UpdateDialog(hDlg);

//if we stopped service restart it.
if ( bSvcRunning )
{
    SetDlgItemText(hDlg, IDC_STATUS, "Starting
Web Service.");
    SendDlgItemMessage(hDlg, IDC_PROGRESS1,
PBM_STEPIT, 0, 0);
    UpdateDialog(hDlg);
    StartWWWebService();
}

SendDlgItemMessage(hDlg, IDC_PROGRESS1,
PBM_STEPIT, 0, 0);
UpdateDialog(hDlg);

return 1;
}

static BOOL GetInstallPath(char *szDllPath)
{
    HKEY hKey;

```

```

    BYTE szData[256];
    DWORD sv;
    BOOL bRc;
    int len;
    char *ptr;
    int iRc;

    szDllPath[0] = 0;
    bRc = TRUE;
    if ( RegOpenKeyEx(HKEY_LOCAL_MACHINE,
"SYSTEM\\CurrentControlSet\\Services\\W3SVC\\Parameters\\Virtual
Roots", 0, KEY_ALL_ACCESS, &hKey) == ERROR_SUCCESS )
    {
        sv = sizeof(szData);
        iRc = RegQueryValueEx( hKey, "/", NULL,
NULL, szData, &sv ); // used by IIS 3.0
        if (iRc == ERROR_FILE_NOT_FOUND)
            iRc = RegQueryValueEx( hKey, "/",
NULL, NULL, szData, &sv ); // used by IIS 4.0
        if (iRc == ERROR_SUCCESS)
        {
            bRc = FALSE;
            strcpy(szDllPath, szData);
            if ( ( ptr = strchr(szDllPath, '\\) ) )
                *ptr = 0;

            len = strlen(szDllPath);
            if ( szDllPath[len-1] != '\\ )
            {
                szDllPath[len] = '\\;
                szDllPath[len+1] = 0;
            }
            RegCloseKey(hKey);
        }
    }

    return bRc;
}

static void GetVersionInfo(char *szDLLPath, char *szExePath)
{
    DWORD d;
    DWORD dwSize;
    DWORD dwBytes;
    char *ptr;
    VS_FIXEDFILEINFO*vs;

    versionDIIMS = 0;
    versionDIILS = 0;
    if ( _access(szDLLPath, 00) == 0 )
    {
        dwSize = GetFileVersionInfoSize(szDLLPath,
&d);
        if ( dwSize )
        {
            ptr = (char *)malloc(dwSize);
            GetFileVersionInfo(szDLLPath, 0,
dwSize, ptr);
            VerQueryValue(ptr, "\\",&vs,
&dwBytes);
            versionDIIMS =
vs->dwProductVersionMS;
            versionDIILS =
vs->dwProductVersionLS;
            free(ptr);
        }
    }
}

```

```

versionExeMS = 0x7FFF;
versionExeLS = 0x7FFF;
dwSize = GetFileVersionInfoSize(szExePath, &d);
if ( dwSize )
{
    ptr = (char *)malloc(dwSize);
    GetFileVersionInfo(szExePath, 0, dwSize, ptr);
    VerQueryValue(ptr, "\\", &vs, &dwBytes);

    versionExeMS = vs->dwProductVersionMS;
    versionExeLS =
LOWORD(vs->dwProductVersionLS);
    versionExeMM =
HIWORD(vs->dwProductVersionLS);
    free(ptr);
}
return;
}

static BOOL CheckWWWebService(void)
{
    SC_HANDLE          schSCManager;
    SC_HANDLE          schService;
    SERVICE_STATUS    ssStatus;

    schSCManager = OpenSCManager(NULL, NULL,
SC_MANAGER_ALL_ACCESS);
    schService = OpenService(schSCManager,
TEXT("W3SVC"), SERVICE_ALL_ACCESS);
    if (schService == NULL)
        return FALSE;

    if (! QueryServiceStatus(schService, &ssStatus) )
        goto ServiceNotRunning;

    if ( !ControlService(schService,
SERVICE_CONTROL_STOP, &ssStatus) )
        goto ServiceNotRunning;
//start Service pending, Check the status until the service is
running.
    if (! QueryServiceStatus(schService, &ssStatus) )
        goto ServiceNotRunning;

    CloseServiceHandle(schService);
    return TRUE;

ServiceNotRunning:
    CloseServiceHandle(schService);
    return FALSE;
}

static BOOL StartWWWebService(void)
{
    SC_HANDLE          schSCManager;
    SC_HANDLE          schService;
    SERVICE_STATUS    ssStatus;
    DWORD              dwOldCheckPoint;

    schSCManager = OpenSCManager(NULL, NULL,
SC_MANAGER_ALL_ACCESS);
    schService = OpenService(schSCManager,
TEXT("W3SVC"), SERVICE_ALL_ACCESS);
    if (schService == NULL)
        return FALSE;

    if (! StartService(schService, 0, NULL) )

```

```

        goto StartWWWebErr;
//start Service pending, Check the status until the service is
running.
    if (! QueryServiceStatus(schService, &ssStatus) )
        goto StartWWWebErr;
    while( ssStatus.dwCurrentState != SERVICE_RUNNING)
    {
        dwOldCheckPoint = ssStatus.dwCheckPoint;
        //Save the current checkpoint.
        Sleep(ssStatus.dwWaitHint);
        //Wait for the specified
interval.
        if ( !QueryServiceStatus(schService, &ssStatus) )
//Check the status again.
            break;
        if (dwOldCheckPoint >=
ssStatus.dwCheckPoint)
//Break if the checkpoint
has not been incremented.
            break;
    }

    if (ssStatus.dwCurrentState == SERVICE_RUNNING)
        goto StartWWWebErr;

    CloseServiceHandle(schService);
    return TRUE;

StartWWWebErr:
    CloseServiceHandle(schService);
    return FALSE;
}

static BOOL StopWWWebService(void)
{
    SC_HANDLE          schSCManager;
    SC_HANDLE          schService;
    SERVICE_STATUS    ssStatus;
    DWORD              dwOldCheckPoint;

    schSCManager = OpenSCManager(NULL, NULL,
SC_MANAGER_ALL_ACCESS);
    schService = OpenService(schSCManager,
TEXT("W3SVC"), SERVICE_ALL_ACCESS);
    if (schService == NULL)
        return FALSE;

    if (! QueryServiceStatus(schService, &ssStatus) )
        goto StopWWWebErr;

    if ( !ControlService(schService,
SERVICE_CONTROL_STOP, &ssStatus) )
        goto StopWWWebErr;
//start Service pending, Check the status until the service is
running.
    if (! QueryServiceStatus(schService, &ssStatus) )
        goto StopWWWebErr;
    while( ssStatus.dwCurrentState == SERVICE_RUNNING)
    {
        dwOldCheckPoint = ssStatus.dwCheckPoint;
        //Save the current checkpoint.
        Sleep(ssStatus.dwWaitHint);
        //Wait for the specified
interval.
        if ( !QueryServiceStatus(schService, &ssStatus) )
//Check the status again.
            break;

```



```

        if (dwOldCheckPoint >=
ssStatus.dwCheckPoint) //Break if the checkpoint
has not been incremented.
            break;
    }

    if (ssStatus.dwCurrentState == SERVICE_RUNNING)
        goto StopWWWebErr;

    CloseServiceHandle(schService);
    return TRUE;

StopWWWebErr:
    CloseServiceHandle(schService);
    return FALSE;
}

static void UpdateDialog(HWND hDlg)
{
    MSG msg;

    UpdateWindow(hDlg);
    while( PeekMessage(&msg, hDlg, 0, 0, PM_REMOVE) )
    {
        TranslateMessage(&msg);
        DispatchMessage(&msg);
    }
    Sleep(250);
    return;
}

```

## install.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_ICON2           106
#define IDR_DELIVERY        107
#define IDD_DIALOG3        108

#define BN_LOG              1001
#define ED_KEEP             1002
#define ED_THREADS         1003
#define ED_THREADS2        1004
#define IDC_PATH           1007
#define IDC_VERSION        1009
#define IDC_RESULTS        1010
#define IDC_PROGRESS1      1011
#define IDC_STATUS         1012
#define IDC_BUTTON1        1013
#define ED_MAXCONNECTION   1014
#define ED_IIS_MAX_THREAD_POOL_LIMIT 1015
#define ED_WEB_SERVICE_BACKLOG_QUEUE_SIZE 1017
#define ED_IIS_THREAD_TIMEOUT 1018
#define ED_IIS_LISTEN_BACKLOG 1019
#define IDC_DBLIB          1021
#define IDC_ODBC           1022
#define IDC_CONNECT_POOL  1023

```

```

#define ED_USER_CONNECT_DELAY_TIME
1024

// Next default values for new objects
//

install.rc

//Microsoft Developer Studio generated resource script.
//
#include "resource.h"

#define APSTUDIO_READONLY_SYMBOLS
//
// Generated from the TEXTINCLUDE2 resource.
//
#include "afxres.h"

//
// English (U.S.) resources
//

#if !defined(AFX_RESOURCE_DLL) || defined(AFX_TARG_ENU)
#ifdef _WIN32
LANGUAGE LANG_ENGLISH, SUBLANG_ENGLISH_US
#pragma code_page(1252)
#endif // _WIN32

//
// Dialog
//

IDD_DIALOG1 DIALOGEX 0, 0, 367, 255
STYLE DS_MODALFRAME | DS_CENTER | WS_MINIMIZEBOX |
WS_POPUP | WS_CAPTION |
WS_VSCROLL | WS_SYSMENU
CAPTION "TPC-C Web Client Installation Utility"
FONT 8, "MS Sans Serif"
BEGIN
    EDITTEXT    ED_THREADS,134,36,34,12,ES_RIGHT |
ES_NUMBER,
        WS_EX_RTLREADING
    EDITTEXT    ED_MAXDELIVERIES,134,53,34,12,ES_RIGHT |
ES_NUMBER,
        WS_EX_RTLREADING
    EDITTEXT    ED_MAXCONNECTION,134,67,34,12,ES_RIGHT | ES_NUMBER,
        WS_EX_RTLREADING
    CONTROL
    "None",IDC_TM_NONE,"Button",BS_AUTORADIOBUTTON|
        WS_GROUP | WS_TABSTOP,43,100,33,10
    CONTROL
    "COM",IDC_TM_MTS,"Button",BS_AUTORADIOBUTTON|
        WS_TABSTOP,43,113,32,10
    CONTROL
    "TUXEDO",IDC_TM_TUXEDO,"Button",BS_AUTORADIOBUTTO
N |
        WS_TABSTOP,106,100,46,10
    CONTROL
    "ENCINA",IDC_TM_ENCINA,"Button",BS_AUTORADIOBUTTON
|

```

```

        WS_DISABLED| WS_TABSTOP,106,113,43,10
    EDITTEXT
ED_DB_SERVER,270,89,67,12,ES_AUTOHSCROLL
    EDITTEXT
ED_DB_USER_ID,270,103,67,12,ES_AUTOHSCROLL
    EDITTEXT
ED_DB_PASSWORD,270,114,67,12,ES_AUTOHSCROLL
    EDITTEXT
ED_DB_NAME,270,127,67,12,ES_AUTOHSCROLL
    CONTROL
"DBLIB",IDC_DBLIB,"Button",BS_AUTORADIOBUTTON|
WS_GROUP|
        WS_TABSTOP,214,164,39,12
    CONTROL
"ODBC",IDC_ODBC,"Button",BS_AUTORADIOBUTTON|
WS_TABSTOP,
        259,164,39,12
    EDITTEXT
ED_IIS_MAX_THREAD_POOL_LIMIT,127,148,41,12,ES_RIGHT|
        ES_NUMBER,WS_EX_RTREADING
    EDITTEXT

ED_WEB_SERVICE_BACKLOG_QUEUE_SIZE,127,162,41,12,ES_
RIGHT|
        ES_NUMBER,WS_EX_RTREADING
    EDITTEXT
ED_IIS_THREAD_TIMEOUT,127,178,41,12,ES_RIGHT|
ES_NUMBER,
        WS_EX_RTREADING
    EDITTEXT
ED_IIS_LISTEN_BACKLOG,127,193,41,12,ES_RIGHT|
ES_NUMBER,
        WS_EX_RTREADING
    DEFPUSHBUTTON "OK",IDOK,115,233,50,14
    PUSHBUTTON "Cancel",IDCANCEL,181,233,50,14
    EDITTEXT IDC_PATH,76,20,92,13,ES_AUTOHSCROLL|
ES_READONLY
    LTEXT "Number of Delivery
Threads:",IDC_STATIC,29,39,93,8
    LTEXT "Max Number of
Connections:",IDC_STATIC,29,67,93,8
    RTEXT "Version 4.22",IDC_VERSION,302,4,45,9
    LTEXT "IIS Max Thread Pool
Limit:",IDC_STATIC,36,148,86,12
    LTEXT "Web Service Backlog Queue
Size:",IDC_STATIC,35,159,76,
        15
    LTEXT "IIS Thread Timeout
(seconds):",IDC_STATIC,36,177,83,17
    LTEXT "IIS Listen Backlog:",IDC_STATIC,35,196,73,10
    GROUPBOX "Database
Interface",IDC_STATIC,192,151,146,27,WS_GROUP
    LTEXT "Installation directory:",IDC_STATIC,29,20,39,16
    GROUPBOX "Transaction
Monitor",IDC_STATIC,22,90,146,37
    LTEXT "Server Name:",IDC_STATIC,200,92,56,8
    LTEXT "User ID:",IDC_STATIC,200,105,60,8
    LTEXT "User Password:",IDC_STATIC,200,119,57,8
    LTEXT "Database Name:",IDC_STATIC,200,130,54,8
    GROUPBOX "SQL Server Connection
Properties",IDC_STATIC,192,69,150,
        116
    GROUPBOX "Web Client
Properties",IDC_STATIC,22,11,146,116
    GROUPBOX "IIS Settings",IDC_STATIC,22,132,146,79
    LTEXT "Max Pending Deliveries:",IDC_STATIC,29,53,80,8
END

```

```

IDD_DIALOG2DIALOGEX 0, 0, 117, 62
STYLE DS_SETFOREGROUND| DS_3DLOOK| DS_CENTER|
WS_POPUP| WS_BORDER
EXSTYLE WS_EX_STATICEDGE
FONT 12, "MS Sans Serif", 0, 0, 0x1
BEGIN
    DEFPUSHBUTTON "OK",IDOK,33,45,50,9
    CTEXT "HTML TPC-C Installation
Successful",IDC_RESULTS,7,22,
        102,18,0,WS_EX_CLIENTEDGE
    ICON
IDI_ICON2,IDC_STATIC,50,7,18,20,SS_REALSIZEIMAGE,
        WS_EX_TRANSPARENT
END

IDD_DIALOG3DIALOG DISCARDABLE 0, 0, 91, 40
STYLE DS_SYSMODAL| DS_MODALFRAME| DS_3DLOOK|
DS_CENTER| WS_CAPTION
CAPTION "Installing TPC-C Web Client"
FONT 12, "Arial Black"
BEGIN
    CONTROL
"Progress1",IDC_PROGRESS1,"msctls_progress32",WS_BORDER,
        7,20,77,13
    CTEXT "Static",IDC_STATUS,7,7,77,12,SS_SUNKEN
END

IDD_DIALOG4DIALOG DISCARDABLE 0, 0, 291, 202
STYLE DS_MODALFRAME| DS_CENTER| WS_POPUP|
WS_CAPTION| WS_SYSMENU
CAPTION "Client End User License"
FONT 8, "MS Sans Serif"
BEGIN
    EDITTEXT IDC_LICENSE,7,7,271,167,ES_MULTILINE|
ES_AUTOVSCROLL|
        ES_AUTOHSCROLL| ES_READONLY|
WS_VSCROLL| WS_HSCROLL
    DEFPUSHBUTTON "I & Agree",IDOK,87,181,50,14
    PUSHBUTTON "&Cancel",IDCANCEL,153,181,50,14
END

////////////////////////////////////
//
// DESIGNINFO
//

#ifdef APSTUDIO_INVOKED
GUIDELINES DESIGNINFO DISCARDABLE
BEGIN
    IDD_DIALOG1,DIALOG
    BEGIN
        LEFTMARGIN, 22
        RIGHTMARGIN, 347
        VERTGUIDE, 29
        VERTGUIDE, 168
        VERTGUIDE, 179
        VERTGUIDE, 342
        TOPMARGIN, 4
        BOTTOMMARGIN, 249
    END

    IDD_DIALOG2,DIALOG
    BEGIN
        LEFTMARGIN, 7
        RIGHTMARGIN, 109
        TOPMARGIN, 7
        BOTTOMMARGIN, 54
    END

```

```

END

IDD_DIALOG3, DIALOG
BEGIN
    LEFTMARGIN, 7
    RIGHTMARGIN, 84
    TOPMARGIN, 7
    BOTTOMMARGIN, 33
END

IDD_DIALOG4, DIALOG
BEGIN
    LEFTMARGIN, 7
    RIGHTMARGIN, 278
    TOPMARGIN, 7
    BOTTOMMARGIN, 195
END
END
#endif // APSTUDIO_INVOKED

#ifdef APSTUDIO_INVOKED
////////////////////////////////////
//
// TEXTINCLUDE
//

1 TEXTINCLUDE DISCARDABLE
BEGIN
    "resource.h\0"
END

2 TEXTINCLUDE DISCARDABLE
BEGIN
    "#include ""afxres.h""\r\n"
    "\0"
END

3 TEXTINCLUDE DISCARDABLE
BEGIN
    "\r\n"
    "\0"
END

#endif // APSTUDIO_INVOKED

////////////////////////////////////
//
// Icon
//

// Icon with lowest ID value placed first to ensure application icon
// remains consistent on all systems.
IDI_ICON1      ICON  DISCARDABLE  "icon1.ico"
IDI_ICON2      ICON  DISCARDABLE  "icon2.ico"

////////////////////////////////////
//
// TPCCDLL
//

#ifdef _DEBUG
IDR_TPCCDLL    TPCCDLL DISCARDABLE
"..\\..\\isapi_dll\\bin\\debug\\tpcc.dll"
#else
IDR_TPCCDLL    TPCCDLL DISCARDABLE
"..\\..\\isapi_dll\\bin\\release\\tpcc.dll"

```

```

#endif

#ifndef _MAC
////////////////////////////////////
//
// Version
//

VS_VERSION_INFO VERSIONINFO
FILEVERSION 0,4,22,0
PRODUCTVERSION 0,4,22,0
FILEFLAGSMASK 0x3fL
#ifdef _DEBUG
FILEFLAGS 0x1L
#else
FILEFLAGS 0x0L
#endif
FILEOS 0x40004L
FILETYPE 0x1L
FILESUBTYPE 0x0L
BEGIN
    BLOCK "StringFileInfo"
    BEGIN
        BLOCK "040904b0"
        BEGIN
            VALUE "Comments", "TPC-C Web Client Installer\0"
            VALUE "CompanyName", "Microsoft\0"
            VALUE "FileDescription", "install\0"
            VALUE "FileVersion", "0, 4, 30, 0\0"
            VALUE "InternalName", "install\0"
            VALUE "LegalCopyright", "Copyright © 2001\0"
            VALUE "LegalTrademarks", "\0"
            VALUE "OriginalFilename", "install.exe\0"
            VALUE "PrivateBuild", "\0"
            VALUE "ProductName", "Microsoft install\0"
            VALUE "ProductVersion", "0, 4, 30, 0\0"
            VALUE "SpecialBuild", "\0"
        END
    END
    BLOCK "VarFileInfo"
    BEGIN
        VALUE "Translation", 0x409, 1200
    END
END

#endif // !_MAC

////////////////////////////////////
//
// LICENSE
//

IDR_LICENSE1    LICENSE DISCARDABLE  "license.txt"

////////////////////////////////////
//
// DBLIB_DLL
//

#ifdef _DEBUG
IDR_DBLIB_DLL    DBLIB_DLL DISCARDABLE
"..\\..\\db_dblib_dll\\bin\\debug\\tpcc_dblib.dll"
#else
IDR_DBLIB_DLL    DBLIB_DLL DISCARDABLE
"..\\..\\db_dblib_dll\\bin\\release\\tpcc_dblib.dll"
#endif

```

```

////////////////////////////////////
//
// ODBC_DLL
//

#ifdef _DEBUG
IDR_ODBC_DLL      ODBC_DLL DISCARDABLE
"..\\..\\db_odbc_dll\\bin\\debug\\tpcc_odbc.dll"
#else
IDR_ODBC_DLL      ODBC_DLL DISCARDABLE
"..\\..\\db_odbc_dll\\bin\\release\\tpcc_odbc.dll"
#endif

////////////////////////////////////
//
// TUXEDO_APP
//

IDR_TUXEDO_APP    TUXEDO_APP DISCARDABLE
"..\\..\\tuxapp\\bin\\tuxapp.exe"

////////////////////////////////////
//
// TUXEDO_DLL
//

IDR_TUXEDO_DLL    TUXEDO_DLL DISCARDABLE
"..\\..\\tm_tuxedo_dll\\bin\\tpcc_tuxedo.dll"

////////////////////////////////////
//
// COM_DLL
//

#ifdef _DEBUG
IDR_COM_DLL       COM_DLL DISCARDABLE
"..\\..\\tm_com_dll\\bin\\debug\\tpcc_com.dll"
#else
IDR_COM_DLL       COM_DLL DISCARDABLE
"..\\..\\tm_com_dll\\bin\\release\\tpcc_com.dll"
#endif

////////////////////////////////////
//
// COM_PS_DLL
//

#ifdef _DEBUG
IDR_COMPS_DLL     COM_PS_DLL DISCARDABLE
"..\\..\\tpcc_com_ps\\bin\\debug\\tpcc_com_ps.dll"
#else
IDR_COMPS_DLL     COM_PS_DLL DISCARDABLE
"..\\..\\tpcc_com_ps\\bin\\release\\tpcc_com_ps.dll"
#endif

////////////////////////////////////
//
// COM_ALL_DLL
//

#ifdef _DEBUG
IDR_COMALL_DLL    COM_ALL_DLL DISCARDABLE
"..\\..\\tpcc_com_all\\bin\\debug\\tpcc_com_all.dll"
#else
IDR_COMALL_DLL    COM_ALL_DLL DISCARDABLE
"..\\..\\tpcc_com_all\\bin\\release\\tpcc_com_all.dll"
#endif

```

```

////////////////////////////////////
//
// COM_TYPLIB
//

IDR_COMTYPLIB_DLL COM_TYPLIB DISCARDABLE
"..\\..\\tpcc_com_all\\src\\tpcc_com_all.tlb"

////////////////////////////////////
//
// COM_REMOTE_DLL
//

#ifdef _DEBUG
IDR_REMOTE_DLL    COM_REMOTE_DLL DISCARDABLE
"..\\..\\tpcc_com_remote\\bin\\debug\\tpcc_com_remote.dll"
#else
IDR_REMOTE_DLL    COM_REMOTE_DLL DISCARDABLE
"..\\..\\tpcc_com_remote\\bin\\release\\tpcc_com_remote.dll"
#endif

////////////////////////////////////
//
// COM_REMYPLIB
//

IDR_COMREMYPLIB_DLL COM_REMYPLIB
DISCARDABLE "..\\..\\tpcc_com_remote\\src\\tpcc_com_remote.tlb"

////////////////////////////////////
//
// COM_REMPS_DLL
//

#ifdef _DEBUG
IDR_COMREMPS_DLL  COM_REMPS_DLL DISCARDABLE
"..\\..\\tpcc_com_remote_ps\\bin\\debug\\tpcc_com_remote_ps.dll"
#else
IDR_COMREMPS_DLL  COM_REMPS_DLL DISCARDABLE
"..\\..\\tpcc_com_remote_ps\\bin\\release\\tpcc_com_remote_ps.dll"
#endif

#endif // English (U.S.) resources

////////////////////////////////////
//
// Generated from the TEXTINCLUDE3 resource.
//

////////////////////////////////////
#endif // not APSTUDIO_INVOKED

install_com.cpp

/*      FILE:          INSTALL_COM.CPP
 *
4,20,000
 *
1999
 *      All Rights Reserved
 *
Microsoft TPC-C Kit Ver.
Copyright Microsoft,

```

```

*
*                                     not audited
*
*   PURPOSE:           installation code for COM
application for TPC-C Web Kit
*   Contact:   Charles Levine (clevine@microsoft.com)
*
* Change history:
*   4.20.000 - first version
*/

#define _WIN32_WINNT 0x0500

#include <comdef.h>
#include <comadmin.h>
#include <stdio.h>
#include <tchar.h>

extern "C"
{
    BOOL install_com(char *szDllPath);
}

BOOL install_com(char *szDllPath)
{
    ICOMAdminCatalog*      pCOMAdminCat =
NULL;
    ICatalogCollection*   pCatalogCollectionApp = NULL;
    ICatalogCollection*   pCatalogCollectionCo = NULL;
    ICatalogCollection*   pCatalogCollectionItf = NULL;
    ICatalogCollection*   pCatalogCollectionMethod = NULL;

    ICatalogObject*      pCatalogObjectApp =
NULL;
    ICatalogObject*      pCatalogObjectCo =
NULL;
    ICatalogObject*      pCatalogObjectItf =
NULL;
    ICatalogObject*      pCatalogObjectMethod =
NULL;

    _bstr_t                bstrTemp,
bstrTemp2, bstrTemp3, bstrTemp4,
bstrTemp5, bstrTemp6, bstrTemp7;
    _bstr_t                bstrDllPath =
szDllPath;
    _variant_t             vTmp, vKey;
    long                   lActProp,
ICount, ICountCo, ICountItf, ICountMethod;
    bool                   bTmp;
    bool
bRemote=FALSE;

    CoInitializeEx(NULL, COINIT_MULTITHREADED);

    HRESULT hr =
CoCreateInstance(CLSID_COMAdminCatalog,

        NULL,

        CLSCTX_INPROC_SERVER,

        IID_ICOMAdminCatalog,

        (void**) &pCOMAdminCat);

    if (!SUCCEEDED(hr)) goto Error;

```

```

bstrTemp = "Applications";

// Attempt to connect to "Applications" in the Catalog
hr = pCOMAdminCat->GetCollection(bstrTemp,

        (IDispatch**) &pCatalogCollectionApp);
if (!SUCCEEDED(hr)) goto Error;

// Attempt to load the "Applications" collection
hr = pCatalogCollectionApp->Populate();
if (!SUCCEEDED(hr)) goto Error;

hr = pCatalogCollectionApp->get_Count(&lCount);
if (!SUCCEEDED(hr)) goto Error;

// iterate through applications to delete existing "TPC-C"
application (if any)
while (lCount > 0)
{
    hr = pCatalogCollectionApp->get_Item(lCount -
1, (IDispatch**) &pCatalogObjectApp);
    if (!SUCCEEDED(hr)) goto Error;

    hr = pCatalogObjectApp->get_Name(&vTmp);
    if (!SUCCEEDED(hr)) goto Error;

    if (wcsncmp(vTmp.bstrVal, L"TPC-C"))
    {
        lCount--;
        continue;
    }
    else
    {
        hr =
pCatalogCollectionApp->Remove(lCount - 1);
        if (!SUCCEEDED(hr)) goto Error;
        break;
    }
}

hr = pCatalogCollectionApp->SaveChanges(&lActProp);
if (!SUCCEEDED(hr)) goto Error;

// add the new application
hr = pCatalogCollectionApp->Add((IDispatch**)
&pCatalogObjectApp);
if (!SUCCEEDED(hr)) goto Error;

// set properties
bstrTemp = "Name";
vTmp = "TPC-C";
hr = pCatalogObjectApp->put_Value(bstrTemp, vTmp);
if (!SUCCEEDED(hr)) goto Error;

// set as a library (in process) application
bstrTemp = "Activation";
lActProp = COMAdminActivationInproc;
vTmp = lActProp;
hr = pCatalogObjectApp->put_Value(bstrTemp, vTmp);
if (!SUCCEEDED(hr)) goto Error;

// set security level to process
bstrTemp = "AccessChecksLevel";
lActProp = COMAdminAccessChecksApplicationLevel;
vTmp = lActProp;
hr = pCatalogObjectApp->put_Value(bstrTemp, vTmp);
if (!SUCCEEDED(hr)) goto Error;

```

```

// save key to get the Components collection later
hr = pCatalogObjectApp->get_Key(&vKey);
if (!SUCCEEDED(hr)) goto Error;

// save changes (app creation) so component installation
will work
hr = pCatalogCollectionApp->SaveChanges(&lActProp);
if (!SUCCEEDED(hr)) goto Error;

pCatalogObjectApp->Release();
pCatalogObjectApp = NULL;

bstrTemp = "TPC-C";
// app name
bstrTemp2 = bstrDllPath + "tpcc_com_all.dll";
// DLL
bstrTemp3 = bstrDllPath + "tpcc_com_all.tlb";
// type library
(TLB)
bstrTemp4 = bstrDllPath + "tpcc_com_ps.dll";
// proxy/stub dll
bstrTemp2 = bstrDllPath +
"tpcc_com_all.dll"; // DLL
bstrTemp5 = bstrDllPath + "tpcc_com_remote.dll";
bstrTemp6 = bstrDllPath + "tpcc_com_remote.tlb";
bstrTemp7 = bstrDllPath + "tpcc_com_remote_ps.dll";

hr = pCOMAdminCat->InstallComponent(bstrTemp,
bstrTemp2,
bstrTemp3,
bstrTemp4);
if (!SUCCEEDED(hr)) goto Error;

hr = pCOMAdminCat->InstallComponent(bstrTemp,
bstrTemp5,
bstrTemp6,
bstrTemp7);
if (!SUCCEEDED(hr)) goto Error;

bstrTemp = "Components";
hr = pCatalogCollectionApp->GetCollection(bstrTemp,
vKey, (IDispatch**) &pCatalogCollectionCo);
if (!SUCCEEDED(hr)) goto Error;

hr = pCatalogCollectionCo->Populate();
if (!SUCCEEDED(hr)) goto Error;

hr = pCatalogCollectionCo->get_Count(&lCountCo);
if (!SUCCEEDED(hr)) goto Error;

// iterate through components in application and set the
properties
while (lCountCo > 0)
{
hr = pCatalogCollectionCo->get_Item(lCountCo
- 1, (IDispatch**) &pCatalogObjectCo);
if (!SUCCEEDED(hr)) goto Error;

// used for debugging (view the name)
hr = pCatalogObjectCo->get_Name(&vTmp);
if (!SUCCEEDED(hr)) goto Error;

```

```

if
(!wcsncmp(vTmp.bstrVal.L"TPCC.AllRemoteTxns.1"))
bRemote=TRUE;
else
bRemote=FALSE;

bstrTemp = "ConstructionEnabled";
bTmp = TRUE;
vTmp = bTmp;
hr = pCatalogObjectCo->put_Value(bstrTemp,
vTmp);

if (!SUCCEEDED(hr)) goto Error;

bstrTemp = "ConstructorString";
bstrTemp2 = "dummy string (do not remove)";
vTmp = bstrTemp2;
hr = pCatalogObjectCo->put_Value(bstrTemp,
vTmp);

if (!SUCCEEDED(hr)) goto Error;

bstrTemp = "JustInTimeActivation";
bTmp = TRUE;
vTmp = bTmp;
hr = pCatalogObjectCo->put_Value(bstrTemp,
vTmp);

if (!SUCCEEDED(hr)) goto Error;

bstrTemp = "MaxPoolSize";
vTmp.Clear(); // clear variant so it isn't
stored as a bool (_variant_t feature)
if (!bRemote)
{
vTmp = (long)30;
}
else
{
vTmp = (long)5;
}

hr = pCatalogObjectCo->put_Value(bstrTemp,
vTmp);

if (!SUCCEEDED(hr)) goto Error;

bstrTemp = "CreationTimeout";
vTmp.Clear(); // clear variant so it isn't
stored as a bool (_variant_t feature)
vTmp = (long)300000;
hr = pCatalogObjectCo->put_Value(bstrTemp,
vTmp);

if (!SUCCEEDED(hr))
{
goto Error;
}

bstrTemp = "MinPoolSize";
vTmp.Clear(); // clear variant so it isn't
stored as a bool (_variant_t feature)
if (!bRemote)
{
vTmp = (long)1;
}
else
{
vTmp = (long)1;
}

hr = pCatalogObjectCo->put_Value(bstrTemp,
vTmp);

```

```

        if (!SUCCEEDED(hr))
        {
            goto Error;
        }

        bstrTemp = "ObjectPoolingEnabled";
        vTmp.Clear(); // clear variant so it isn't
stored as a bool (_variant_t feature)
        bTmp = TRUE;
        vTmp = bTmp;
vTmp);
        hr = pCatalogObjectCo->put_Value(bstrTemp,

        if (!SUCCEEDED(hr)) goto Error;

        bstrTemp = "EventTrackingEnabled";
        vTmp.Clear();
        bTmp = FALSE;
        vTmp = bTmp;
vTmp);
        hr = pCatalogObjectCo->put_Value(bstrTemp,

        if (!SUCCEEDED(hr))
        {
            goto Error;
        }

        // save key to get the InterfacesForComponent
collection
        hr = pCatalogObjectCo->get_Key(&vKey);
        if (!SUCCEEDED(hr)) goto Error;

        bstrTemp = "InterfacesForComponent";
        hr =
pCatalogCollectionCo->GetCollection(bstrTemp, vKey, (IDispatch**)
&pCatalogCollectionItf);
        if (!SUCCEEDED(hr)) goto Error;

        hr = pCatalogCollectionItf->Populate();
        if (!SUCCEEDED(hr)) goto Error;

        hr =
pCatalogCollectionItf->get_Count(&lCountItf);
        if (!SUCCEEDED(hr)) goto Error;

        // iterate through interfaces in component
        while (lCountItf > 0)
        {
            hr =
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";
            hr =
pCatalogCollectionItf->GetCollection(bstrTemp, vKey, (IDispatch**)
&pCatalogCollectionMethod);
            if (!SUCCEEDED(hr)) goto Error;

            hr =
pCatalogCollectionMethod->Populate();
            if (!SUCCEEDED(hr)) goto Error;

```

```

        hr =
pCatalogCollectionMethod->get_Count(&lCountMethod);
        if (!SUCCEEDED(hr)) goto Error;

        // iterate through methods of interface
        while (lCountMethod > 0)
        {
            hr =
pCatalogCollectionMethod->get_Item(lCountMethod - 1,
(IDispatch**) &pCatalogObjectMethod);
            if (!SUCCEEDED(hr))
                goto Error;

            bstrTemp =
"AutoComplete";
            vTmp.Clear(); //
clear variant so it isn't stored as a bool (_variant_t feature)

            bTmp = TRUE;
            vTmp = bTmp;
            hr =
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;
}

```

## install\_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_ICON2           106
#define IDR_DELIVERY        107
#define IDD_DIALOG3         108
#define IDR_LICENSE1        112
#define IDD_DIALOG4         113
#define IDR_TPCCOBJ1        117
#define IDR_TPCCSTUB1       118
#define IDR_DBLIB_DLL       122
#define IDR_ODBC_DLL        123
#define IDR_TUXEDO_APP      124
#define IDR_TUXEDO_DLL      125
#define IDR_COM_DLL         126
#define IDR_COMPS_DLL       127
#define IDR_COMALL_DLL      128
#define IDR_COMTYPLIB_DLL   129
#define IDR_REMOTE_DLL      131
#define IDR_COMREMTYPLIB_DLL 132
#define IDR_COMREMPS_DLL    133
#define BN_LOG               1001
#define ED_KEEP              1002
#define ED_THREADS          1003
#define ED_THREADS2         1004

```

```

#define IDC_PATH             1007
#define IDC_VERSION         1009
#define IDC_RESULTS         1010
#define IDC_PROGRESS1       1011
#define IDC_STATUS          1012
#define IDC_BUTTON1         1013
#define ED_MAXCONNECTION    1014
#define ED_IIS_MAX_THREAD_POOL_LIMIT 1015
#define ED_MAXDELIVERIES    1016
#define ED_WEB_SERVICE_BACKLOG_QUEUE_SIZE 1017
#define ED_IIS_THREAD_TIMEOUT 1018
#define ED_IIS_LISTEN_BACKLOG 1019
#define ED_MAXWAREHOUSES   1020
#define IDC_DBLIB           1021
#define IDC_LICENSE         1022
#define IDC_ODBC            1022
#define IDC_CONNECT_POOL   1023
#define ED_DB_SERVER        1023
#define ED_USER_CONNECT_DELAY_TIME 1024
#define ED_DB_USER_ID      1024
#define IDC_MTS             1025
#define IDC_TM_MTS          1025
#define IDC_TM_TUXEDO       1026
#define IDC_TM_NONE         1027
#define ED_DB_PASSWORD      1028
#define ED_DB_NAME          1029
#define IDC_TM_ENCINA       1030
#define ED_WAREHOUSE_PATH  1032

// Next default values for new objects
//
#ifdef APSTUDIO_INVOKED
#ifndef APSTUDIO_READONLY_SYMBOLS
#define _APS_NEXT_RESOURCE_VALUE 134
#define _APS_NEXT_COMMAND_VALUE 40001
#define _APS_NEXT_CONTROL_VALUE 1032
#define _APS_NEXT_SYMED_VALUE 101
#endif
#endif

```

## isapi\_dll\_resource.h

```

//{{NO_DEPENDENCIES}}
// Microsoft Developer Studio generated include file.
// Used by tpcc.rc
//
#define IDD_DIALOG1          101

// Next default values for new objects
//
#ifdef APSTUDIO_INVOKED
#ifndef APSTUDIO_READONLY_SYMBOLS
#define _APS_NEXT_RESOURCE_VALUE 102
#define _APS_NEXT_COMMAND_VALUE 40001
#define _APS_NEXT_CONTROL_VALUE 1000
#define _APS_NEXT_SYMED_VALUE 101
#endif
#endif

```

## 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 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 NON-INFRINGEMENT.

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 limitée suivante vous concerne:

**EXCLUSION DE GARANTIES.** Microsoft renonce entièrement ... 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 qualité, marchande ou un usage particulier. Le risque total d'écoulement de l'utilisation ou de la performance du LOGICIEL est entre vos mains.

**RESPONSABILITÉ LIMITÉE.** La seule obligation de Microsoft et votre recours exclusif concernant ce contrat n'excéderont pas cinq dollars (US\$5.00).

**ABSENCE DE RESPONSABILITÉ POUR LES DOMMAGES INDIRECTS.**

Microsoft ou ses fournisseurs ne pourront être tenus responsables en aucune circonstance de tout dommage quel qu'il soit (y compris mais non de façon limitative les dommages directs ou indirects causés par la perte de bénéfices commerciaux, l'interruption des affaires, la perte d'information commerciale ou toute autre perte pécuniaire) résultant de l'utilisation ou de l'impossibilité d'utilisation de ce produit, et ce, même si la société, Microsoft a, à l'avance, avisé de l'éventualité de tels dommages. Certains États/juridictions ne permettent pas l'exclusion ou la limitation de responsabilité, relative aux dommages indirects ou consécutifs, et la limitation ci-dessus peut ne pas s'appliquer ... votre regard. La présente Convention est régie par les lois de la province d'Ontario, Canada. Chacune des parties ... la présente reconnaît et irrevocablement la compétence des tribunaux de la province d'Ontario et consent ... instituer tout litige qui pourrait découler de la présente auprès des tribunaux situés dans le district judiciaire de York, province d'Ontario. Au cas où vous auriez des questions concernant cette licence ou que vous désiriez 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 écrire ...: Microsoft Customer Sales and Service, One Microsoft Way, Redmond, Washington 98052 6399.

**methods.h**

```
/* FILE: METHODS.H
 * Microsoft TPC-C Kit Ver.
4.20.000
 * Copyright Microsoft,
1999
 * All Rights Reserved
 *
 * not yet audited
 *
 * PURPOSE: Header file for COM components.
 *
 * Change history:
 * 4.20.000 - first version
 */
```

```
enum COMPONENT_ERROR
{
    ERR_MISSING_REGISTRY_ENTRIES = 1,
```

```
    ERR_LOADDLL_FAILED,
    ERR_GETPROCADDR_FAILED,
    ERR_UNKNOWN_DB_PROTOCOL
};

class CCOMPONENT_ERR : public CBaseErr
{
public:
    CCOMPONENT_ERR(COMPONENT_ERROR
Err)
    {
        m_Error = Err;
        m_szTextDetail = NULL;
        m_SystemErr = 0;
        m_szErrorText = NULL;
    };

    CCOMPONENT_ERR(COMPONENT_ERROR
Err, char *szTextDetail, DWORD dwSystemErr)
    {
        m_Error = Err;
        m_szTextDetail = new
char[strlen(szTextDetail)+1];
strcpy( m_szTextDetail, szTextDetail
);
        m_SystemErr = dwSystemErr;
        m_szErrorText = NULL;
    };

    ~CCOMPONENT_ERR()
    {
        if (m_szTextDetail != NULL)
            delete [] m_szTextDetail;
        if (m_szErrorText != NULL)
            delete [] m_szErrorText;
    };

    COMPONENT_ERROR m_Error;
    char
    *m_szTextDetail;
    char
    *m_szErrorText;
    DWORD m_SystemErr;

    int ErrorType() {return
ERR_TYPE_COMPONENT;};
    int ErrorNum() {return m_Error;};
    char *ErrorText();
};

static void WriteMessageToEventLog(LPTSTR lpszMsg);

////////////////////////////////////
// CTPCC_Common
class CTPCC_Common :
public ITPCC,
public IObjectControl,
public IObjectConstruct,
public CComObjectRootEx<CComSingleThreadModel>
{
public:
    BEGIN_COM_MAP(CTPCC_Common)
        COM_INTERFACE_ENTRY(ITPCC)
        COM_INTERFACE_ENTRY(IObjectControl)
        COM_INTERFACE_ENTRY(IObjectConstruct)
    END_COM_MAP()
};
```

```

        CTPCC_Common();
        ~CTPCC_Common();

// ITPCC
public:
    HRESULT __stdcall NewOrder(          VARIANT
    txn_in, VARIANT* txn_out);
    HRESULT __stdcall Payment(          VARIANT
    txn_in, VARIANT* txn_out);
    HRESULT __stdcall Delivery(        VARIANT
    txn_in) {return E_NOTIMPL;}
    HRESULT __stdcall StockLevel(VARIANT txn_in,
    VARIANT* txn_out);
    HRESULT __stdcall OrderStatus(     VARIANT
    txn_in, VARIANT* txn_out);

        HRESULT __stdcall CallSetComplete();

// IObjectControl
    STDMETHODIMP_(BOOL) CanBePooled() { return
    m_bCanBePooled; }
    STDMETHODIMP Activate() { return S_OK; } //
we don't support COM Services transactions (no enlistment)
    STDMETHODIMP_(void) Deactivate() { /* nothing to do
*/ }

// IObjectConstruct
    STDMETHODIMP Construct(IDispatch * pUnk);

// helper methods
private:
    BOOL          m_bCanBePooled;
    CTPCC_BASE   *m_pTxn;

    struct COM_DATA
    {
        int retval;
        int error;
        union
        {
            NEW_ORDER_DATA
            PAYMENT_DATA
            DELIVERY_DATA
            STOCK_LEVEL_DATA
            ORDER_STATUS_DATA
        } u;
    };

};

////////////////////////////////////
// CTPCC
class CTPCC :
    public CTPCC_Common,
    public CComCoClass<CTPCC, &CLSID_TPCC>
{
public:
    DECLARE_REGISTRY_RESOURCEID(IDR_TPCC)

    BEGIN_COM_MAP(CTPCC)
        COM_INTERFACE_ENTRY2(IUnknown,
        CComObjectRootEx)
    END_COM_MAP()
};

```

```

        COM_INTERFACE_ENTRY_CHAIN(CTPCC_Common)
    END_COM_MAP()
};

////////////////////////////////////
// CNewOrder
class CNewOrder :
    public CTPCC_Common,
    public CComCoClass<CNewOrder, &CLSID_NewOrder>
{
public:
    DECLARE_REGISTRY_RESOURCEID(IDR_NEWORDER)

    BEGIN_COM_MAP(CNewOrder)
        COM_INTERFACE_ENTRY2(IUnknown,
        CComObjectRootEx)
        COM_INTERFACE_ENTRY_CHAIN(CTPCC_Common)
    END_COM_MAP()

// ITPCC
public:
//      HRESULT __stdcall NewOrder(          int* iSize,
    UCHAR** txn) {return E_NOTIMPL;}
    HRESULT __stdcall Payment(          VARIANT
    txn_in, VARIANT* txn_out) {return E_NOTIMPL;}
    HRESULT __stdcall StockLevel(VARIANT txn_in,
    VARIANT* txn_out) {return E_NOTIMPL;}
    HRESULT __stdcall OrderStatus(     VARIANT
    txn_in, VARIANT* txn_out) {return E_NOTIMPL;}
};

////////////////////////////////////
// COrderStatus
class COrderStatus :
    public CTPCC_Common,
    public CComCoClass<COrderStatus,
    &CLSID_OrderStatus>
{
public:
    DECLARE_REGISTRY_RESOURCEID(IDR_ORDERSTATUS)

    BEGIN_COM_MAP(COrderStatus)
        COM_INTERFACE_ENTRY2(IUnknown,
        CComObjectRootEx)
        COM_INTERFACE_ENTRY_CHAIN(CTPCC_Common)
    END_COM_MAP()

// ITPCC
public:
    HRESULT __stdcall NewOrder(VARIANT txn_in,
    VARIANT* txn_out) {return E_NOTIMPL;}
    HRESULT __stdcall Payment(VARIANT txn_in,
    VARIANT* txn_out) {return E_NOTIMPL;}
    HRESULT __stdcall StockLevel(VARIANT txn_in,
    VARIANT* txn_out) {return E_NOTIMPL;}
//      HRESULT __stdcall OrderStatus(          int* iSize,
    UCHAR** txn) {return E_NOTIMPL;}
};

////////////////////////////////////
// CPayment
class CPayment :
    public CTPCC_Common,
    public CComCoClass<CPayment, &CLSID_Payment>
};

```

```

{
public:
DECLARE_REGISTRY_RESOURCEID(IDR_PAYMENT)

BEGIN_COM_MAP(CPayment)
    COM_INTERFACE_ENTRY2(IUnknown,
    CComObjectRootEx)
    COM_INTERFACE_ENTRY_CHAIN(CTPCC_Common)
END_COM_MAP()

// ITPCC
public:
    HRESULT __stdcall NewOrder(VARIANT txn_in,
    VARIANT* txn_out) {return E_NOTIMPL;}
//    HRESULT __stdcall Payment(          int* iSize,
    UCHAR** txn) {return E_NOTIMPL;}
    HRESULT __stdcall StockLevel(VARIANT txn_in,
    VARIANT* txn_out) {return E_NOTIMPL;}
    HRESULT __stdcall OrderStatus(VARIANT txn_in,
    VARIANT* txn_out) {return E_NOTIMPL;}
};

////////////////////////////////////
// CStockLevel
class CStockLevel :
    public CTPCC_Common,
    public CComCoClass<CStockLevel, &CLSID_StockLevel>
{
public:
DECLARE_REGISTRY_RESOURCEID(IDR_STOCKLEVEL)

BEGIN_COM_MAP(CStockLevel)
    COM_INTERFACE_ENTRY2(IUnknown,
    CComObjectRootEx)
    COM_INTERFACE_ENTRY_CHAIN(CTPCC_Common)
END_COM_MAP()

// ITPCC
public:
    HRESULT __stdcall NewOrder(VARIANT txn_in,
    VARIANT* txn_out) {return E_NOTIMPL;}
    HRESULT __stdcall Payment(VARIANT txn_in,
    VARIANT* txn_out) {return E_NOTIMPL;}
//    HRESULT __stdcall StockLevel(int* iSize, UCHAR** txn)
    {return E_NOTIMPL;}
    HRESULT __stdcall OrderStatus(VARIANT txn_in,
    VARIANT* txn_out) {return E_NOTIMPL;}
};

```

## ReadRegistry.cpp

```

/* FILE: READREGISTRY.CPP
 * Microsoft TPC-C Kit Ver.
4.20.000
 * Copyright Microsoft,
1999
 * All Rights Reserved
 *
 * not yet audited
 *
 * PURPOSE: Implementation for TPC-C Tuxedo
class.
 * Contact: Charles Levine (clevine@microsoft.com)
 *
 * 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 ( !strcmp(szTmp, szDBNames[ODBC]) )
            pReg->eDB_Protocol = ODBC;
        else if ( !strcmp(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 ( !strcmp(szTmp,
    szTxnMonNames[TUXEDO]) )
            pReg->eTxnMon = TUXEDO;
        else if ( !strcmp(szTmp,
    szTxnMonNames[ENCINA]) )
            pReg->eTxnMon = ENCINA;
        else if ( !strcmp(szTmp,
    szTxnMonNames[COM]) )
            pReg->eTxnMon = COM;
    }

    pReg->bCOM_SinglePool = FALSE;
    size = sizeof(szTmp);
    if ( RegQueryValueEx(hKey, "COM_SinglePool", 0, &type,
    (BYTE *)&szTmp, &size) == ERROR_SUCCESS )
    {
        if ( !strcmp(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
*
4.22.000      Microsoft TPC-C Kit Ver.
*
*          Copyright Microsoft,
2000
*      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 );

```

## ReadWHouse.cpp

```

/*      FILE:          ReadWHouse.cpp
*
4.20.000      Microsoft TPC-C Kit Ver.
*
*          Copyright Microsoft,
1999
*      All Rights Reserved
*
*      PURPOSE:      define entry points for
ReadWHouse.cpp
*      Contact:      Sally Martin (sallym@microsoft.com)
*
*      Change history:
*          4/25/00 - initial version
*/

#include <windows.h>
#include <stdio.h>

#include <assert.h>
#include <time.h>

#define DBNTWIN32
#include <sqltypes.h>
#include <sql.h>
#include <sqlext.h>
#include <odbcss.h>

#define SQLSUCCEEDED(rc) (rc == SQL_SUCCESS || rc ==
SQL_SUCCESS_WITH_INFO || rc == SQL_NO_DATA)

// forward decl of routine

```

```

static int ProcessError(SQLHANDLE henv, SQLHDBC hdbc,
SQLHSTMT hstmt);
static void WriteMessageToEventLog(LPTSTR lpszMsg);

// begin code

int GetWareHouseTable(void **ppWareHouseArray, // array of ints
int *piCount,
int
*piTotalWhs,
LPCSTR
szUser, // user name for login
LPCSTR
szPassword, // password for login
LPCSTR
szDSN // name of DSN to use
)
{
// this routine opens the local database and queries it for the
list
// of local warehouses
// it allocates space and builds an array of those warehouse
ids.
// first get the number of ids, and allocate the space required
// the second query gets the actual numbers
//SELECT W_ID FROM WAREHOUSE WHERE
NODENUMBER(W_ID)= CURRENT NODE ORDER BY W_ID
//SELECT COUNT (*) FROM WAREHOUSE WHERE
NODENUMBER(W_ID)= CURRENT NODE
// initialization
unsigned char szSQLStmt[1024];
RETCODE rc;
int
iNumWareHouses=0;
int iTTotalWhs=0;
int
iWareHouseNum=0;
int ii;
int *pArray =
NULL;
SQLHDBC hdbc =
SQL_NULL_HDBC;
SQLHSTMT hstmt =
SQL_NULL_HSTMT;
SQLHANDLE henv = 0;

rc=SQLSetEnvAttr(henv,
SQL_ATTR_CONNECTION_POOLING,
SQL_CP_OFF,
0);

if (!SQLSUCCEEDED(rc))
return ProcessError(henv,hdbc,hstmt);

if( SQLAllocEnv(&henv) != SQL_SUCCESS)
return ProcessError(henv,hdbc,hstmt);

if( SQLAllocConnect(henv, &hdbc) != SQL_SUCCESS)
return ProcessError(henv,hdbc,hstmt);

//
// Set AUTOCOMMIT OFF
//
if( SQLSetConnectOption(hdbc, SQL_AUTOCOMMIT,
SQL_AUTOCOMMIT_OFF) != SQL_SUCCESS)
return ProcessError(henv,hdbc,hstmt);

```

```

//
// set isolation
//
if ( SQLSetConnectAttr(hdbc,
SQL_ATTR_TXN_ISOLATION,
(SQLPOINTER)SQL_TXN_SERIALIZABLE,0) != SQL_SUCCESS )
return ProcessError(henv,hdbc,hstmt);

//
// Connect to db
//
{
char
szConnectStr[256];
char
szOutStr[1024];
SQLSMALLINT iOutStrLen;

sprintf( szConnectStr,
"UID=%s;PWD=%s;DSN=%s", szUser, szPassword, szDSN );

rc = SQLDriverConnect(hdbc, NULL,
(SQLCHAR*)szConnectStr, sizeof(szConnectStr),
(SQLCHAR*)szOutStr,
sizeof(szOutStr), &iOutStrLen, SQL_DRIVER_NOPROMPT);
if (rc != SQL_SUCCESS && rc !=
SQL_SUCCESS_WITH_INFO)
return ProcessError(henv,hdbc,hstmt);

}

// find out how large the array needs to be

rc = SQLAllocStmt(hdbc, &hstmt);
if (!SQLSUCCEEDED(rc)) return
ProcessError(henv,hdbc,hstmt);

// bind to the database
wsprintf((char *)szSQLStmt, "USE tpcc");
// exec the command
rc = SQLExecDirect(hstmt, szSQLStmt, SQL_NTS);
if (!SQLSUCCEEDED(rc)) return
ProcessError(henv,hdbc,hstmt);

wsprintf((char *)szSQLStmt, "SELECT COUNT (*) FROM
t_warehouse");

// bind the columns
rc = SQLBindCol(hstmt, 1, SQL_C_LONG,
&iNumWareHouses, 0, NULL);
if (!SQLSUCCEEDED(rc)) return
ProcessError(henv,hdbc,hstmt);

// exec the command
rc = SQLExecDirect(hstmt, szSQLStmt, SQL_NTS);
if (!SQLSUCCEEDED(rc)) return
ProcessError(henv,hdbc,hstmt);

rc = SQLFetch(hstmt);
if (!SQLSUCCEEDED(rc)) return
ProcessError(henv,hdbc,hstmt);

SQLFreeStmt(hstmt, SQL_CLOSE);
if (!SQLSUCCEEDED(rc))

```

```

    {
        return ProcessError(henv,hdbc,hstmt);
    }

    rc = SQLAllocStmt(hdbc, &hstmt);
    if (!SQLSUCCEEDED(rc))
    {
        return ProcessError(henv,hdbc,hstmt);
    }

    // get the total number of warehouses
    wsprintf((char *)szSQLStmt, "SELECT MAX (w_id)
FROM warehouse");

    // bind the columns
    rc = SQLBindCol(hstmt, 1, SQL_C_LONG, &iTotalWhs, 0,
NULL);
    if (!SQLSUCCEEDED(rc)) return
ProcessError(henv,hdbc,hstmt);

    // exec the command
    rc = SQLExecDirect(hstmt, szSQLStmt, SQL_NTS);
    if (!SQLSUCCEEDED(rc)) return
ProcessError(henv,hdbc,hstmt);

    rc = SQLFetch(hstmt);
    if (!SQLSUCCEEDED(rc)) return
ProcessError(henv,hdbc,hstmt);

    SQLFreeStmt(hstmt, SQL_CLOSE);
    if (!SQLSUCCEEDED(rc))
    {
        return ProcessError(henv,hdbc,hstmt);
    }

    if (piTotalWhs)
        *piTotalWhs = iTotWhs;

    // allocate the array
    pArray=(int *)malloc(iTotalWhs * sizeof(int));
    *ppWareHouseArray = (void *)pArray;

    // zero out array
    memset ((void*)pArray,0,iTotalWhs);

    // select the data and copy to the array

    rc = SQLAllocStmt(hdbc, &hstmt);
    if (!SQLSUCCEEDED(rc))
    {
        free(pArray);
        *ppWareHouseArray = NULL;
        return ProcessError(henv,hdbc,hstmt);
    }

    wsprintf((char *)szSQLStmt,
"SELECT w_id FROM t_warehouse ORDER
BY w_id");

    // bind the columns
    rc = SQLBindCol(hstmt, 1, SQL_C_LONG,
&iWareHouseNum, 0, NULL);
    if (!SQLSUCCEEDED(rc))
    {
        free(pArray);

```

```

        *ppWareHouseArray = NULL;
        return ProcessError(henv,hdbc,hstmt);
    }

    // exec the command
    rc = SQLExecDirect(hstmt, szSQLStmt, SQL_NTS);
    if (!SQLSUCCEEDED(rc))
    {
        free(pArray);
        *ppWareHouseArray = NULL;
        return ProcessError(henv,hdbc,hstmt);
    }

    // in a loop do the fetches
    // loop around the result set using fetch
    ii = 0;
    while ((rc = SQLFetch(hstmt)) != SQL_NO_DATA)
    {
        if (!SQLSUCCEEDED(rc))
        {
            free(pArray);
            *ppWareHouseArray = NULL;
            return ProcessError(henv,hdbc,hstmt);
        }
        pArray[ii] = iWareHouseNum;
        ii++;
    }

    if (!SQLSUCCEEDED(rc))
    {
        free(pArray);
        *ppWareHouseArray = NULL;
        return ProcessError(henv,hdbc,hstmt);
    }

    *piCount = ii;

    rc=SQLFreeStmt(hstmt, SQL_CLOSE);
    if (!SQLSUCCEEDED(rc)) return
ProcessError(henv,hdbc,hstmt);

    // commit all transactions
    rc = SQLEndTran(SQL_HANDLE_ENV,
henv,
SQL_COMMIT);

    if (!SQLSUCCEEDED(rc))
    {
        free(pArray);
        *ppWareHouseArray = NULL;
        return ProcessError(henv,hdbc,hstmt);
    }

    // cleanup

    rc=SQLDisconnect(hdbc);
    if (!SQLSUCCEEDED(rc)) return
ProcessError(henv,hdbc,hstmt);
    rc=SQLFreeHandle(SQL_HANDLE_DBC,hdbc);
    if (!SQLSUCCEEDED(rc)) return
ProcessError(henv,hdbc,hstmt);
    rc=SQLFreeEnv(henv);
    if (!SQLSUCCEEDED(rc)) return
ProcessError(henv,hdbc,hstmt);

```

```

        return (ERROR_SUCCESS);
    }

static int ProcessError(SQLHANDLE henv, SQLHDBC hdbc,
SQLHSTMT hstmt)
{
    RETCODE          rc;
    SDWORD           lNativeError;
    char             szState[6];
    char
szMsg[SQL_MAX_MESSAGE_LENGTH];
    char
szTmp[6*SQL_MAX_MESSAGE_LENGTH];

    szTmp[0] = 0;
    while (TRUE)
    {
        rc = SQLError(henv, hdbc, hstmt, (BYTE
*)&szState, &lNativeError,
                                (BYTE
*)&szMsg, sizeof(szMsg), NULL);
        if (rc == SQL_NO_DATA)
            break;

        // 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");
        strcat( szTmp, szMsg );
    }

    if (strlen(szTmp) > 0)
    {
        WriteMessageToEventLog( szTmp );
    }

    // cleanup
    SQLFreeStmt(hstmt, SQL_CLOSE);
    SQLDisconnect(hdbc);
    SQLFreeHandle(SQL_HANDLE_DBC, hdbc);

    return (E_FAIL);
}

static void WriteMessageToEventLog(LPTSTR lpszMsg)
{
    TCHAR  szMsg[256];
    HANDLE hEventSource;
    LPTSTR lpszStrings[2];

    // Use event logging to log the error.
    //
    hEventSource = RegisterEventSource(NULL,
TEXT("TPCC.DLL"));

    sprintf(szMsg, TEXT("Error in ReadWHouse.DLL: "));
    lpszStrings[0] = szMsg;
    lpszStrings[1] = lpszMsg;

    if (hEventSource != NULL)
    {

```

```

        ReportEvent(hEventSource, // handle of event source
EVENTLOG_ERROR_TYPE, // event type
0, // event category
0, // event ID
NULL, // current user's SID
2, // strings in lpszStrings
0, // no bytes of raw data
(LPCTSTR *)lpszStrings, // array of error strings
NULL); // no raw data

        (VOID) DeregisterEventSource(hEventSource);
    }
}

```

## ReadWHouse.h

```

/* FILE: ReadWHouse.h
 * Microsoft TPC-C Kit Ver.
4.20.000
 * Copyright Microsoft,
1999
 * All Rights Reserved
 *
 * PURPOSE: define entry points for
ReadWHouse.cpp
 * Contact: Sally Martin (sallym@microsoft.com)
 *
 * Change history:
 * 4/25/00 - initial version
 */

int GetWareHouseTable(void **ppWareHouseArray, // array of ints
int *piCount,
int
*piTotalWhs,
LPCSTR
szUser, // user name for login
LPCSTR
szPassword, // password for login
LPCSTR
szDSN // name of DSN to use
);

```

## rtetime.h

```

/* FILE: rtetime.h : header file
 * Copyright 1997 Microsoft Corp., All rights reserved.
 *
 * Authors: Charles Levine, Philip Durr
 * Microsoft Corp.
 */

#define MAX_JULIAN_TIME
0x7FFFFFFFFFFFFFFF
#define JULIAN_TIME __int64
#define TC_TIME DWORD
extern "C"
{
    BOOL InitJulianTime(LPSYSTEMTIME lpInitTime);
    JULIAN_TIME GetJulianTime(void);
    DWORD MyTickCount(void);
    void GetJulianAndTC(JULIAN_TIME *pJulian,
    DWORD *pTC);
    JULIAN_TIME ConvertTo64BitTime(int iYear, int iMonth, int
iDay, int iHour, int iMinute, int iSecond);
}

```



```

JULIAN_TIME      Get64BitTime(LPSYSTEMTIME lpInitTime);
int              JulianDay( int yr, int mm, int dd );
void            JulianToTime(JULIAN_TIME julianTS, int* yr,
int* mm, int* dd, int *hh, int *mi, int *ss );
void            JulianToCalendar( int day, int* yr, int* mm, int*
dd );
}

```

## SetAuditTime.cpp

```

/*      FILE:          ReadWHouse.cpp
*
*      4.20.000
*
*      Copyright Microsoft,
1999
*
*      All Rights Reserved
*
*
*      PURPOSE:       define entry points for
ReadWHouse.cpp
*      Contact:      Sally Martin (sallym@microsoft.com)
*
*      Change history:
*      4/25/00 - initial version
*/

#include <windows.h>
#include <stdio.h>

#include <stdlib.h> // db2

#include <assert.h>
#include <time.h>

#define DBNTWIN32
#define DB2NT
#include <sql.h>

#include <sqlcli.h>
#include <sqlcli1.h>

#define DB2OUT "C:\\TMP\\" //db2
#include "sqlenv.h" //db2
#include "lval.h" //db2
#include "sqlca.h" //db2

#define SQLSUCCEEDED(rc) (rc == SQL_SUCCESS || rc ==
SQL_SUCCESS_WITH_INFO || rc == SQL_NO_DATA)

// forward decl of routine
static int ProcessError(SQLHANDLE henv, SQLHDBC hdbc,
SQLHSTMT hstmt);
static void WriteMessageToEventLog(LPTSTR lpszMsg);
void ConvertTime(TIMESTAMP_STRUCT *tsTime, __int64 iTime);
__int64 TPCC_GetTime(void);
int SetAuditTime( LPCSTR szUser, // user name
for login
LPCSTR
szPassword, // password for login
LPCSTR
szDSN // name of DSN to use
);
// begin code
int main(int argc, TCHAR* argv[])
{

```

```

SetAuditTime("tpcc","tpcc","tpcc");
}
int SetAuditTime( LPCSTR szUser, // user name
for login
LPCSTR
szPassword, // password for login
LPCSTR
szDSN // name of DSN to use
)
{
// this routine opens the local database and queries it for the
list
// of local warehouses
// it allocates space and builds an array of those warehouse
ids.
// first get the number of ids, and allocate the space required
// the second query gets the actual numbers
//SELECT W_ID FROM WAREHOUSE WHERE
NODENUMBER(W_ID)= CURRENT NODE ORDER BY W_ID
//SELECT COUNT (*) FROM WAREHOUSE WHERE
NODENUMBER(W_ID)= CURRENT NODE
// initialization
unsigned char szSQLStmt[1024];
RETCODE rc;
int *pArray =
NULL;
SQLHDBC hdbc =
SQL_NULL_HDBC;
SQLHSTMT hstmt =
SQL_NULL_HSTMT;
SQLHANDLE henv = 0;
TIMESTAMP_STRUCT tsTime1;
__int64 Auditdate_time;

rc=SQLSetEnvAttr(henv,
SQL_ATTR_CONNECTION_POOLING,
SQL_CP_OFF,
0);

if (!SQLSUCCEEDED(rc))
return ProcessError(henv,hdbc,hstmt);

if( SQLAllocEnv(&henv) != SQL_SUCCESS)
return ProcessError(henv,hdbc,hstmt);

if( SQLAllocConnect(henv, &hdbc) != SQL_SUCCESS)
return ProcessError(henv,hdbc,hstmt);

//
// Set AUTOCOMMIT OFF
//
if( SQLSetConnectOption(hdbc, SQL_AUTOCOMMIT,
SQL_AUTOCOMMIT_OFF) != SQL_SUCCESS)
return ProcessError(henv,hdbc,hstmt);

//
// set isolation
//
if ( SQLSetConnectAttr(hdbc,
SQL_ATTR_TXN_ISOLATION,
(SQLPOINTER)SQL_TXN_SERIALIZABLE,0) != SQL_SUCCESS )
return ProcessError(henv,hdbc,hstmt);

//
// Connect to db
//
{

```

```

char
szConnectStr[256];
char
szOutStr[1024];
SQLSMALLINT          iOutStrLen;

sprintf( szConnectStr,
"UID=%s;PWD=%s;DSN=%s", szUser, szPassword, szDSN );

rc = SQLDriverConnect(hdbc, NULL,
(SQLCHAR*)szConnectStr, sizeof(szConnectStr),
(SQLCHAR*)szOutStr,
sizeof(szOutStr), &iOutStrLen, SQL_DRIVER_NOPROMPT);
if (rc != SQL_SUCCESS && rc !=
SQL_SUCCESS_WITH_INFO)
return ProcessError(henv,hdbc,hstmt);

if ( SQLSetConnectOption(hdbc,
SQL_CURSOR_HOLD,SQL_CURSOR_HOLD_OFF) !=
SQL_SUCCESS)
return ProcessError(henv,hdbc,hstmt);
}

rc = SQLAllocStmt(hdbc, &hstmt);
if (!SQLSUCCEEDED(rc))return
ProcessError(henv,hdbc,hstmt);
wsprintf((char *)szSQLStmt, "DELETE FROM AUDIT");
// exec the command
rc = SQLExecDirect(hstmt, szSQLStmt, SQL_NTS);
if (!SQLSUCCEEDED(rc))return
ProcessError(henv,hdbc,hstmt);

SQLFreeStmt(hstmt, SQL_CLOSE);
if (!SQLSUCCEEDED(rc))
{
return ProcessError(henv,hdbc,hstmt);
}

rc = SQLAllocStmt(hdbc, &hstmt);
if (!SQLSUCCEEDED(rc))return
ProcessError(henv,hdbc,hstmt);

wsprintf((char *)szSQLStmt, "INSERT INTO AUDIT
VALUES (?");
Auditdate_time = TPCC_GetTime();
// convert the date
ConvertTime(&tsTime1,Auditdate_time);

// bind the columns
rc = SQLBindCol(hstmt, 1, SQL_C_UBIGINT, &tsTime1,
0, NULL);
if (!SQLSUCCEEDED(rc))return
ProcessError(henv,hdbc,hstmt);

// exec the command
rc = SQLExecDirect(hstmt, szSQLStmt, SQL_NTS);
if (!SQLSUCCEEDED(rc))return
ProcessError(henv,hdbc,hstmt);

SQLFreeStmt(hstmt, SQL_CLOSE);
if (!SQLSUCCEEDED(rc))
{
return ProcessError(henv,hdbc,hstmt);
}
}

}

// commit all transactions
rc = SQLEndTran(SQL_HANDLE_ENV,
henv,
SQL_COMMIT);

if (!SQLSUCCEEDED(rc))
{
return ProcessError(henv,hdbc,hstmt);
}

// cleanup

rc=SQLDisconnect(hdbc);
if (!SQLSUCCEEDED(rc)) return
ProcessError(henv,hdbc,hstmt);
rc=SQLFreeHandle(SQL_HANDLE_DBC,hdbc);
if (!SQLSUCCEEDED(rc)) return
ProcessError(henv,hdbc,hstmt);
rc=SQLFreeEnv(henv);
if (!SQLSUCCEEDED(rc)) return
ProcessError(henv,hdbc,hstmt);

return (ERROR_SUCCESS);
}

static int ProcessError(SQLHANDLEhenv, SQLHDBC hdbc,
SQLHSTMT hstmt)
{
RETCODE          rc;
SDWORD          INativeError;
char            szState[6];
char
szMsg[SQL_MAX_MESSAGE_LENGTH];
char
szTmp[6*SQL_MAX_MESSAGE_LENGTH];

szTmp[0] = 0;
while (TRUE)
{
rc = SQLError(henv, hdbc, hstmt, (BYTE
*)&szState, &INativeError,
(BYTE
*)&szMsg, sizeof(szMsg), NULL);
if (rc == SQL_NO_DATA)
break;

// 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");
strcat( szTmp, szMsg );
}

if (strlen(szTmp) > 0)
{
}
}

```

```

        WriteMessageToEventLog(szTmp);
    }

    // cleanup
    SQLFreeStmt(hstmt, SQL_CLOSE);
    SQLDisconnect(hdbc);
    SQLFreeHandle(SQL_HANDLE_DBC, hdbc);

    return (E_FAIL);
}

static void WriteMessageToEventLog(LPTSTR lpszMsg)
{
    TCHAR szMsg[256];
    HANDLE hEventSource;
    LPTSTR lpszStrings[2];

    // Use event logging to log the error.
    //
    hEventSource = RegisterEventSource(NULL,
    TEXT("TPCC.DLL"));

    sprintf(szMsg, TEXT("Error in ReadWHouse.DLL: "));
    lpszStrings[0] = szMsg;
    lpszStrings[1] = lpszMsg;

    if (hEventSource != NULL)
    {
        ReportEvent(hEventSource, // handle of event source
        EVENTLOG_ERROR_TYPE, // event type
        0, // event category
        0, // event ID
        NULL, // current user's SID
        2, // strings in lpszStrings
        0, // no bytes of raw data
        (LPCTSTR *)lpszStrings, // array of error strings
        NULL); // no raw data

        (VOID) DeregisterEventSource(hEventSource);
    }
}

void ConvertTime(TIMESTAMP_STRUCT *tsTime, __int64 iTime)
{
    union
    {
        __int64 i64Date;
        FILETIME ftDate;
    } comboDate;

    SYSTEMTIME sysTime;

    comboDate.i64Date=iTime;

    BOOL ret = FileTimeToSystemTime(&comboDate.ftDate,
    &sysTime);

    tsTime->year = sysTime.wYear;
    tsTime->month = sysTime.wMonth;
    tsTime->day = sysTime.wDay;
    tsTime->hour = sysTime.wHour;
    tsTime->minute = sysTime.wMinute;
    tsTime->second = sysTime.wSecond;
    tsTime->fraction = sysTime.wMilliseconds/1000;

    return;
}

```

```

}

__int64 TPCC_GetTime(void)
{
    union
    {
        __int64 i64Date;
        FILETIME ftDate;
    } retValDate;

    SYSTEMTIME sysTime;

    GetLocalTime(&sysTime);

    SystemTimeToFileTime(&sysTime, &retValDate.ftDate);
    return retValDate.i64Date;
}

```

### spinlock.h

```

/*      FILE: SPINLOCK.H
 *
 * Copyright 1997 Microsoft Corp., All rights reserved.
 *
 * Authors: Mike Parkes, Charles Levine, Philip Durr
 *          Microsoft Corp.
 */

#ifndef _INC_Spinlock

    const LONG LockClosed      = 1;
    const LONG LockOpen        = 0;

    /**
     *
     * Spinlock and Semaphore locking.
     *
     * This class provides a very conservative locking scheme.
     *
     * The assumption behind the code is that locks will be
     * held for a very short time. When a lock is taken a
     * location is exchanged. All other threads that want this
     * lock wait by spinning and sometimes sleeping on a
     * semaphore
     * until it becomes free again. The only other choice is not
     * to wait at all and move on to do something else. This
     * module should normally be used in conjunction with
     * cache
     * aligned memory in minimize cache line misses.
     *
     */

    /**
     */

    class Spinlock
    {
        // Private data.
        HANDLE          Semaphore;
        volatile LONG   m_Spinlock;
        volatile LONG   Waiting;
    }

```

```

#ifdef _DEBUG
    // Counters for debugging builds.
    volatile LONG    TotalLocks;
    volatile LONG    TotalSleeps;
    volatile LONG    TotalSpins;
    volatile LONG    TotalWaits;
#endif

public:
    // Public functions.

    Spinlock( void );

    inline BOOL ClaimLock( BOOL
Wait = TRUE );

    inline void ReleaseLock( void );
    ~Spinlock( void );
    // Disabled operations.
    Spinlock( const Spinlock & Copy );
    void operator=( const Spinlock &
Copy );

private:
    // Private functions.
    inline BOOL ClaimSpinlock( volatile
LONG *s1 );

    void WaitForLock( void );
    void WakeAllSleepers( void );

};

/*****
 *
 * A guaranteed atomic exchange.
 *
 * An attempt is made to claim the Spinlock. This action
is
 * guaranteed to be atomic.
 *
 *****/

inline BOOL Spinlock::ClaimSpinlock( volatile LONG
*Spinlock )
{
#ifdef _DEBUG
    InterlockedIncrement( (LPLONG) &
TotalLocks );
#endif
    return ( (*Spinlock) == LockOpen ) &&
( InterlockedExchange( (LPLONG) Spinlock, LockClosed ) ==
LockOpen );
}

/*****
 *
 * Claim the Spinlock.
 *
 * Claim the lock if available else wait or exit.
 *
 *****/

```

```

inline BOOL Spinlock::ClaimLock( BOOL Wait )
{
    if ( ! ClaimSpinlock( ( volatile LONG* ) &
m_Spinlock ) )
    {
        if ( Wait )
            WaitForLock();
        return Wait;
    }
    return TRUE;
}

/*****
 *
 * Release the Spinlock.
 *
 * Release the lock and if needed wakeup any sleepers.
 *
 *****/

inline void Spinlock::ReleaseLock( void )
{
    m_Spinlock = LockOpen;
    if ( Waiting > 0 )
        WakeAllSleepers();
}

#define _INC_Spinlock

#endif

tpcc.cpp

/*      FILE:          TPCC.C
 *
 *      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:      Main module for TPCC.DLL which
is an ISAPI service dll.
 *
 *      Contact:     Charles Levine (clevine@microsoft.com)
 *
 *
 *      Change history:
 *
 *      4.20.000 - reworked error handling; added
options for COM and Encina txn monitors
 */

#include <windows.h>
#include <process.h>
#include <tchar.h>
#include <stdio.h>
#include <stdarg.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>

#ifdef ICECAP
#include <icapexp.h>
#endif

#include "..\..\common\src\trans.h" //tpckit
transaction header contains definitions of structures specific to TPC-C
#include "..\..\common\src\error.h"
#include "..\..\common\src\txn_base.h"
#include "..\..\common\src\ReadRegistry.h"

#include "..\..\common\txnlog\include\rtetime.h"
#include "..\..\common\txnlog\include\spinlock.h"
#include "..\..\common\txnlog\include\txnlog.h"

// Database layer includes
#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

// Txn monitor layer includes
#include "..\..\tm_com_dll\src\tpcc_com.h" //
COM Services implementation on TPC-C txns
#include "..\..\tm_tuxedo_dll\src\tpcc_tux.h" // interface to
Tuxedo libraries
#include "..\..\tm_encina_dll\src\tpcc_enc.h" // interface to
Encina libraries

#include "httpext.h" //ISAPI DLL
information header
#include "tpcc.h"
//this dlls specific structure, value e.t. header.

#define LEN_ERR_STRING 256

// defines for Make<Txn>Form calls to distinguish input and output
flavors
#define OUTPUT_FORM 0
#define INPUT_FORM 1
#define EOL 10

char
szMyComputerName[MAX_COMPUTERNAME_LENGTH+1];

//Terminal client id structure
TERM Term = { 0, 0, 0, NULL };

// The WEBCLIENT_VERSION string specifies the version level of
this web client interface.
// The RTE must be synchronized with the interface level on login,
otherwise the login
// will fail. This is a sanity check to catch problems resulting from
mismatched versions
// of the RTE and web client.
#define WEBCLIENT_VERSION "430"

static CRITICAL_SECTION
TermCriticalSection;

static HINSTANCE hLibInstanceTm = NULL;
static HINSTANCE hLibInstanceDb = NULL;

TYPE_CTPCC_DBLIB *pCTPCC_DBLIB_new;
TYPE_CTPCC_ODBC *pCTPCC_ODBC_new;
TYPE_CTPCC_TUXEDO *pCTPCC_TUXEDO_new;

TYPE_CTPCC_ENCINA *pCTPCC_ENCINA_new;
TYPE_CTPCC_ENCINA *pCTPCC_ENCINA_post_init;
TYPE_CTPCC_COM *pCTPCC_COM_new;

// For deferred Delivery txns:

CTxnLog *txnDelilog =
NULL; //used to log delivery transaction
information

HANDLE
hWorkerSemaphore = INVALID_HANDLE_VALUE;
HANDLE hDoneEvent
= INVALID_HANDLE_VALUE;
HANDLE *pDeliHandles
= NULL;

// configuration settings from registry
TPCCREGISTRYDATA Reg;

DWORD
dwNumDeliveryThreads = 4;
CRITICAL_SECTION DelBuffCriticalSection;
//critical section for delivery transactions cache
DELIVERY_TRANSACTION *pDelBuff =
NULL;
DWORD dwDelBuffSize
= 100; // size of circular buffer for delivery
txns
DWORD
dwDelBuffFreeCount; // number of
buffers free
DWORD
dwDelBuffBusyIndex = 0; // index position of entry
waiting to be delivered
DWORD
dwDelBuffFreeIndex = 0; // index position of unused
entry

#define MAXWHS 60001
BYTE
*pbLocalWhsTbl=NULL;
#ifdef COUNTSPLIT
LONG
gdwLocalNOCnt = 0;
LONG
gdwRemoteNOCnt = 0;
LONG
gdwLocalPayCnt = 0;
LONG
gdwRemotePayCnt = 0;
#endif

#include "..\..\common\src\ReadRegistry.cpp"
#include "ReadWHouse.h"

/* FUNCTION: DllMain
*
* PURPOSE: This function is the entry point for the DLL.
This implementation is based on the
fact that DLL_PROCESS_ATTACH
is only called from the inet service once.
*/

```

```

* ARGUMENTS:  HANDLE hModule
module handle
*
*           DWORD
ul_reason_for_call  reason for call
*
*           LPVOID lpReserved
reserved for future use
*
* RETURNS:    BOOL  FALSE
errors occurred in initialization
*
TRUE           DLL successfully
initialized
*/

BOOL APIENTRY DllMain(HANDLE hModule, DWORD
ul_reason_for_call, LPVOID lpReserved)
{
    DWORD i;
    char szEvent[LEN_ERR_STRING] = "\0";
    char szLogFile[128];
    char szDllName[128];
    BOOL  fComment = FALSE;
    int    cbSize=0;
    int    iWhs = 0;
    int *pWareHouseArray=NULL;
    int iCount = 0;
    int iLoop=0;
    int iTotalWhs=0;

    try
    {
        switch( ul_reason_for_call )
        {
            case DLL_PROCESS_ATTACH:
                DWORD
                dwSize = MAX_COMPUTERNAME_LENGTH+1;
                GetComputerName(szMyComputerName, &dwSize);
                szMyComputerName[dwSize] = 0;
        }

        DisableThreadLibraryCalls((HMODULE)hModule);
        InitializeCriticalSection(&TermCriticalSection);

        if (
        ReadTPCCRegistrySettings( &Reg ) )
            throw new
            CWEBCLNT_ERR( ERR_MISSING_REGISTRY_ENTRIES);

            dwDelBuffSize = min(
            Reg.dwMaxPendingDeliveries, 10000 ); // min with 10000 as a sanity
            constraint
            dwNumDeliveryThreads =
            min( Reg.dwNumberOfDeliveryThreads, 100 ); // min with 100 as a
            sanity constraint

            TermInit();
            // load DLL for txn
        monitor
            if (Reg.eTxnMon ==
            TUXEDO)
            {

```

```

strcpy(
szDllName, Reg.szPath );
strcat(
szDllName, "tpcc_tuxedo.dll");
hLibInstanceTm = LoadLibrary( szDllName );
if
(hLibInstanceTm == NULL)
throw new CWEBCLNT_ERR( ERR_LOADDLL_FAILED,
szDllName, GetLastError() );

// get function
pointer to wrapper for class constructor
pCTPCC_TUXEDO_new = (TYPE_CTPCC_TUXEDO*)
GetProcAddress(hLibInstanceTm,"CTPCC_TUXEDO_new");
if
(pCTPCC_TUXEDO_new == NULL)
throw new CWEBCLNT_ERR( ERR_GETPROCADDR_FAILED,
szDllName, GetLastError() );
}
else if (Reg.eTxnMon ==
ENCINA)
{
strcpy(
szDllName, Reg.szPath );
strcat(
szDllName, "tpcc_encina.dll");
hLibInstanceTm = LoadLibrary( szDllName );
if
(hLibInstanceTm == NULL)
throw new CWEBCLNT_ERR( ERR_LOADDLL_FAILED,
szDllName, GetLastError() );

// get function
pointer to wrapper for class constructor
pCTPCC_ENCINA_new = (TYPE_CTPCC_ENCINA*)
GetProcAddress(hLibInstanceTm,"CTPCC_ENCINA_new");
pCTPCC_ENCINA_post_init = (TYPE_CTPCC_ENCINA*)
GetProcAddress(hLibInstanceTm,"CTPCC_ENCINA_post_init");
if
(pCTPCC_ENCINA_new == NULL)
throw new CWEBCLNT_ERR( ERR_GETPROCADDR_FAILED,
szDllName, GetLastError() );
}
else if (Reg.eTxnMon ==
COM)
{
strcpy(
szDllName, Reg.szPath );
strcat(
szDllName, "tpcc_com.dll");
hLibInstanceTm = LoadLibrary( szDllName );
if
(hLibInstanceTm == NULL)
throw new CWEBCLNT_ERR( ERR_LOADDLL_FAILED,
szDllName, GetLastError() );

// get function
pointer to wrapper for class constructor

```

```

pCTPCC_COM_new = (TYPE_CTPCC_COM*)
GetProcAddress(hLibInstanceTm,"CTPCC_COM_new");
if
(pCTPCC_COM_new == NULL)

throw new CWEBCLNT_ERR( ERR_GETPROCADDR_FAILED,
szDllName, GetLastError());

GetWareHouseTable(
                (void
***)&pWareHouseArray, // array of ints
                &iCount,
                &iTotalWhs,
Reg.szDbUser,          // user name for login
Reg.szDbPassword,    // password for login
Reg.szDbName);       // name of DSN to use

pbLocalWhsTbl = (BYTE *)malloc(sizeof(BYTE)*iTotalWhs + 1);
if
(!pbLocalWhsTbl)

throw new CWEBCLNT_ERR( ERR_GETPROCADDR_FAILED,
szDllName, GetLastError());

memset((void*)pbLocalWhsTbl,0,sizeof(BYTE)*iTotalWhs + 1);

for
(iLoop=0;iLoop<iCount;iLoop++)
{
if
(pWareHouseArray[iLoop] < (int)iTotalWhs)
pbLocalWhsTbl[pWareHouseArray[iLoop]] = 1; //local warehouse
}

free((void*)pWareHouseArray);
}
// load DLL for database
connection
if ((Reg.eTxnMon ==
None) || (dwNumDeliveryThreads > 0))
{
if
(Reg.eDB_Protocol == DBLIB)
{
strcpy( szDllName, Reg.szPath );
strcat( szDllName, "tpcc_dblib.dll");
hLibInstanceDb = LoadLibrary( szDllName );

```

```

if
(hLibInstanceDb == NULL)
throw new CWEBCLNT_ERR( ERR_LOADDLL_FAILED,
szDllName, GetLastError());
//
get function pointer to wrapper for class constructor
pCTPCC_DBLIB_new = (TYPE_CTPCC_DBLIB*)
GetProcAddress(hLibInstanceDb,"CTPCC_DBLIB_new");
if
(pCTPCC_DBLIB_new == NULL)
throw new CWEBCLNT_ERR( ERR_GETPROCADDR_FAILED,
szDllName, GetLastError());
}
else if
(Reg.eDB_Protocol == ODBC)
{
strcpy( szDllName, Reg.szPath );
strcat( szDllName, "tpcc_odbc.dll");
hLibInstanceDb = LoadLibrary( szDllName );
if
(hLibInstanceDb == NULL)
throw new CWEBCLNT_ERR( ERR_LOADDLL_FAILED,
szDllName, GetLastError());
//
get function pointer to wrapper for class constructor
pCTPCC_ODBC_new = (TYPE_CTPCC_ODBC*)
GetProcAddress(hLibInstanceDb,"CTPCC_ODBC_new");
if
(pCTPCC_ODBC_new == NULL)
throw new CWEBCLNT_ERR( ERR_GETPROCADDR_FAILED,
szDllName, GetLastError());
}
}
if
(dwNumDeliveryThreads)
{
// for deferred
delivery txns:
hDoneEvent =
CreateEvent( NULL, TRUE /* manual reset */, FALSE /* initially not
signalled */, NULL );
InitializeCriticalSection(&DelBuffCriticalSection);
hWorkerSemaphore = CreateSemaphore( NULL, 0, dwDelBuffSize,
NULL );
dwDelBuffFreeCount = dwDelBuffSize;

InitJulianTime(NULL);
// create unique
log file name based on delilog-yyymmdd-hhmm.log

```

<pre> SYSTEMTIME Time; &amp;Time ); 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 ); new CTxnLog(szLogFile, TXN_LOG_WRITE); into txn log for START txnDelilog-&gt;WriteCtrlRecToLog(TXN_EVENT_START, szMyComputerName, sizeof(szMyComputerName)); // allocate structures for delivery buffers and thread mgmt = new HANDLE[dwNumDeliveryThreads]; new DELIVERY_TRANSACTION[dwDelBuffSize]; DeliveryWorkerThread to perform actual delivery txns i&lt;dwNumDeliveryThreads; i++) { pDeliHandles[i] = (HANDLE) _beginthread( DeliveryWorkerThread, 0, NULL ); if (pDeliHandles[i] == INVALID_HANDLE_VALUE) throw new CWEBCLNT_ERR( ERR_DELIVERY_THREAD_FAILED); } break; case DLL_PROCESS_DETACH: if (dwNumDeliveryThreads) { if (txnDelilog != NULL) { //write event into txn log for STOP txnDelilog-&gt;WriteCtrlRecToLog(TXN_EVENT_STOP, szMyComputerName, sizeof(szMyComputerName)); // This will do a clean shutdown of the delivery log file CTxnLog *txnDelilogLocal = txnDelilog; txnDelilog= NULL; delete txnDelilogLocal; } delete [] pDeliHandles; </pre>	<pre> pDelBuff; hWorkerSemaphore ); hDoneEvent ); DeleteCriticalSection(&amp;DelBuffCriticalSection); } DeleteCriticalSection(&amp;TermCriticalSection); if (hLibInstanceTm != NULL) FreeLibrary( hLibInstanceTm ); hLibInstanceTm = NULL; if (hLibInstanceDb != NULL) FreeLibrary( hLibInstanceDb ); hLibInstanceDb = NULL; Sleep(500); break; default: /* nothing */; } } catch (CBaseErr *e) { WriteMessageToEventLog(e-&gt;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-&gt;dwExtensionVersion = MAKELONG(HSE_VERSION_MINOR, HSE_VERSION_MAJOR); </pre>
--	--



```

        Istrcpy(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];

```

```

    int                lpbSize;
    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 !=
Term.pClientData[TermId].iSyncId)
                throw new
CWEBCLNT_ERR( ERR_INVALID_SYNC_CONNECTION);
            //set use time
            Term.pClientData[TermId].iTickCount = GetTickCount();
        }

        switch(iCmd)
        {
        case 0:
            WelcomeForm(pECB, szBuffer);
            break;
        case 1:
            switch( FormId )
            {
            case WELCOME_FORM:
            case
MAIN_MENU_FORM:
                break;
            case
NEW_ORDER_FORM:
                ProcessNewOrderForm(pECB, TermId, szBuffer);
                break;
            case PAYMENT_FORM:
                ProcessPaymentForm(pECB, TermId, szBuffer);
                break;

```

```

                                case DELIVERY_FORM:
ProcessDeliveryForm(pECB, TermId, szBuffer);
                                break;
                                case
ORDER_STATUS_FORM:
ProcessOrderStatusForm(pECB, TermId, szBuffer);
                                break;
                                case
STOCK_LEVEL_FORM:
ProcessStockLevelForm(pECB, TermId, szBuffer);
                                break;
                                }
                                break;
                                case 2:
display new-order input form
                                // new-order selected from menu;
INPUT_FORM, szBuffer);
                                MakeNewOrderForm(TermId, NULL,
                                break;
                                case 3:
display payment input form
                                // payment selected from menu;
INPUT_FORM, szBuffer);
                                MakePaymentForm(TermId, NULL,
                                break;
                                case 4:
display delivery input form
                                // delivery selected from menu;
INPUT_FORM, szBuffer);
                                MakeDeliveryForm(TermId, NULL,
                                break;
                                case 5:
display order-status input form
                                // order-status selected from menu;
NULL, INPUT_FORM, szBuffer);
                                MakeOrderStatusForm(TermId,
                                break;
                                case 6:
display stock-level input form
                                // stock-level selected from menu;
NULL, INPUT_FORM, szBuffer);
                                MakeStockLevelForm(TermId,
                                break;
                                case 7:
                                // ExitCmd
                                TermDelete(TermId);
                                WelcomeForm(pECB, szBuffer);
                                break;
                                case 8:
                                SubmitCmd(pECB, szBuffer);
                                break;
                                case 9:
                                // menu
                                MakeMainMenuForm(TermId,
Term.pClientData[TermId].iSyncId, szBuffer);
                                break;
                                case 10:
                                // CMD=Clear
                                // resets all connections; should only
                                // be used when no other connections are active
                                TermDeleteAll();
                                TermInit();
                                WelcomeForm(pECB, szBuffer);
                                break;
                                case 11:
                                // CMD=Stats
StatsCmd(pECB, szBuffer);
                                break;
                                }
                                catch (CBaseErr *e)
                                {
                                ErrorForm( pECB, e->ErrorType(),
e->ErrorNum(), TermId, iSyncId, e->ErrorText(), szBuffer );
                                delete e;
                                }
                                catch (...)
                                {
                                ErrorForm( pECB, ERR_TYPE_WEBDLL, 0,
TermId, iSyncId, "Error: Unhandled exception in Web Client.",
szBuffer );
                                }
                                #ifdef ICECAP
                                StopCAP();
                                #endif
                                lpbSize = strlen(szBuffer);
                                wsprintf(szHeader1,
                                "Content-Type: text/html\r\n"
                                "Content-Length: %d\r\n"
                                "Connection: Keep-Alive\r\n\r\n",
                                lpbSize);
                                strcat( szHeader1, szBuffer );
                                (*pECB->ServerSupportFunction)(pECB->ConnID,
HSE_REQ_SEND_RESPONSE_HEADER, szHeader, (LPDWORD)
&dwSize, (LPDWORD)szHeader1);
                                //finish up and keep connection
                                pECB->dwHttpStatusCode = 200;
                                return HSE_STATUS_SUCCESS_AND_KEEP_CONN;
                                }
void WriteMessageToEventLog(LPTSTR lpszMsg)
{
    TCHAR szMsg[256];
    HANDLE hEventSource;
    LPTSTR lpszStrings[2];
    // Use event logging to log the error.
    //
    hEventSource = RegisterEventSource(NULL,
TEXT("TPCC.DLL"));
    _stprintf(szMsg, TEXT("Error in TPCC.DLL: "));
    lpszStrings[0] = szMsg;
    lpszStrings[1] = lpszMsg;
    if (hEventSource != NULL)
    {
        ReportEvent(hEventSource, // handle of event source
EVENTLOG_ERROR_TYPE, // event type
0, // event category
0, // event ID
NULL, // current user's SID
2, // strings in lpszStrings
0, // no bytes of raw data
(LPCTSTR *)lpszStrings, // array of error strings
NULL); // no raw data
    }
    (VOID) DeregisterEventSource(hEventSource);
}

```

```

}
}

/* FUNCTION: DeliveryWorkerThread
 *
 * PURPOSE: This function processes deferred delivery txns.
 There are typically several
 * threads running this routine. The
 number of threads is determined by an entry
 * read from the registry. The thread
 waits for work by waiting on semaphore.
 * When a delivery txn is posted, the
 semaphore is released. After processing
 * the delivery txn, information is
 logged to record the txn status and execution
 * time.
 */

/*static*/ void DeliveryWorkerThread(void *ptr)
{
    CTPCC_BASE *pTxn = NULL;

    DELIVERY_TRANSACTION delivery;
    PDELIVERY_DATA
pDeliveryData;
    TXN_RECORD_TPCC_DELIV_DEF txnDeliRec;

    DWORD
index;
    HANDLE
handles[2];

    SYSTEMTIME trans_end;
//delivery transaction finished time
    SYSTEMTIME trans_start;
//delivery transaction start time
    int iRetryCnt = 0;
    static int iMaxRetries = 10;

    assert(txnDeliRec != NULL);

    assert(txnDeliRec != NULL);

Reconnect:
    try
    {
        if (Reg.eDB_Protocol == ODBC)
            pTxn = pCTPCC_ODBC_new(
Reg.szDbServer, Reg.szDbUser, Reg.szDbPassword,
szMyComputerName, Reg.szDbName );
        else if (Reg.eDB_Protocol == DBLIB)
            pTxn = pCTPCC_DBLIB_new(
Reg.szDbServer, Reg.szDbUser, Reg.szDbPassword,
szMyComputerName, Reg.szDbName );
        pDeliveryData = pTxn->BuffAddr_Delivery();
    }
    catch (CBaseErr *e)
    {
        char szTmp[1024];
        wsprintf( szTmp, "Error in Delivery Txn thread.
Could not connect to database. "
"%s. Server=%s,
User=%s, Password=%s, Database=%s",
e->ErrorText(),
Reg.szDbServer, Reg.szDbUser, Reg.szDbPassword, Reg.szDbName );
        WriteMessageToEventLog( szTmp );
        delete e;

```

```

// will retry connection up to ten times
if (iRetryCnt++ < iMaxRetries)
{
    Sleep(5000); // delay for 5
seconds
    goto Reconnect;
}

wsprintf( szTmp, "Delivery Txn thread
terminating after %d retries.", iMaxRetries );
WriteMessageToEventLog( szTmp );
goto ErrorExit;
}
catch (...)
{
    WriteMessageToEventLog(TEXT("Unhandled
exception caught in DeliveryWorkerThread. Delivery Txn thread
terminating."));
    goto ErrorExit;
}

while (TRUE)
{
    try
    {
        //while delivery thread running, i.e.
user has not requested termination
        while (TRUE)
        {
            // need to wait for multiple
objects: program exit or worker semaphore;
            handles[0] = hDoneEvent;
            handles[1] =
hWorkerSemaphore;
            index =
WaitForMultipleObjects( 2, &handles[0], FALSE, INFINITE );
            if (index ==
WAIT_OBJECT_0)
                goto ErrorExit;

ZeroMemory(&txnDeliRec, sizeof(txnDeliRec));
            txnDeliRec.TxnType =
TXN_REC_TYPE_TPCC_DELIV_DEF;

            // make a local copy of
current entry from delivery buffer and increment buffer index
            EnterCriticalSection(&DelBuffCriticalSection);
            delivery =
*(pDelBuff+dwDelBuffBusyIndex);
            dwDelBuffFreeCount++;
            dwDelBuffBusyIndex++;
            if (dwDelBuffBusyIndex
== dwDelBuffSize) // wrap-around if at end of buffer
                dwDelBuffBusyIndex = 0;

LeaveCriticalSection(&DelBuffCriticalSection);

            pDeliveryData->w_id =
delivery.w_id;
            pDeliveryData->o_carrier_id = delivery.o_carrier_id;

```

```

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

```

```

        "Process", "..NewOrder..", "..Payment..",
"..Delivery..", "..Order-Status..", "..Stock-Level..",
        "..Exit..", "Submit", "Menu", "Clear", "Stats", ""
    };

    *pCmd = 0; // default is the login
screen
    *pTermId = 0;

    // if no params (i.e., empty query string), then return login
screen
    if (strlen(pECB->lpszQueryString) == 0)
        return;

    // parse FORMID, TERMID, and SYNCID
    *pFormId = GetIntKeyValue(&ptr, "FORMID", NO_ERR,
NO_ERR);
    *pTermId = GetIntKeyValue(&ptr, "TERMID", NO_ERR,
NO_ERR);
    *pSyncId = GetIntKeyValue(&ptr, "SYNCID", NO_ERR,
NO_ERR);

    // parse CMD
    GetKeyValue(&ptr, "CMD", szBuffer, sizeof(szBuffer),
ERR_COMMAND_UNDEFINED);

    // see which command it matches
    for(i=0; ; i++)
    {
        if (szCmds[i][0] == 0)
            // no more; no match; return error
            throw new CWEBCLNT_ERR(
ERR_COMMAND_UNDEFINED);
        if ( !strcmp(szCmds[i], szBuffer) )
        {
            *pCmd = i+1;
            break;
        }
    }

/* FUNCTION: void WelcomeForm
*
*/

void WelcomeForm(EXTENSION_CONTROL_BLOCK *pECB, char
 *szBuffer)
{
    char szTmp[1024];

    //welcome to tpc-c html form buffer, this is first form client
    sees.
    strcpy( szBuffer, "<HTML><HEAD><TITLE>TPC-C
Web Client</TITLE></HEAD><BODY>"

"<B><BIG>Microsoft TPC-C Web Client (ver 4.30)</BIG></B>
<BR> <BR>"

"<font face='Courier New'><PRE>"

"Compiled: __DATE__, __TIME__ <BR>"

"Source: __FILE__ (" __TIMESTAMP__ ")<BR>"

"</PRE></font>"

"<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='1'>"
"<INPUT TYPE='hidden' NAME='TERMID' VALUE='0'>"
"<INPUT TYPE='hidden' NAME='SYNCID' VALUE='0'>"
"<INPUT TYPE='hidden' NAME='VERSION' VALUE=''"
WEBCLIENT_VERSION"'>"
    );

    sprintf( szTmp, "Configuration Settings: <BR><font
face='Courier New' color='blue'><PRE>"
        "Txn Monitor
    = <B>%s</B><BR>"
        "Database
protocol = <B>%s</B><BR>"
        "Max
Connections = <B>%d</B><BR>"
        "# of Delivery
Threads = <B>%d</B><BR>"
        "Max Pending
Deliveries = <B>%d</B><BR>"
        , szTxnMonNames[Reg.eTxnMon],
szDBNames[Reg.eDB_Protocol],
        Reg.dwMaxConnections,
dwNumDeliveryThreads, dwDelBuffSize );
    strcat( szBuffer, szTmp);

    if (Reg.eTxnMon == COM)
    {
        sprintf( szTmp, "COM Single Pool =
<B>%s</B><BR>",
            Reg.bCOM_SinglePool ? "YES" :
"NO" );
        strcat( szBuffer, szTmp);
    }
    strcat( szBuffer, "</PRE></font>");

    if (Reg.eTxnMon == None)
        // connection options may be specified when not
using a txn monitor
        sprintf( szTmp, "Please enter your
database options for this connection:<BR>"

"<font face='Courier New' color='blue'><PRE>"
        "DB
Server = <INPUT NAME='db_server' SIZE=20
VALUE='%s'><BR>"
        "DB
User ID = <INPUT NAME='db_user' SIZE=20
VALUE='%s'><BR>"
        "DB
Password = <INPUT NAME='db_passwd' SIZE=20
VALUE='%s'><BR>"
        "DB
Name = <INPUT NAME='db_name' SIZE=20
VALUE='%s'><BR>"

"</PRE></font>"
        , Reg.szDbServer,
Reg.szDbUser, Reg.szDbPassword, Reg.szDbName );
    else

```

```

// if using a txn monitor, connection options are
determined from registry; can't
// set per user. show options fyi
sprintf( szTmp, "Database options which
will be used by the transaction monitor:<BR>"

"<font face=\"Courier New\" color=\"blue\"><PRE>"
Server = <B>%s</B><BR>"
User ID = <B>%s</B><BR>"
Password = <B>%s</B><BR>"
Name = <B>%s</B><BR>"

"</PRE></font>"
, Reg.szDbServer,
Reg.szDbUser, Reg.szDbPassword, Reg.szDbName);
strcat( szBuffer, szTmp);

sprintf( szTmp, "Please enter your Warehouse and
District for this session:<BR>"

"<font
face=\"Courier New\" color=\"blue\"><PRE>";
strcat( szBuffer, szTmp);
strcat( szBuffer, "Warehouse ID = <INPUT
NAME=\"w_id\" SIZE=6><BR>"

"District ID = <INPUT NAME=\"d_id\" SIZE=2><BR>"

"</PRE></font><HR>"

"<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"Submit\">"

"</FORM></BODY></HTML>");
}

/* FUNCTION: SubmitCmd
*
* PURPOSE: This function allocated a new terminal id in the
Term structure array.
*
*/

void SubmitCmd(EXTENSION_CONTROL_BLOCK *pECB, char
*szBuffer)
{
int iNewTerm;
char *ptr = pECB->lpszQueryString;

char szVersion[32] = { 0 };
char szServer[32] = { 0 };
char szUser[32] = "sa";
char szPassword[32] = { 0 };
char szDatabase[32] = "tpcc";

// validate version field; the version field ensures that the
RTE is synchronized with the web client
GetKeyValue(&ptr, "VERSION", szVersion,
sizeof(szVersion), ERR_VERSION_MISMATCH);
if ( strcmp( szVersion, WEBCLIENT_VERSION ) )
throw new CWEBCLNT_ERR(
ERR_VERSION_MISMATCH);

if (Reg.eTxnMon == None)
{
// parse Server name
GetKeyValue(&ptr, "db_server", szServer,
sizeof(szServer), ERR_NO_SERVER_SPECIFIED);
// parse User name
GetKeyValue(&ptr, "db_user", szUser,
sizeof(szUser), NO_ERR);
// parse Password
GetKeyValue(&ptr, "db_passwd", szPassword,
sizeof(szPassword), NO_ERR);
// parse Database name
GetKeyValue(&ptr, "db_name", szDatabase,
sizeof(szDatabase), NO_ERR);
}

// parse warehouse ID
int w_id = GetIntKeyValue(&ptr, "w_id",
ERR_HTML_ILL_FORMED, ERR_W_ID_INVALID);
if ( w_id < 1 )
throw new CWEBCLNT_ERR(
ERR_W_ID_INVALID);

// parse district ID
int d_id = GetIntKeyValue(&ptr, "d_id",
ERR_HTML_ILL_FORMED, ERR_D_ID_INVALID);
if ( d_id < 1 || d_id > 10 )
throw new CWEBCLNT_ERR(
ERR_D_ID_INVALID);

iNewTerm = TermAdd();

Term.pClientData[iNewTerm].w_id = w_id;
Term.pClientData[iNewTerm].d_id = d_id;

try
{
if (Reg.eTxnMon == TUXEDO)
Term.pClientData[iNewTerm].pTxn
= pCTPCC_TUXEDO_new();
else if (Reg.eTxnMon == ENCINA)
Term.pClientData[iNewTerm].pTxn
= pCTPCC_ENCINA_new();
else if (Reg.eTxnMon == COM)
Term.pClientData[iNewTerm].pTxn
= pCTPCC_COM_new( Reg.bCOM_SinglePool );
else if (Reg.eDB_Protocol == ODBC)
Term.pClientData[iNewTerm].pTxn
= pCTPCC_ODBC_new( szServer, szUser, szPassword,
szMyComputerName, szDatabase );
else if (Reg.eDB_Protocol == DBLIB)
Term.pClientData[iNewTerm].pTxn
= pCTPCC_DBLIB_new( szServer, szUser, szPassword,
szMyComputerName, szDatabase );
}
catch (...)
{
TermDelete(iNewTerm);
throw; // pass exception upward
}

MakeMainMenuForm(iNewTerm,
Term.pClientData[iNewTerm].iSyncId, szBuffer);
}

/* FUNCTION: StatsCmd
*
* PURPOSE: This function returns to the browser the total
number of active terminal ids.

```

```

*           This routine is for
development/debugging purposes.
*/

void StatsCmd(EXTENSION_CONTROL_BLOCK *pECB, char
*szBuffer)
{
    int i;
    int    iTotals;

    EnterCriticalSection(&TermCriticalSection);

    iTotals = 0;
    for(i=0; i<Term.iNumEntries; i++)
    {
        if (Term.pClientData[i].iNextFree == -1)
            iTotals++;
    }

    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>"
    , iTotals);
}

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_QTY_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."
    },
{
0,
""
    },
};

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)
    sprintf( 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:   char                *pQueryString
http string from client browser
*
*              char
* pKey         key value to look for
*              char
* pValue       character array into which to place
key's value
*
*              int
iMax          maximum length of key
value array.
*
*              WEBERROR
err           error value to throw
*
* 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\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:   char                *pQueryString
http string from client browser
*
*              char
* pKey         key value to look for
*              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++;

ptr0 = ptr;          // remember starting point
// 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:   none
*
* 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 = 0;
    if ( Term.pClientData )
        free(Term.pClientData);
    Term.pClientData = NULL;

    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;
    int 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
               id                Terminal id of client exiting
*
*/

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);
    }
}

/* FUNCTION: MakeErrorForm
*/

void ErrorForm(EXTENSION_CONTROL_BLOCK *pECB, int
iType, int iErrorNum, int iTermId, int iSyncId, char *szErrorText, char
*szBuffer )
{

```

```

        wsprintf(szBuffer,
                "<HTML><HEAD><TITLE>TPC-C
Error</TITLE></HEAD><BODY>"
                "<FORM ACTION='tpcc.dll'"
METHOD="GET">"
                "<INPUT TYPE='hidden'"
NAME="STATUSID" VALUE="%d">"
                "<INPUT TYPE='hidden'" NAME="ERROR"
VALUE="%d">"
                "<INPUT TYPE='hidden'"
NAME="FORMID" VALUE="%d">"
                "<INPUT TYPE='hidden'"
NAME="TERMINID" VALUE="%d">"
                "<INPUT TYPE='hidden'"
NAME="SYNCID" VALUE="%d">"
                "<BOLD>An Error
Occurred</BOLD><BR><BR>"
                "%s"
                "<BR><BR><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></BODY></HTML>"
                , iType, iErrorNum, MAIN_MENU_FORM,
iTermId, iSyncId, szErrorText );
}

/* FUNCTION: MakeMainMenuForm
*/

void MakeMainMenuForm(int iTermId, int iSyncId, char *szForm)
{
    wsprintf(szForm,
            "<HTML><HEAD><TITLE>TPC-CMain
Menu</TITLE></HEAD><BODY>"
            "Select Desired Transaction.<BR><HR>"
            "<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="TERMINID" VALUE="%d">"
            "<INPUT TYPE='hidden'"
NAME="SYNCID" VALUE="%d">"
            "<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></BODY></HTML>"
        , MAIN_MENU_FORM, iTermId, iSyncId);
}

/* FUNCTION: MakeStockLevelForm
*
* PURPOSE:      This function constructs the Stock Level HTML
page.
*
* 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 MakeStockLevelForm(int iTermId, STOCK_LEVEL_DATA
*pStockLevelData, BOOL bInput, char *szForm)
{
        int            c;

        c = wsprintf(szForm,
                "<HTML><HEAD><TITLE>TPC-CStock
Level</TITLE></HEAD><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=\"TERMINID\" VALUE=\"%d\">"
                "<INPUT TYPE=\"hidden\"
NAME=\"SYNCID\" VALUE=\"%d\">"
                "<PRE><font face=\"Courier\">
        Stock-Level<BR>"
                "Warehouse: %6.6d District: %2.2d<BR>"
<BR>",
                STOCK_LEVEL_FORM, iTermId,
Term.pClientData[iTermId].iSyncId,
        Term.pClientData[iTermId].w_id,
Term.pClientData[iTermId].d_id);

        if ( bInput )
        {
                strcpy(szForm+c,
                        "Stock Level Threshold: <INPUT
NAME=\"TT*\" SIZE=2><BR><BR>"
                        "low stock: </font><BR> <BR>
<BR> <BR> <BR> <BR> <BR> <BR> <BR> <BR>"
                        " <BR> <BR> <BR> <BR> <BR>
<BR> <BR></PRE><HR>"
                        "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"Process\">"
                        "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"Menu\">"
                        "</FORM></HTML>");
        }
        else
        {
                wsprintf(szForm+c,
                        "Stock Level Threshold: %2.2d<BR>
<BR>"
                        "low stock: %3.3d</font> <BR>
<BR> <BR> <BR> <BR> <BR> <BR> <BR> <BR> <BR>"

```

```

                "<BR> <BR> <BR> <BR> <BR>
<BR> <BR> <BR></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>"
                , pStockLevelData->threshold,
pStockLevelData->low_stock);
        }
}

/* FUNCTION: MakeNewOrderForm
*
* 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 MakeNewOrderForm(int iTermId, NEW_ORDER_DATA
*pNewOrderData, BOOL bInput, char *szForm)
{
        int            i, c;
        BOOL            bValid;
        static char szBR[] = " <BR> <BR> <BR> <BR>
<BR> <BR> <BR> <BR> <BR> <BR> <BR> <BR> <BR>
<BR>";

        if (!bInput)
                assert( pNewOrderData->exec_status_code ==
eOK || pNewOrderData->exec_status_code == eInvalidItem);

        bValid = (bInput || (pNewOrderData->exec_status_code ==
eOK));

        c = wsprintf(szForm,
                "<HTML><HEAD><TITLE>TPC-CNew
Order</TITLE></HEAD><BODY>"
                "<FORM ACTION=\"tpcc.dll\"
METHOD=\"GET\">"
                "<INPUT TYPE=\"hidden\"
NAME=\"STATUSID\" VALUE=\"%d\">"
                "<INPUT TYPE=\"hidden\" NAME=\"ERROR\"
VALUE=\"0\">"
                "<INPUT TYPE=\"hidden\"
NAME=\"FORMID\" VALUE=\"%d\">"
                "<INPUT TYPE=\"hidden\"
NAME=\"TERMINID\" VALUE=\"%d\">"
                "<INPUT TYPE=\"hidden\"
NAME=\"SYNCID\" VALUE=\"%d\">"
                "<PRE><font face=\"Courier\">
                New Order<BR>"
                , bValid ? 0 : ERR_BAD_ITEM_ID,
NEW_ORDER_FORM, iTermId,
Term.pClientData[iTermId].iSyncId);

        if ( bInput )
        {

```

```

        c += sprintf(szForm+c, "Warehouse: %6.6d ",
Term.pClientData[iTermId].w_id);

        strcpy( szForm+c,
                "District: <INPUT NAME=\"DID*\"
SIZE=1>
                Date:<BR>"
                "Customer: <INPUT
NAME=\"CID*\" SIZE=4> Name:
                Credit:
                %Disc:<BR>"
                "Order Number:
                Number of
Lines:
                W_tax:
                D_tax:<BR> <BR>"
                " Supp_W Item_Id Item Name
                Qty Stock B/G Price
                Amount<BR>"
                " <INPUT NAME=\"SP00*\"
SIZE=4> <INPUT NAME=\"IID00*\" SIZE=6>
<INPUT NAME=\"Qty00*\" SIZE=1><BR>"
                " <INPUT NAME=\"SP01*\"
SIZE=4> <INPUT NAME=\"IID01*\" SIZE=6>
<INPUT NAME=\"Qty01*\" SIZE=1><BR>"
                " <INPUT NAME=\"SP02*\"
SIZE=4> <INPUT NAME=\"IID02*\" SIZE=6>
<INPUT NAME=\"Qty02*\" SIZE=1><BR>"
                " <INPUT NAME=\"SP03*\"
SIZE=4> <INPUT NAME=\"IID03*\" SIZE=6>
<INPUT NAME=\"Qty03*\" SIZE=1><BR>"
                " <INPUT NAME=\"SP04*\"
SIZE=4> <INPUT NAME=\"IID04*\" SIZE=6>
<INPUT NAME=\"Qty04*\" SIZE=1><BR>"
                " <INPUT NAME=\"SP05*\"
SIZE=4> <INPUT NAME=\"IID05*\" SIZE=6>
<INPUT NAME=\"Qty05*\" SIZE=1><BR>"
                " <INPUT NAME=\"SP06*\"
SIZE=4> <INPUT NAME=\"IID06*\" SIZE=6>
<INPUT NAME=\"Qty06*\" SIZE=1><BR>"
                " <INPUT NAME=\"SP07*\"
SIZE=4> <INPUT NAME=\"IID07*\" SIZE=6>
<INPUT NAME=\"Qty07*\" SIZE=1><BR>"
                " <INPUT NAME=\"SP08*\"
SIZE=4> <INPUT NAME=\"IID08*\" SIZE=6>
<INPUT NAME=\"Qty08*\" SIZE=1><BR>"
                " <INPUT NAME=\"SP09*\"
SIZE=4> <INPUT NAME=\"IID09*\" SIZE=6>
<INPUT NAME=\"Qty09*\" SIZE=1><BR>"
                " <INPUT NAME=\"SP10*\"
SIZE=4> <INPUT NAME=\"IID10*\" SIZE=6>
<INPUT NAME=\"Qty10*\" SIZE=1><BR>"
                " <INPUT NAME=\"SP11*\"
SIZE=4> <INPUT NAME=\"IID11*\" SIZE=6>
<INPUT NAME=\"Qty11*\" SIZE=1><BR>"
                " <INPUT NAME=\"SP12*\"
SIZE=4> <INPUT NAME=\"IID12*\" SIZE=6>
<INPUT NAME=\"Qty12*\" SIZE=1><BR>"
                " <INPUT NAME=\"SP13*\"
SIZE=4> <INPUT NAME=\"IID13*\" SIZE=6>
<INPUT NAME=\"Qty13*\" SIZE=1><BR>"
                " <INPUT NAME=\"SP14*\"
SIZE=4> <INPUT NAME=\"IID14*\" SIZE=6>
<INPUT NAME=\"Qty14*\" SIZE=1><BR>"
                "Execution Status:
                Total:<BR>"
                "</font></PRE><HR>"
                "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"Process\">"
                "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"Menu\">"
                "</FORM></HTML>"
                );
    }

```

```

    else
    {
        c += sprintf(szForm+c, "Warehouse: %6.6d
District: %2.2d
                Date: ",
                pNewOrderData->w_id,
                pNewOrderData->d_id);

        if ( bValid )
        {
            c += sprintf(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 += sprintf(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 <BR> <BR>"
                "
                Supp_W Item_Id Item Name
                Qty Stock B/G Price
                Amount<BR>",
                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, "
%6.6d %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 += sprintf(szForm+c,
            "% Disc:<BR>"
            "Order Number: %8.8d
Number of Lines:      W_tax:      D_tax:<BR> <BR>"
            " Supp_W Item_Id Item
Name          Qty Stock B/G Price  Amount<BR>"
            , 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 += sprintf(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 = sprintf(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=\"TERMINID\" VALUE=\"%d\">"
        "<INPUT TYPE=\"hidden\"
NAME=\"SYNCID\" VALUE=\"%d\">"
        "<PRE><font face=\"Courier\">
Payment<BR>"
        "Date: "
        , PAYMENT_FORM, iTermId,
Term.pClientData[iTermId].iSyncId);

    if ( !bInput )
    {
        c += sprintf(szForm+c, "% 2.2d-% 2.2d-% 4.4d
% 2.2d:% 2.2d:% 2.2d",

        pPaymentData->h_date.day,
        pPaymentData->h_date.month,
        pPaymentData->h_date.year,
        pPaymentData->h_date.hour,
        pPaymentData->h_date.minute,
        pPaymentData->h_date.second);
    }

    if ( bInput )
    {
        c += sprintf(szForm+c,
            "<BR> <BR>Warehouse: % 6.6d"
            " District: <INPUT
NAME=\"DID*\" SIZE=1><BR> <BR> <BR> <BR> <BR>"
            "Customer: <INPUT
NAME=\"CID*\" SIZE=4>"
            "Cust-Warehouse: <INPUT
NAME=\"CWI*\" SIZE=6> "
            "Cust-District: <INPUT
NAME=\"CDI*\" SIZE=1><BR>"
            "Name:      <INPUT
NAME=\"CLT*\" SIZE=16>      Since:<BR>"
            "
            Credit:<BR>"
            "
            Disc:<BR>"
            "
            Phone:<BR> <BR>"
            "Amount Paid:      $<INPUT
NAME=\"HAM*\" SIZE=7>      New Cust-Balance:<BR>"
            "Credit Limit:<BR> <BR>Cust-Data:
<BR> <BR> <BR> <BR> <BR></font></PRE><HR>"
            "<INPUT TYPE=\"submit\""
NAME=\"CMD\" VALUE=\"Process\"><INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"Menu\">"
            "</BODY></FORM></HTML>"
            , Term.pClientData[iTermId].w_id);
    }
    else
    {
        c += sprintf(szForm+c,
            "<BR> <BR>Warehouse: % 6.6d
District: % 2.2d<BR>"
            "%-20s      %-20s<BR>"
            "%-20s      %-20s<BR>"

```

```

"%-20s %-2s %5.5s-%4.4s  %-20s" strcpy(szForm+c, "Cust-Data: <BR>
%-2s %5.5s-%4.4s<BR> <BR>")
"Customer: %4.4d Cust-Warehouse: <BR> <BR> <BR>");
%6.6d Cust-District: %2.2d<BR>" strcat(szForm, "
"Name: %-16s %-2s %-16s <BR></font></PRE><HR>"
Since: %2.2d-%2.2d-%4.4d<BR>" " %-20s Credit: "<INPUT TYPE="submit" NAME="CMD"
%-2s<BR>" " " " VALUE="..NewOrder..">"
, Term.pClientData[iTermId].w_id, "<INPUT TYPE="submit" NAME="CMD"
pPaymentData->d_id " " " VALUE="..Payment..">"
, pPaymentData->w_street_1, "<INPUT TYPE="submit" NAME="CMD"
pPaymentData->d_street_1 " " " VALUE="..Delivery..">"
, pPaymentData->w_street_2, "<INPUT TYPE="submit" NAME="CMD"
pPaymentData->d_street_2 " " " VALUE="..Order-Status..">"
, pPaymentData->w_city, "<INPUT TYPE="submit" NAME="CMD"
pPaymentData->w_state, pPaymentData->w_zip, " " " VALUE="..Stock-Level..">"
pPaymentData->w_zip+5 " " " VALUE="..Exit..">"
, pPaymentData->d_city, "<INPUT TYPE="submit" NAME="CMD"
pPaymentData->d_state, pPaymentData->d_zip, " " " VALUE="..Exit..">"
pPaymentData->d_zip+5 " " " VALUE="..Exit..">"
, pPaymentData->c_id, "<INPUT TYPE="submit" NAME="CMD" VALUE="..Exit..">"
pPaymentData->c_w_id, pPaymentData->c_d_id, " " " VALUE="..Exit..">"
, pPaymentData->c_first, "</BODY></FORM></HTML>");
pPaymentData->c_middle, pPaymentData->c_last " " " "
, pPaymentData->c_since.day, " " " "
pPaymentData->c_since.month, pPaymentData->c_since.year, " " " "
, pPaymentData->c_street_1, " " " "
pPaymentData->c_credit " " " "
);
/* FUNCTION: MakeOrderStatusForm
*
* 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 MakeOrderStatusForm(int iTermId, ORDER_STATUS_DATA
*pOrderStatusData, BOOL bInput, char *szForm)
{
int i, c;
static char szBR[] = " <BR> <BR> <BR> <BR> <BR>
<BR> <BR> <BR> <BR> <BR> <BR> <BR> <BR> <BR> <BR>";
c = sprintf(szForm,
"<HTML><HEAD><TITLE>TPC-C
Order-Status</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="SYCID" VALUE="%"d">"
"<PRE><font face="Courier">
Order-Status<BR>"
"Warehouse: %6.6d ",
ORDER_STATUS_FORM, iTermId,
Term.pClientData[iTermId].iSyncId,
Term.pClientData[iTermId].w_id);
if ( bInput )
{
strcpy(szForm+c,
c += sprintf(szForm+c, " %-20s %%Disc:
%5.2f<BR>", " %-20s %%Disc:
100.0*pPaymentData->c_discount);
c += sprintf(szForm+c,
" %-20s %-2s %5.5s-%4.4s
Phone: %6.6s-%3.3s-%3.3s-%4.4s<BR> <BR>",
pPaymentData->c_city,
pPaymentData->c_state, pPaymentData->c_zip,
pPaymentData->c_zip+5,
pPaymentData->c_phone,
pPaymentData->c_phone+6, pPaymentData->c_phone+9,
pPaymentData->c_phone+12 );
c += sprintf(szForm+c,
"Amount Paid: $%7.2f New
Cust-Balance: $%14.2f<BR>"
"Credit Limit: $%13.2f<BR>
<BR>"
, pPaymentData->h_amount,
pPaymentData->c_balance
, pPaymentData->c_credit_lim
);
if ( pPaymentData->c_credit[0] == 'B' &&
pPaymentData->c_credit[1] == 'C' )
c += sprintf(szForm+c,
"Cust-Data:
%-50.50s<BR> %-50.50s<BR> %-50.50s<BR>
%-50.50s<BR>",
pPaymentData->c_data, pPaymentData->c_data+50,
pPaymentData->c_data+100, pPaymentData->c_data+150 );
else
}
}

```

```

        "District: <INPUT NAME=\"DID*\"
SIZE=1><BR>"
        "Customer: <INPUT
NAME=\"CID*\" SIZE=4> Name:
NAME=\"CLT*\" SIZE=23><BR>"
        "Cust-Balance:<BR> <BR>"
        "Order-Number:      Entry-Date:
Carrier-Number:<BR>"
        "Supply-W Item-Id Qty
Amount Delivery-Date<BR> <BR> <BR> <BR> <BR>"
        " <BR> <BR> <BR> <BR> <BR>
<BR> <BR> <BR> <BR> <BR> <BR></font></PRE>"
        "<HR><INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"Process\"><INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"Menu\">"
        "</BODY></FORM></HTML>");
    }
    else
    {
        c += sprintf(szForm+c,
        "District: %2.2d<BR>"
        "Customer: %4.4d Name: %-16s
%-2s %-16s<BR>",
        pOrderStatusData->d_id,
        pOrderStatusData->c_id,
        pOrderStatusData->c_first,
        pOrderStatusData->c_middle, pOrderStatusData->c_last);

        c += sprintf(szForm+c, "Cust-Balance:
%$9.2f<BR> <BR>",
        pOrderStatusData->c_balance);

        c += sprintf(szForm+c,
        "Order-Number: %8.8d Entry-Date:
%2.2d-%2.2d-%4.4d %2.2d:%2.2d:%2.2d Carrier-Number:
%2.2d<BR>"
        "Supply-W Item-Id Qty
Amount Delivery-Date<BR>",
        pOrderStatusData->o_id,
        pOrderStatusData->o_entry_d.day,
        pOrderStatusData->o_entry_d.month,
        pOrderStatusData->o_entry_d.year,
        pOrderStatusData->o_entry_d.hour,

        pOrderStatusData->o_entry_d.minute,

        pOrderStatusData->o_entry_d.second,
        pOrderStatusData->o_carrier_id);

        for(i=0; i< pOrderStatusData->o_ol_cnt; i++)
        {
            c += sprintf(szForm+c, " %6.6d
%6.6d %2.2d %$8.2f %2.2d-%2.2d-%4.4d<BR>",
            pOrderStatusData->OL[i].ol_supply_w_id,
            pOrderStatusData->OL[i].ol_i_id,
            pOrderStatusData->OL[i].ol_quantity,
            pOrderStatusData->OL[i].ol_amount,
            pOrderStatusData->OL[i].ol_delivery_d.day,
            pOrderStatusData->OL[i].ol_delivery_d.month,
            pOrderStatusData->OL[i].ol_delivery_d.year);
        }
    }
}

```

```

        strcpy( szForm+c, szBR, (15-i)*5);
        c += (15-i)*5;

        strcpy(szForm+c,
        "</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..\">"
        "</BODY></FORM></HTML>");
    }
}

/* FUNCTION: MakeDeliveryForm
*
* 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 MakeDeliveryForm(int iTermId, DELIVERY_DATA
*pDeliveryData, BOOL bInput, char *szForm)
{
    int c;

    c = sprintf(szForm,
    "<HTML><HEAD><TITLE>TPC-C
Delivery</TITLE></HEAD><BODY>"
    "<FORM ACTION=\"tpcc.dll\"
METHOD=\"GET\">"
    "<INPUT TYPE=\"hidden\"
NAME=\"STATUSID\" VALUE=\"%d\">"
    "<INPUT TYPE=\"hidden\" NAME=\"ERROR\"
VALUE=\"0\">"
    "<INPUT TYPE=\"hidden\"
NAME=\"FORMID\" VALUE=\"%d\">"
    "<INPUT TYPE=\"hidden\"
NAME=\"TERMINID\" VALUE=\"%d\">"
    "<INPUT TYPE=\"hidden\"
NAME=\"SYNCID\" VALUE=\"%d\">"
    "<PRE><font face=\"Courier\">
Delivery<BR>"
    "Warehouse: %6.6d<BR> <BR>",
    (!bInput && (pDeliveryData->exec_status_code
!= eOK)) ? ERR_TYPE_DELIVERY_POST : 0,
    DELIVERY_FORM, iTermId,
    Term.pClientData[iTermId].iSyncId,
    Term.pClientData[iTermId].w_id);

    if ( bInput )
    {
        strcpy( szForm+c,
        "Carrier Number: <INPUT
NAME=\"OCD*\" SIZE=1><BR> <BR>"
        "Execution Status: <BR> <BR>
<BR> <BR> <BR> <BR> <BR> <BR>
" <BR> <BR> <BR> <BR> <BR> <BR>
<BR> <BR> <BR> </font></PRE><HR>"

```



```

        "<INPUT TYPE=\\"submit\\"
NAME=\\"CMD\\" VALUE=\\"Process\\">"
        "<INPUT TYPE=\\"submit\\"
NAME=\\"CMD\\" VALUE=\\"Menu\\">"
        "</BODY></FORM></HTML>");
    }
    else
    {
        wsprintf( szForm+c,
            "Carrier Number: %2.2d<BR>
<BR>"
            "Execution Status: %s <BR> <BR>
<BR> <BR> <BR> <BR> <BR> <BR>"
            "<BR> <BR> <BR> <BR> <BR>
<BR> <BR> <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..\\>"
            "</BODY></FORM></HTML>"
            , pDeliveryData->o_carrier_id,
            (pDeliveryData->exec_status_code
            == eOK) ? "Delivery has been queued." : "Delivery Post Failed "
            );
    }
}

/* FUNCTION: ProcessNewOrderForm
*
* PURPOSE: This function gets and validates the input data
from the new order form
* filling in the required input variables.
it then calls the SQLNewOrder
* transaction, constructs the output
form and writes it back to client
* browser.
*/

void ProcessNewOrderForm(EXTENSION_CONTROL_BLOCK
*pECB, int iTermId, char *szBuffer)
{
    PNEW_ORDER_DATA pNewOrder;
    BOOL fLocalFlag = FALSE;

    pNewOrder =
Term.pClientData[iTermId].pTxn->BuffAddr_NewOrder();

    ZeroMemory(pNewOrder, sizeof(NEW_ORDER_DATA));
    pNewOrder->w_id = Term.pClientData[iTermId].w_id;
    GetNewOrderData(pECB->lpszQueryString, pNewOrder,
&fLocalFlag);

    Term.pClientData[iTermId].pTxn->NewOrder(fLocalFlag);

    pNewOrder =
Term.pClientData[iTermId].pTxn->BuffAddr_NewOrder();
    MakeNewOrderForm(iTermId, pNewOrder,
OUTPUT_FORM, szBuffer );
#ifdef COUNTSPLIT

```

```

        if (fLocalFlag)
            InterlockedIncrement(&gdwLocalNOCnt);
        else
            InterlockedIncrement(&gdwRemoteNOCnt);
    #endif
}

/* FUNCTION: void ProcessPaymentForm
*
* PURPOSE: This function gets and validates the input data
from the payment form
* filling in the required input variables.
It then calls the SQLPayment
* transaction, constructs the output
form and writes it back to client
* browser.
*
* ARGUMENTS: EXTENSION_CONTROL_BLOCK
*pECB passed in structure pointer from inetsrv.
* int iTermId client browser terminal id
*/

void ProcessPaymentForm(EXTENSION_CONTROL_BLOCK
*pECB, int iTermId, char *szBuffer)
{
    PPAYMENT_DATA pPayment;
    BOOL fLocalFlag = FALSE;

    pPayment =
Term.pClientData[iTermId].pTxn->BuffAddr_Payment();
    ZeroMemory(pPayment, sizeof(PAYMENT_DATA));
    pPayment->w_id = Term.pClientData[iTermId].w_id;
    GetPaymentData(pECB->lpszQueryString, pPayment,
&fLocalFlag);

    Term.pClientData[iTermId].pTxn->Payment(fLocalFlag);

    pPayment =
Term.pClientData[iTermId].pTxn->BuffAddr_Payment();
    MakePaymentForm(iTermId, pPayment, OUTPUT_FORM,
szBuffer);
#ifdef COUNTSPLIT
    if (fLocalFlag)
        InterlockedIncrement(&gdwLocalPayCnt);
    else
        InterlockedIncrement(&gdwRemotePayCnt);
#endif
}

/* FUNCTION: ProcessOrderStatusForm
*
* PURPOSE: This function gets and validates the input data
from the Order Status
* form filling in the required input
variables. It then calls the
* SQLOrderStatus transaction,
constructs the output form and writes it
* back to client browser.
*
* ARGUMENTS: EXTENSION_CONTROL_BLOCK
*pECB passed in structure pointer from inetsrv.
* int iTermId client browser terminal id
*/

```

```

void ProcessOrderStatusForm(EXTENSION_CONTROL_BLOCK
*pECB, int iTermId, char *szBuffer)
{
    PORDER_STATUS_DATA    pOrderStatus;

    pOrderStatus =
Term.pClientData[iTermId].pTxn->BuffAddr_OrderStatus();
    ZeroMemory(pOrderStatus,
sizeof(ORDER_STATUS_DATA));
    pOrderStatus->w_id = Term.pClientData[iTermId].w_id;
    GetOrderStatusData(pECB->lpszQueryString,
pOrderStatus);

    Term.pClientData[iTermId].pTxn->OrderStatus();

    pOrderStatus =
Term.pClientData[iTermId].pTxn->BuffAddr_OrderStatus();
    MakeOrderStatusForm(iTermId, pOrderStatus,
OUTPUT_FORM, szBuffer);
}

/* FUNCTION: ProcessDeliveryForm
*
* PURPOSE:      This function gets and validates the input data
from the delivery form
*
*              filling in the required input variables.
It then calls the PostDeliveryInfo
*
*              Api, The client is then informed that
the transaction has been posted.
*
* ARGUMENTS:   EXTENSION_CONTROL_BLOCK
*pECB passed in structure pointer from inetsrv.
*
*              int
*
*              iTermId client browser terminal id
*/

void ProcessDeliveryForm(EXTENSION_CONTROL_BLOCK
*pECB, int iTermId, char *szBuffer)
{
    char    *ptr = pECB->lpszQueryString;

    PDELIVERY_DATA pDelivery;

    pDelivery =
Term.pClientData[iTermId].pTxn->BuffAddr_Delivery();
    ZeroMemory(pDelivery, sizeof(DELIVERY_DATA));
    pDelivery->w_id = Term.pClientData[iTermId].w_id;

    pDelivery->o_carrier_id = GetIntKeyValue(&ptr,
"OCD*", ERR_DELIVERY_MISSING_OCD_KEY,
ERR_DELIVERY_CARRIER_INVALID);
    if ( pDelivery->o_carrier_id > 10 || pDelivery->o_carrier_id
< 1 )
        throw new CWEBCLNT_ERR(
ERR_DELIVERY_CARRIER_ID_RANGE);

    if (dwNumDeliveryThreads)
    {
        //post delivery info
        if ( PostDeliveryInfo(pDelivery->w_id,
pDelivery->o_carrier_id) )
            pDelivery->exec_status_code =
eDeliveryFailed;
        else
            pDelivery->exec_status_code = eOK;
    }
}

```

```

else // delivery is done synchronously if no delivery threads
configured
    Term.pClientData[iTermId].pTxn->Delivery();

    pDelivery =
Term.pClientData[iTermId].pTxn->BuffAddr_Delivery();
    MakeDeliveryForm(iTermId, pDelivery, OUTPUT_FORM,
szBuffer);
}

/* FUNCTION: ProcessStockLevelForm
*
* PURPOSE:      This function gets and validates the input data
from the Stock Level
*
*              form filling in the required input
variables. It then calls the
*
*              SQLStockLevel transaction,
constructs the output form and writes it
*
*              back to client browser.
*
* ARGUMENTS:   EXTENSION_CONTROL_BLOCK
*pECB passed in structure pointer from inetsrv.
*
*              int
*
*              iTermId client browser terminal id
*/

void ProcessStockLevelForm(EXTENSION_CONTROL_BLOCK
*pECB, int iTermId, char *szBuffer)
{
    char    *ptr = pECB->lpszQueryString;

    PSTOCK_LEVEL_DATA    pStockLevel;

    pStockLevel =
Term.pClientData[iTermId].pTxn->BuffAddr_StockLevel();
    ZeroMemory( pStockLevel,
sizeof(STOCK_LEVEL_DATA));

    pStockLevel->w_id = Term.pClientData[iTermId].w_id;
    pStockLevel->d_id = Term.pClientData[iTermId].d_id;

    pStockLevel->threshold = GetIntKeyValue(&ptr, "TT*",
ERR_STOCKLEVEL_MISSING_THRESHOLD_KEY,
ERR_STOCKLEVEL_THRESHOLD_INVALID);
    if ( pStockLevel->threshold >= 100 ||
pStockLevel->threshold < 0 )
        throw new CWEBCLNT_ERR(
ERR_STOCKLEVEL_THRESHOLD_RANGE);

    Term.pClientData[iTermId].pTxn->StockLevel();

    pStockLevel =
Term.pClientData[iTermId].pTxn->BuffAddr_StockLevel();
    MakeStockLevelForm(iTermId, pStockLevel,
OUTPUT_FORM, szBuffer);
}

/* FUNCTION: GetNewOrderData
*
* PURPOSE:      This function extracts and validates the new
order form data from an http command string.
*
* ARGUMENTS:   LPSTR
lpszQueryString client browser http command string
*
*              NEW_ORDER_DATA
*pNewOrderData pointer to new order data structure
*

```

```

*/
void GetNewOrderData (LPSTR lpszQueryString,
NEW_ORDER_DATA *pNewOrderData, BOOL *pfLocalFlag)
{
    char    szTmp[26];
    int     items;
    short   ol_i_id, ol_quantity;
    char    *ptr = lpszQueryString;
    BOOL    fRemoteFlag = FALSE;

    static char szSP[MAX_OL_NEW_ORDER_ITEMS][6] =
        { "SP00*", "SP01*", "SP02*", "SP03*",
"SP04*",
        "SP05*", "SP06*", "SP07*", "SP08*",
"SP09*",
        "SP10*", "SP11*", "SP12*", "SP13*", "SP14*"
};
    static char szIID[MAX_OL_NEW_ORDER_ITEMS][7] =
        { "IID00*", "IID01*", "IID02*", "IID03*",
"IID04*",
        "IID05*", "IID06*", "IID07*", "IID08*",
"IID09*",
        "IID10*", "IID11*", "IID12*", "IID13*",
"IID14*" };
    static char szQty[MAX_OL_NEW_ORDER_ITEMS][7] =
        { "Qty00*", "Qty01*", "Qty02*", "Qty03*",
"Qty04*",
        "Qty05*", "Qty06*", "Qty07*", "Qty08*",
"Qty09*",
        "Qty10*", "Qty11*", "Qty12*", "Qty13*",
"Qty14*" };

    pNewOrderData->d_id = GetIntKeyValue(&ptr, "DID*",
ERR_NEWORDER_FORM_MISSING_DID,
ERR_NEWORDER_DISTRICT_INVALID);
    pNewOrderData->c_id = GetIntKeyValue(&ptr, "CID*",
ERR_NEWORDER_CUSTOMER_KEY,
ERR_NEWORDER_CUSTOMER_INVALID);

    for(i=0, items=0; i<MAX_OL_NEW_ORDER_ITEMS;
i++)
    {
        GetKeyValue(&ptr, szSP[i], szTmp,
sizeof(szTmp), ERR_NEWORDER_MISSING_SUPPW_KEY);
        if ( szTmp[0] )
        {
            if ( !IsNumeric(szTmp) )
                throw new
CWEBCLNT_ERR( ERR_NEWORDER_SUPPW_INVALID);
            pNewOrderData->OL[items].ol_supply_w_id = (long)atoi(szTmp);
            if ( !fRemoteFlag )
            {
                if
(!pbLocalWhsTbl[pNewOrderData->OL[items].ol_supply_w_id])
                    fRemoteFlag=TRUE;
            }

            ol_i_id =
pNewOrderData->OL[items].ol_i_id =
                GetIntKeyValue(&ptr,
szIID[i], ERR_NEWORDER_MISSING_IID_KEY,
ERR_NEWORDER_ITEMID_INVALID);
            if ( ol_i_id > 999999 || ol_i_id < 1 )

```

```

                throw new
CWEBCLNT_ERR( ERR_NEWORDER_ITEMID_RANGE);

            ol_quantity =
pNewOrderData->OL[items].ol_quantity =
                GetIntKeyValue(&ptr,
szQty[i], ERR_NEWORDER_MISSING_QTY_KEY,
ERR_NEWORDER_QTY_INVALID);
            if ( ol_quantity > 99 || ol_quantity < 1
)
                throw new
CWEBCLNT_ERR( ERR_NEWORDER_QTY_RANGE );

            items++;
        }
        else
        {
            // nothing entered for supply
            warehouse, so item id and qty must also be blank
            GetKeyValue(&ptr, szIID[i], szTmp,
sizeof(szTmp), ERR_NEWORDER_MISSING_IID_KEY);
            if ( szTmp[0] )
                throw new
CWEBCLNT_ERR(
ERR_NEWORDER_ITEMID_WITHOUT_SUPPW);

            GetKeyValue(&ptr, szQty[i], szTmp,
sizeof(szTmp), ERR_NEWORDER_MISSING_QTY_KEY);
            if ( szTmp[0] )
                throw new
CWEBCLNT_ERR( ERR_NEWORDER_QTY_WITHOUT_SUPPW
);
        }
    }
    if ( items == 0 )
        throw new CWEBCLNT_ERR(
ERR_NEWORDER_NOITEMS_ENTERED );

    pNewOrderData->o_ol_cnt = items;
    if ( pfLocalFlag )
        *pfLocalFlag = !fRemoteFlag;
}

/* FUNCTION: GetPaymentData
*
* PURPOSE:      This function extracts and validates the payment
form data from an http command string.
*
* ARGUMENTS:   LPSTR
lpszQueryString      client browser http command string
                    PAYMENT_DATA
*pPaymentData        pointer to payment data structure
*/

void GetPaymentData(LPSTR lpszQueryString, PAYMENT_DATA
*pPaymentData, BOOL *pfLocalFlag)
{
    char    szTmp[26];
    char    *ptr = lpszQueryString;
    BOOL    bCustIdBlank;
    BOOL    fRemoteFlag = FALSE;

    pPaymentData->d_id = GetIntKeyValue(&ptr, "DID*",
ERR_PAYMENT_MISSING_DID_KEY,
ERR_PAYMENT_DISTRICT_INVALID);

    GetKeyValue(&ptr, "CID*", szTmp, sizeof(szTmp),
ERR_PAYMENT_MISSING_CID_KEY);
    if ( szTmp[0] == 0 )

```

```

        {
            bCustIdBlank = TRUE;
            pPaymentData->c_id = 0;
        }
        else
        {
            // parse customer id and verify that last name
was NOT entered
            bCustIdBlank = FALSE;
            if ( !IsNumeric(szTmp))
                throw new CWEBCLNT_ERR(
ERR_PAYMENT_CUSTOMER_INVALID);
            pPaymentData->c_id = atoi(szTmp);
        }

        pPaymentData->c_w_id = GetIntKeyValue(&ptr, "CWI*",
ERR_PAYMENT_MISSING_CWI_KEY,
ERR_PAYMENT_CWI_INVALID);
        pPaymentData->c_d_id = GetIntKeyValue(&ptr, "CDI*",
ERR_PAYMENT_MISSING_CDI_KEY,
ERR_PAYMENT_CDI_INVALID);
        if (!pLocalWhsTbl[pPaymentData->c_w_id])

fRemoteFlag=TRUE;

        if ( bCustIdBlank )
        {
            // customer id is blank, so last name must be
entered
            GetKeyValue(&ptr, "CLT*", szTmp,
sizeof(szTmp), ERR_PAYMENT_MISSING_CLT_KEY);
            if ( szTmp[0] == 0 )
                throw new CWEBCLNT_ERR(
ERR_PAYMENT_MISSING_CID_CLT);

            _strupr( szTmp );
            if ( strlen(szTmp) > LAST_NAME_LEN )
                throw new CWEBCLNT_ERR(
ERR_PAYMENT_LAST_NAME_TO_LONG);
            strcpy(pPaymentData->c_last, szTmp);
        }
        else
        {
            // parse customer id and verify that last name
was NOT entered
            GetKeyValue(&ptr, "CLT*", szTmp,
sizeof(szTmp), ERR_PAYMENT_MISSING_CLT_KEY);
            if ( szTmp[0] != 0 )
                throw new CWEBCLNT_ERR(
ERR_PAYMENT_CID_AND_CLT);
        }

        GetKeyValue(&ptr, "HAM*", szTmp, sizeof(szTmp),
ERR_PAYMENT_MISSING_HAM_KEY);
        if (!IsDecimal(szTmp))
            throw new CWEBCLNT_ERR(
ERR_PAYMENT_HAM_INVALID);
        pPaymentData->h_amount = atof(szTmp);
        if ( pPaymentData->h_amount >= 10000.00 ||
pPaymentData->h_amount < 0 )
            throw new CWEBCLNT_ERR(
ERR_PAYMENT_HAM_RANGE);
        if (pfLocalFlag)
            *pfLocalFlag = !fRemoteFlag;
    }

/* FUNCTION: GetOrderStatusData
*
* PURPOSE:      This function extracts and validates the payment
form data from an http command string.

```

```

*
*/
void GetOrderStatusData(LPSTR lpszQueryString,
ORDER_STATUS_DATA *pOrderStatusData)
{
    char    szTmp[26];
    char    *ptr = lpszQueryString;

    pOrderStatusData->d_id = GetIntKeyValue(&ptr, "DID*",
ERR_ORDERSTATUS_MISSING_DID_KEY,
ERR_ORDERSTATUS_DID_INVALID);

    GetKeyValue(&ptr, "CID*", szTmp, sizeof(szTmp),
ERR_ORDERSTATUS_MISSING_CID_KEY);
    if ( szTmp[0] == 0 )
    {
        // customer id is blank, so last name must be
entered
        pOrderStatusData->c_id = 0;
        GetKeyValue(&ptr, "CLT*", szTmp,
sizeof(szTmp), ERR_ORDERSTATUS_MISSING_CLT_KEY);
        if ( szTmp[0] == 0 )
            throw new CWEBCLNT_ERR(
ERR_ORDERSTATUS_MISSING_CID_CLT);

        _strupr( szTmp );
        if ( strlen(szTmp) > LAST_NAME_LEN )
            throw new CWEBCLNT_ERR(
ERR_ORDERSTATUS_CLT_RANGE);
        //
        strcpy(pOrderStatusData->in_c_last, szTmp);
        strcpy(pOrderStatusData->c_last, szTmp);
    }
    else
    {
        // parse customer id and verify that last name
was NOT entered
        if ( !IsNumeric(szTmp))
            throw new CWEBCLNT_ERR(
ERR_ORDERSTATUS_CID_INVALID);
        pOrderStatusData->c_id = atoi(szTmp);
        GetKeyValue(&ptr, "CLT*", szTmp,
sizeof(szTmp), ERR_ORDERSTATUS_MISSING_CLT_KEY);
        if ( szTmp[0] != 0 )
            throw new CWEBCLNT_ERR(
ERR_ORDERSTATUS_CID_AND_CLT);
    }
}

/* FUNCTION: BOOL IsNumeric(char *ptr)
*
* PURPOSE:      This function determines if a string is numeric. It
fails if any characters other
*
*               than numeric and null terminator are
present.
*
* ARGUMENTS:   char                *ptr
pointer to string to check.
*
* RETURNS:     BOOL    FALSE    if string is not
all numeric
*
TRUE    if string contains only numeric characters i.e. '0' - '9'
*/

BOOL IsNumeric(char *ptr)
{
    if ( *ptr == 0 )
        return FALSE;

```

```

        while( *ptr && isdigit(*ptr) )
            ptr++;
        return ( !*ptr );
    }

/* FUNCTION: BOOL IsDecimal(char *ptr)
 *
 * PURPOSE:      This function determines if a string is a
non-negative decimal value.
 *      It fails if any characters other than a series of numbers
followed by
 *              a decimal point, another series of
numbers, and a null terminator are present.
 *
 * ARGUMENTS:   char                *ptr
pointer to string to check.
 *
 * RETURNS:     BOOL    FALSE    if string is not
a valid non-negative decimal value
TRUE    if string is OK
 */

BOOL IsDecimal(char *ptr)
{
    char *dotptr;
    BOOL bValid;

    if ( *ptr == 0 )
        return FALSE;

    // find decimal point
    dotptr = strchr( ptr, '.' );
    if (dotptr == NULL)
        // no decimal point, so just check for numeric
        return IsNumeric(ptr);
    *dotptr = 0; // temporarily replace decimal with a
terminator

    if ( *ptr != 0 )
        bValid = IsNumeric(ptr);
    // string starts with decimal point
    else if (*(dotptr+1) == 0)
        return FALSE; // nothing but a decimal point is
bad
    else
        bValid = TRUE;

    if (*(dotptr+1) != 0)
        // check text after decimal point
        bValid &= IsNumeric(dotptr+1);

    *dotptr = '.'; // replace decimal point
    return bValid;
}

```

## tpc.def

```

LIBRARY TPCC.DLL

EXPORTS

    GetExtensionVersion @1
    HttpExtensionProc   @2
    TerminateExtension  @3

```

## tpcc.h

```

/*      FILE:          TPCC.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 ISAPI TPCC.DLL,
defines structures and functions used in the isapi tpcc.dll.
 *
 */

//VERSION RESOURCE DEFINES
#define _APS_NEXT_RESOURCE_VALUE
    101
#define _APS_NEXT_COMMAND_VALUE
    40001
#define _APS_NEXT_CONTROL_VALUE
    1000
#define _APS_NEXT_SYMED_VALUE
    101

#define TP_MAX_RETRIES
    50

//note that the welcome form must be processed first as terminal ids
assigned here, once the
//terminal id is assigned then the forms can be processed in any order.
#define WELCOME_FORM
    1 //beginning form no term id assigned,
form id
#define MAIN_MENU_FORM
    2 //term id assigned main menu form id
#define NEW_ORDER_FORM
    3 //new order form id
#define PAYMENT_FORM
    4 //payment form id
#define DELIVERY_FORM
    5 //delivery form id
#define ORDER_STATUS_FORM
    6 //order status id
#define STOCK_LEVEL_FORM
    7 //stock level form id

//This macro is used to prevent the compiler error unused formal
parameter
#define UNUSEDPARAM(x) (x = x)

//This structure defines the data necessary to keep distinct for each
terminal or client connection.
typedef struct _CLIENTDATA
{
    int                iNextFree;
    //index of next free element or -1 if this entry in use.
    int                w_id;
    //warehouse id assigned at welcome form
    int                d_id;
    //district id assigned at welcome form
}

```

```

int iSyncId;
//synchronization id
int iTickCount;
//time of last access;

CTPCC_BASE *pTxn;
} CLIENTDATA, *PCLIENTDATA;

//This structure is used to define the operational interface for terminal
id support
typedef struct _TERM
{
int iNumEntries;
//total allocated terminal array entries
int iFreeList;
//next available terminal array
element or -1 if none
int iMasterSyncId;
//synchronization id
CLIENTDATA *pClientData;
//pointer to allocated client data
} TERM;

typedef TERM *PTERM;
//pointer to terminal structure type

enum WEBERROR
{
NO_ERR,
ERR_COMMAND_UNDEFINED,
ERR_D_ID_INVALID,
ERR_DELIVERY_CARRIER_ID_RANGE,
ERR_DELIVERY_CARRIER_INVALID,
ERR_DELIVERY_MISSING_OCD_KEY,
ERR_DELIVERY_THREAD_FAILED,
ERR_GETPROCADDR_FAILED,
ERR_HTML_ILL_FORMED,
ERR_INVALID_SYNC_CONNECTION,
ERR_INVALID_TERMID,
ERR_LOADDLL_FAILED,
ERR_MAX_CONNECTIONS_EXCEEDED,
ERR_MEM_ALLOC_FAILED,
ERR_MISSING_REGISTRY_ENTRIES,
ERR_NEWORDER_CUSTOMER_INVALID,
ERR_NEWORDER_CUSTOMER_KEY,
ERR_NEWORDER_DISTRICT_INVALID,
ERR_NEWORDER_FORM_MISSING_DID,
ERR_NEWORDER_ITEMID_INVALID,
ERR_NEWORDER_ITEMID_RANGE,

ERR_NEWORDER_ITEMID_WITHOUT_SUPPW,
ERR_NEWORDER_MISSING_IID_KEY,
ERR_NEWORDER_MISSING_QTY_KEY,
ERR_NEWORDER_MISSING_SUPPW_KEY,
ERR_NEWORDER_NOITEMS_ENTERED,
ERR_NEWORDER_QTY_INVALID,
ERR_NEWORDER_QTY_RANGE,

ERR_NEWORDER_QTY_WITHOUT_SUPPW,
ERR_NEWORDER_SUPPW_INVALID,
ERR_NO_SERVER_SPECIFIED,
ERR_ORDERSTATUS_CID_AND_CLT,
ERR_ORDERSTATUS_CID_INVALID,
ERR_ORDERSTATUS_CLT_RANGE,
ERR_ORDERSTATUS_DID_INVALID,

```

```

ERR_ORDERSTATUS_MISSING_CID_CLT,
ERR_ORDERSTATUS_MISSING_CID_KEY,
ERR_ORDERSTATUS_MISSING_CLT_KEY,
ERR_ORDERSTATUS_MISSING_DID_KEY,
ERR_PAYMENT_CDI_INVALID,
ERR_PAYMENT_CID_AND_CLT,
ERR_PAYMENT_CUSTOMER_INVALID,
ERR_PAYMENT_CWI_INVALID,
ERR_PAYMENT_DISTRICT_INVALID,
ERR_PAYMENT_HAM_INVALID,
ERR_PAYMENT_HAM_RANGE,
ERR_PAYMENT_LAST_NAME_TO_LONG,
ERR_PAYMENT_MISSING_CDI_KEY,
ERR_PAYMENT_MISSING_CID_CLT,
ERR_PAYMENT_MISSING_CID_KEY,
ERR_PAYMENT_MISSING_CLT,
ERR_PAYMENT_MISSING_CLT_KEY,
ERR_PAYMENT_MISSING_CWI_KEY,
ERR_PAYMENT_MISSING_DID_KEY,
ERR_PAYMENT_MISSING_HAM_KEY,

ERR_STOCKLEVEL_MISSING_THRESHOLD_KEY,

ERR_STOCKLEVEL_THRESHOLD_INVALID,
ERR_STOCKLEVEL_THRESHOLD_RANGE,
ERR_VERSION_MISMATCH,
ERR_W_ID_INVALID
};

class CWEBCLNT_ERR : public CBaseErr
{
public:
CWEBCLNT_ERR(WEBERROR Err)
{
m_Error = Err;
m_szTextDetail = NULL;
m_SystemErr = 0;
m_szErrorText = NULL;
};

CWEBCLNT_ERR(WEBERROR Err, char
*szTextDetail, DWORD dwSystemErr)
{
m_Error = Err;
m_szTextDetail = new
char[strlen(szTextDetail)+1];
strcpy( m_szTextDetail, szTextDetail
);
m_SystemErr = dwSystemErr;
m_szErrorText = NULL;
};

~CWEBCLNT_ERR()
{
if (m_szTextDetail != NULL)
delete [] m_szTextDetail;
if (m_szErrorText != NULL)
delete [] m_szErrorText;
};

WEBERROR m_Error;
char *m_szTextDetail; //
char *m_szErrorText;
DWORD m_SystemErr;

int ErrorType() {return
ERR_TYPE_WEBDLL;};

```

```

int ErrorNum() {return m_Error;};
char *ErrorText();

};

//These constants have already been defined in engstut.h, but since we
do
//not want to include it in the delisrv executable
#define TXN_EVENT_START 2
#define TXN_EVENT_STOP 4
#define TXN_EVENT_WARNING 6
//used to record a warning into the log

//function prototypes

BOOL APIENTRY DllMain(HANDLE hModule, DWORD
ul_reason_for_call, LPVOID lpReserved);
void WriteMessageToEventLog(LPTSTR lpszMsg);
void ProcessQueryString(EXTENSION_CONTROL_BLOCK *pECB,
int *pCmd, int *pFormId, int *pTermId, int *pSyncId);
void WelcomeForm(EXTENSION_CONTROL_BLOCK *pECB, char
*szBuffer);
void SubmitCmd(EXTENSION_CONTROL_BLOCK *pECB, char
*szBuffer);
void BeginCmd(EXTENSION_CONTROL_BLOCK *pECB, int
iFormId, int iTermId);
void ProcessCmd(EXTENSION_CONTROL_BLOCK *pECB, int
iFormId, int iTermId);
void StatsCmd(EXTENSION_CONTROL_BLOCK *pECB, char
*szBuffer);
void ErrorMessage(EXTENSION_CONTROL_BLOCK *pECB, int
iError, int iErrorType, char *szMsg, int iTermId);
void GetKeyValue(char **pQueryString, char *pKey, char *pValue,
int iMax, WEBERROR err);
int GetIntKeyValue(char **pQueryString, char *pKey, WEBERROR
NoKeyErr, WEBERROR NotIntErr);
void TermInit(void);
void TermDeleteAll(void);
int TermAdd(void);
void TermDelete(int id);
void ErrorForm(EXTENSION_CONTROL_BLOCK *pECB, int
iType, int iErrorNum, int iTermId, int iSyncId, char *szErrorText, char
*szBuffer );
void MakeMainMenuForm(int iTermId, int iSyncId, char *szForm);
void MakeStockLevelForm(int iTermId, STOCK_LEVEL_DATA
*pStockLevelData, BOOL bInput, char *szForm);
void MakeNewOrderForm(int iTermId, NEW_ORDER_DATA
*pNewOrderData, BOOL bInput, char *szForm);
void MakePaymentForm(int iTermId, PAYMENT_DATA
*pPaymentData, BOOL bInput, char *szForm);
void MakeOrderStatusForm(int iTermId, ORDER_STATUS_DATA
*pOrderStatusData, BOOL bInput, char *szForm);
void MakeDeliveryForm(int iTermId, DELIVERY_DATA
*pDeliveryData, BOOL bInput, char *szForm);
void ProcessNewOrderForm(EXTENSION_CONTROL_BLOCK
*pECB, int iTermId, char *szBuffer);
void ProcessPaymentForm(EXTENSION_CONTROL_BLOCK
*pECB, int iTermId, char *szBuffer);
void ProcessOrderStatusForm(EXTENSION_CONTROL_BLOCK
*pECB, int iTermId, char *szBuffer);
void ProcessDeliveryForm(EXTENSION_CONTROL_BLOCK
*pECB, int iTermId, char *szBuffer);
void ProcessStockLevelForm(EXTENSION_CONTROL_BLOCK
*pECB, int iTermId, char *szBuffer);
void GetNewOrderData(LPSTR lpszQueryString,
NEW_ORDER_DATA *pNewOrderData, BOOL *pfLocalFlag);
void GetPaymentData(LPSTR lpszQueryString, PAYMENT_DATA
*pPaymentData, BOOL *pfLocalFlag);

```

```

void GetOrderStatusData(LPSTR lpszQueryString,
ORDER_STATUS_DATA *pOrderStatusData);
BOOL PostDeliveryInfo(long w_id, short o_carrier_id);
BOOL IsNumeric(char *ptr);
BOOL IsDecimal(char *ptr);
void DeliveryWorkerThread(void *ptr);

```

## tpcc.rc

```

//Microsoft Developer Studio generated resource script.
//
#include "resource.h"

#define APSTUDIO_READONLY_SYMBOLS
//
// Generated from the TEXTINCLUDE2 resource.
//
#include "afxres.h"

//
// English (U.S.) resources
//
#ifndef AFX_RESOURCE_DLL || defined AFX_TARG_ENU
#ifdef _WIN32
LANGUAGE LANG_ENGLISH, SUBLANG_ENGLISH_US
#pragma code_page(1252)
#endif // _WIN32

#ifndef _MAC
//
// Version
//
VS_VERSION_INFO VERSIONINFO
FILEVERSION 0,4,0,0
PRODUCTVERSION 0,4,0,0
FILEFLAGSMASK 0x3fL
#ifdef _DEBUG
FILEFLAGS 0x1L
#else
FILEFLAGS 0x0L
#endif
FILEOS 0x40004L
FILETYPE 0x2L
FILESUBTYPE 0x0L
BEGIN
    BLOCK "StringFileInfo"
    BEGIN
        BLOCK "040904b0"
        BEGIN
            VALUE "Comments", "TPC-C HTML DLL Server (DBLIB)\0"
            VALUE "CompanyName", "Microsoft\0"
            VALUE "FileDescription", "TPC-C HTML DLL Server
(DBLIB)\0"
            VALUE "FileVersion", "0, 4, 0, 0\0"
            VALUE "InternalName", "tpcc\0"
            VALUE "LegalCopyright", "Copyright © 1997\0"
            VALUE "OriginalFilename", "tpcc.dll\0"
            VALUE "ProductName", "Microsoft tpcc\0"
            VALUE "ProductVersion", "0, 4, 0, 0\0"
        END
    END

```

```

END
BLOCK "VarFileInfo"
BEGIN
    VALUE "Translation", 0x409, 1200
END
END

#endif // !_MAC

#ifdef APSTUDIO_INVOKED
//
// TEXTINCLUDE
//

1 TEXTINCLUDE DISCARDABLE
BEGIN
    "resource.h(0)"
END

2 TEXTINCLUDE DISCARDABLE
BEGIN
    "#include ""afxres.h""\r\n"
    ""\0"
END

3 TEXTINCLUDE DISCARDABLE
BEGIN
    "\r\n"
    ""\0"
END

#endif // APSTUDIO_INVOKED

//
// Dialog
//

IDD_DIALOG1 DIALOG DISCARDABLE 0, 0, 186, 95
STYLE DS_MODALFRAME | WS_POPUP | WS_CAPTION |
WS_SYSMENU
CAPTION "Dialog"
FONT 8, "MS Sans Serif"
BEGIN
    DEFPUSHBUTTON "OK", IDOK, 129, 7, 50, 14
    PUSHBUTTON "Cancel", IDCANCEL, 129, 24, 50, 14
END

//
// DESIGNINFO
//

#ifdef APSTUDIO_INVOKED
GUIDELINES DESIGNINFO DISCARDABLE
BEGIN
    IDD_DIALOG1, DIALOG
    BEGIN
        LEFTMARGIN, 7
        RIGHTMARGIN, 179
        TOPMARGIN, 7
        BOTTOMMARGIN, 88
    END
END

```

```

#endif // APSTUDIO_INVOKED

#endif // English (U.S.) resources
//

#ifdef APSTUDIO_INVOKED
//
// Generated from the TEXTINCLUDE3 resource.
//

#endif // not APSTUDIO_INVOKED

tpcc_com.cpp

/*      FILE:          TPCC_COM.CPP
 *                               Microsoft TPC-C Kit Ver.
4.20.000
 *                               Copyright Microsoft,
1999
 *                               All Rights Reserved
 *
 *                               not yet audited
 *
 *      PURPOSE:       Source file for TPC-C COM+ class
implementation.
 *      Contact:      Charles Levine (clevine@microsoft.com)
 *
 *      Change history:
 *                               4.20.000 - first version
 */

// needed for CoInitializeEx
#define _WIN32_WINNT 0x0400

#include <windows.h>

// need to declare functions for export
#define DllDecl __declspec( dllexport )

#include "..\..\common\src\trans.h" //tpckit
transaction header contains definations of structures specific to TPC-C
#include "..\..\common\src\error.h"
#include "..\..\common\src\txn_base.h"
#include "tpcc_com.h"

#include "..\..\tpcc_com_ps\src\tpcc_com_ps_i.c"
#include "..\..\tpcc_com_all\src\tpcc_com_all_i.c"

#include "..\..\tpcc_com_remote_ps\src\tpcc_com_remote_ps_i.c"
#include "..\..\tpcc_com_remote\src\tpcc_com_remote_i.c"

// wrapper routine for class constructor
__declspec( dllexport ) CTPCC_COM* CTPCC_COM_new(BOOL
bSinglePool)
{
    return new CTPCC_COM(bSinglePool);
}

CTPCC_COM::CTPCC_COM(BOOL bSinglePool)
{

```



```

HRESULT hr = NULL;
long lRet = 0;
ULONG ulTmpSize = 0;

m_bSinglePool = bSinglePool;

m_pNewOrder          = NULL;
m_pPayment           = NULL;
m_pStockLevel        = NULL;
m_pOrderStatus       = NULL;
m_p_remNewOrder      = NULL;
m_p_remPayment       = NULL;
m_p_remStockLevel    = NULL;
m_p_remOrderStatus   = NULL;

/*
m_pTxn =
(COM_DATA*)CoTaskMemAlloc(sizeof(COM_DATA));
if (!m_pTxn)
    throw new CCOMERR( E_FAIL );
*/
ulTmpSize = (ULONG) sizeof(COM_DATA);
VariantInit(&m_vTxn);
m_vTxn.vt = VT_SAFEARRAY;

m_vTxn.parray = SafeArrayCreateVector(
VT_UI1,
ulTmpSize,
ulTmpSize);
if (!m_vTxn.parray)
    throw new CCOMERR( E_FAIL );

memset((void*)m_vTxn.parray->pvData,0,ulTmpSize);
m_pTxn = (COM_DATA*)m_vTxn.parray->pvData;

hr = CoInitializeEx(NULL, COINIT_MULTITHREADED);
if (FAILED(hr))
{
    throw new CCOMERR( hr );
}

// create components
if (m_bSinglePool)
{
    hr = CoCreateInstance(CLSID_TPCC, NULL,
CLSCTX_SERVER, IID_ITPCC, (void **)&m_pNewOrder);
    if (FAILED(hr))
        throw new CCOMERR(hr);

    hr = CoCreateInstance(CLSID_TPCCrem,
NULL, CLSCTX_SERVER, IID_ITPCCrem, (void
**)&m_p_remNewOrder);
    if (FAILED(hr))
        throw new CCOMERR(hr);

    // all local txns will use same component
    m_pPayment = m_pNewOrder;
    m_pStockLevel = m_pNewOrder;
    m_pOrderStatus = m_pNewOrder;

    // all remote txns will use same component
    m_p_remPayment=m_p_remNewOrder;
    m_p_remStockLevel = m_p_remNewOrder;

    m_p_remOrderStatus = m_p_remNewOrder;
}
else
{
    // use different components for each txn

    hr = CoCreateInstance(CLSID_NewOrder,
NULL, CLSCTX_SERVER, IID_ITPCC, (void **)&m_pNewOrder);
    if (FAILED(hr))
        throw new CCOMERR(hr);

    hr = CoCreateInstance(CLSID_Payment, NULL,
CLSCTX_SERVER, IID_ITPCC, (void **)&m_pPayment);
    if (FAILED(hr))
        throw new CCOMERR(hr);

    hr = CoCreateInstance(CLSID_StockLevel,
NULL, CLSCTX_SERVER, IID_ITPCC, (void **)&m_pStockLevel);
    if (FAILED(hr))
        throw new CCOMERR(hr);

    hr = CoCreateInstance(CLSID_OrderStatus,
NULL, CLSCTX_SERVER, IID_ITPCC, (void **)&m_pOrderStatus);
    if (FAILED(hr))
        throw new CCOMERR(hr);
}

// call setcomplete to release each component back into pool
hr = m_pNewOrder->CallSetComplete();
if (FAILED(hr))
    throw new CCOMERR(hr);

hr = m_p_remNewOrder->CallSetComplete();
if (FAILED(hr))
    throw new CCOMERR(hr);

if (!m_bSinglePool)
{
    hr = m_pPayment->CallSetComplete();
    if (FAILED(hr))
        throw new CCOMERR(hr);

    hr = m_pStockLevel->CallSetComplete();
    if (FAILED(hr))
        throw new CCOMERR(hr);

    hr = m_pOrderStatus->CallSetComplete();
    if (FAILED(hr))
        throw new CCOMERR(hr);
}

CTPCC_COM::~~CTPCC_COM()
{
    if (m_pTxn)
        SafeArrayDestroy(m_vTxn.parray);

    ReleaseInterface(m_pNewOrder);

    ReleaseInterface(m_p_remNewOrder);

    if (!m_bSinglePool)
    {
        ReleaseInterface(m_pPayment);
        ReleaseInterface(m_pStockLevel);
        ReleaseInterface(m_pOrderStatus);
    }
}

```

```

        CoUninitialize();
    }
    /*void CTPCC_COM::NewOrder()
    {
        VARIANT vTxn_out;
        HRESULT hr = m_pNewOrder->NewOrder(m_vTxn,
        &vTxn_out);

        if (FAILED(hr))
            throw new CCOMERR( hr );
        memcpy(m_pTxn, (void
        *)vTxn_out.parray->pvData,vTxn_out.parray->rgsabound[0].cElement
        s);
        SafeArrayDestroy(vTxn_out.parray);

        if ( m_pTxn->ErrorType != ERR_SUCCESS )
            throw new CCOMERR( m_pTxn->ErrorType,
        m_pTxn->error );
    }
    */
    void CTPCC_COM::NewOrder(BOOL fLocal)
    {
        VARIANT vTxn_out;
        HRESULT hr;
        if (fLocal)
            hr = m_pNewOrder->NewOrder(m_vTxn, &vTxn_out);
        else
            hr = m_p_remNewOrder->NewOrder(m_vTxn,
        &vTxn_out);

        if (FAILED(hr))
            throw new CCOMERR( hr );
        memcpy(m_pTxn, (void
        *)vTxn_out.parray->pvData,vTxn_out.parray->rgsabound[0].cElement
        s);
        SafeArrayDestroy(vTxn_out.parray);

        if ( m_pTxn->ErrorType != ERR_SUCCESS )
            throw new CCOMERR( m_pTxn->ErrorType,
        m_pTxn->error );
    }
    /*void CTPCC_COM::Payment()
    {
        VARIANT vTxn_out;
        HRESULT hr = m_pPayment->Payment(m_vTxn,
        &vTxn_out);

        if (FAILED(hr))
            throw new CCOMERR( hr );
        memcpy(m_pTxn, (void
        *)vTxn_out.parray->pvData,vTxn_out.parray->rgsabound[0].cElement
        s);
        SafeArrayDestroy(vTxn_out.parray);

        if ( m_pTxn->ErrorType != ERR_SUCCESS )
            throw new CCOMERR( m_pTxn->ErrorType,
        m_pTxn->error );
    }
    */
    void CTPCC_COM::Payment(BOOL fLocal)
    {
        VARIANT vTxn_out;
        HRESULT hr;
        if (fLocal)
            hr = m_pPayment->Payment(m_vTxn,
        &vTxn_out);
        else

```

```

        hr = m_p_remPayment->Payment(m_vTxn,
        &vTxn_out);

        if (FAILED(hr))
            throw new CCOMERR( hr );
        memcpy(m_pTxn, (void
        *)vTxn_out.parray->pvData,vTxn_out.parray->rgsabound[0].cElement
        s);
        SafeArrayDestroy(vTxn_out.parray);

        if ( m_pTxn->ErrorType != ERR_SUCCESS )
            throw new CCOMERR( m_pTxn->ErrorType,
        m_pTxn->error );
    }
    void CTPCC_COM::StockLevel()
    {
        VARIANT vTxn_out;

        HRESULT hr = m_pStockLevel->StockLevel(m_vTxn,
        &vTxn_out);
        if (FAILED(hr))
            throw new CCOMERR( hr );
        memcpy(m_pTxn, (void
        *)vTxn_out.parray->pvData,vTxn_out.parray->rgsabound[0].cElement
        s);
        SafeArrayDestroy(vTxn_out.parray);

        if ( m_pTxn->ErrorType != ERR_SUCCESS )
            throw new CCOMERR( m_pTxn->ErrorType,
        m_pTxn->error );
    }
    void CTPCC_COM::OrderStatus()
    {
        VARIANT vTxn_out;

        HRESULT hr = m_pOrderStatus->OrderStatus(m_vTxn,
        &vTxn_out);
        if (FAILED(hr))
            throw new CCOMERR( hr );
        memcpy(m_pTxn, (void
        *)vTxn_out.parray->pvData,vTxn_out.parray->rgsabound[0].cElement
        s);
        SafeArrayDestroy(vTxn_out.parray);

        if ( m_pTxn->ErrorType != ERR_SUCCESS )
            throw new CCOMERR( m_pTxn->ErrorType,
        m_pTxn->error );
    }

```

### **tpcc\_com.h**

```

/*      FILE:          TPCC_COM.H
 *
 *      4.20.000      Microsoft TPC-C Kit Ver.
 *
 *      1999          Copyright Microsoft,
 *
 *      All Rights Reserved
 *
 *
 *      not yet audited
 *

```

```

*      PURPOSE:      Header file for TPC-C COM+ class
implementation.
*
* Change history:
*           4.20.000 - first version
*/

#pragma once

#include <stdio.h>
#include "..\..\tpcc_com_ps\src\tpcc_com_ps.h"
#include "..\..\tpcc_com_remote_ps\src\tpcc_com_remote_ps.h"

// need to declare functions for import, unless define has already been
created
// by the DLL's .cpp module for export.
#ifdef DllDecl
#define DllDecl __declspec( dllimport )
#endif

class CCOMERR : public CBaseErr
{
private:
    char m_szErrorText[64];

public:
    // use this interface for genuine COM errors
    CCOMERR( HRESULT hr )
    {
        m_hr = hr;
        m_iErrorType = 0;
        m_iError = 0;
    }

    // use this interface to impersonate a non-COM
error type
    CCOMERR( int iErrorType, int iError )
    {
        m_iErrorType = iErrorType;
        m_iError = iError;
        m_hr = S_OK;
    }

    int          m_hr;
    int          m_iErrorType;
    int          m_iError;

```

```

// A CCOMERR class can impersonate another
class, which happens if the error
// was not actually a COM Services error, but
was simply transmitted back via COM.
    int ErrorType()
    {
        if (m_iErrorType == 0)
            return ERR_TYPE_COM;
        else
            return m_iErrorType;
    }

    int ErrorNum() {return m_hr;}

    char *ErrorText()
    {
        if (m_hr == S_OK)
            sprintf( m_szErrorText,
"Error: Class %d, error # %d", m_iErrorType, m_iError );
        else
            sprintf( m_szErrorText,
"Error: COM HRESULT %x", m_hr );
        return m_szErrorText;
    }
};

class DllDecl CTPCC_COM : public CTPCC_BASE
{
private:
    BOOL m_bSinglePool;

    // COM Interface pointers
    ITPCC*          m_pNewOrder;
    ITPCC*          m_pPayment;
    ITPCC*
m_pStockLevel;
    ITPCC*
m_pOrderStatus;
    ITPCCrem*
m_p_remNewOrder;
    ITPCCrem*
m_p_remPayment;
    ITPCCrem*
m_p_remStockLevel;
    ITPCCrem*
m_p_remOrderStatus;

    struct COM_DATA

```

```

        {
            int ErrorType;
            int error;
            union
            {
                NEW_ORDER_DATA
                Payment;
                PAYMENT_DATA
                Delivery;
                DELIVERY_DATA
                StockLevel;
                STOCK_LEVEL_DATA
                ORDER_STATUS_DATA OrderStatus;
            } u;
        } *m_pTxn;

        VARIANT m_vTxn;

    public:
        CTPCC_COM(BOOL bSinglePool);
        ~CTPCC_COM(void);

        inline PNEW_ORDER_DATA
        BuffAddr_NewOrder() { return
        &m_pTxn->u.NewOrder; };

        inline PPAYMENT_DATA
        BuffAddr_Payment() { return &m_pTxn->u.Payment; };

        inline PDELIVERY_DATA
        BuffAddr_Delivery() { return &m_pTxn->u.Delivery; };

        inline PSTOCK_LEVEL_DATA
        BuffAddr_StockLevel() { return &m_pTxn->u.StockLevel; };

        inline PORDER_STATUS_DATA
        BuffAddr_OrderStatus() { return &m_pTxn->u.OrderStatus;
        };

//          void NewOrder          ();
//          void NewOrder          (BOOL
fLocal);
//          void Payment           ();
//          void Payment           (BOOL
fLocal);
        void StockLevel           ();
        void OrderStatus          ();
        void Delivery              () { throw new
CCOMERR(E_NOTIMPL); } // not supported
};

inline void ReleaseInterface(IUnknown *pUnk)

```

```

    {
        if (pUnk)
        {
            pUnk->Release();
            pUnk = NULL;
        }
    }

// wrapper routine for class constructor
extern "C" __declspec(dllexport) CTPCC_COM*
CTPCC_COM_new(BOOL);

typedef CTPCC_COM* (TYPE_CTPCC_COM)(BOOL);

tpcc_com_all.cpp

/*      FILE:          TPCC_COM_ALL.CPP
 *          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
class.
 *      Contact:     Charles Levine (clevine@microsoft.com)
 *
 *      Change history:
 *          4.20.000 - updated rev number to match kit
 */

#define STRICT
#define _WIN32_WINNT 0x0400
#define _ATL_APARTMENT_THREADED

#include <stdio.h>
#include <atlbases.h>
//You may derive a class from CComModule and use it if you want to
override
//something, but do not change the name of _Module
extern CComModule _Module;

#include <atlcom.h>
#include <initguid.h>
#include <transact.h>
#include <atlimpl.cpp>
#include <comsvcs.h>

#include <sqltypes.h>
#include <sql.h>
#include <sqltext.h>

#include "tpcc_com_ps.h"
#include "..\..\common\src\trans.h"
//tpckit transaction header contains definations of structures
specific to TPC-C

```

```

#include "..\..\common\src\txn_base.h"
#include "..\..\common\src\error.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 "resource.h"
#include "tpcc_com_all.h"
#include "tpcc_com_all_i.c"
#include "Methods.h"
#include "..\..\tpcc_com_ps\src\tpcc_com_ps_i.c"
#include "..\..\common\src\ReadRegistry.cpp"

CComModule _Module;

BEGIN_OBJECT_MAP(ObjectMap)
    OBJECT_ENTRY(CLSID_TPCC, CTPCC)
    OBJECT_ENTRY(CLSID_NewOrder, CNewOrder)
    OBJECT_ENTRY(CLSID_OrderStatus, COrderStatus)
    OBJECT_ENTRY(CLSID_Payment, CPayment)
    OBJECT_ENTRY(CLSID_StockLevel, CStockLevel)
END_OBJECT_MAP()

// configuration settings from registry
TPCCREGISTRYDATA Reg;
char
szMyComputerName[MAX_COMPUTERNAME_LENGTH+1];

static HINSTANCE hLibInstanceDb = NULL;

TYPE_CTPCC_DBLIB *pCTPCC_DBLIB_new;
TYPE_CTPCC_ODBC *pCTPCC_ODBC_new;

////////////////////////////////////
// DLL Entry Point

extern "C"
BOOL WINAPI DllMain(HINSTANCE hInstance, DWORD
dwReason, LPVOID /*lpReserved*/)
{
    char szDllName[128];

    try
    {
        if (dwReason == DLL_PROCESS_ATTACH)
        {
            _Module.Init(ObjectMap, hInstance);

DisableThreadLibraryCalls(hInstance);

            DWORD dwSize =
MAX_COMPUTERNAME_LENGTH+1;

GetComputerName(szMyComputerName, &dwSize);
            szMyComputerName[dwSize] = 0;

            if ( ReadTPCCRegistrySettings(
&Reg ) )
                throw new
CCOMPONENT_ERR( ERR_MISSING_REGISTRY_ENTRIES);

            if (Reg.eDB_Protocol == DBLIB)

```

```

        {
            strcpy( szDllName,
Reg.szPath );

            strcat( szDllName,
"tpcc_dblib.dll");

LoadLibrary( szDllName );

            if (hLibInstanceDb ==
NULL)
                throw new
CCOMPONENT_ERR( ERR_LOADDLL_FAILED, szDllName,
GetLastError() );

            // get function pointer to
wrapper for class constructor
            pCTPCC_DBLIB_new =
(TYPE_CTPCC_DBLIB*)
GetProcAddress(hLibInstanceDb, "CTPCC_DBLIB_new");
            if (pCTPCC_DBLIB_new
== NULL)
                throw new
CCOMPONENT_ERR( ERR_GETPROCADDR_FAILED,
szDllName, GetLastError() );
        }
        else if (Reg.eDB_Protocol ==
ODBC)
        {
            strcpy( szDllName,
Reg.szPath );

            strcat( szDllName,
"tpcc_odbc.dll");

LoadLibrary( szDllName );

            if (hLibInstanceDb ==
NULL)
                throw new
CCOMPONENT_ERR( ERR_LOADDLL_FAILED, szDllName,
GetLastError() );

            // get function pointer to
wrapper for class constructor
            pCTPCC_ODBC_new =
(TYPE_CTPCC_ODBC*)
GetProcAddress(hLibInstanceDb, "CTPCC_ODBC_new");
            if (pCTPCC_ODBC_new
== NULL)
                throw new
CCOMPONENT_ERR( ERR_GETPROCADDR_FAILED,
szDllName, GetLastError() );
        }
        else
            throw new
CCOMPONENT_ERR( ERR_UNKNOWN_DB_PROTOCOL );
    }
    else if (dwReason ==
DLL_PROCESS_DETACH)
        _Module.Term();

    }
    catch (CBaseErr *e)
    {
        WriteMessageToEventLog(e->ErrorText());
        delete e;
        return FALSE;
    }
    catch (...)
    {

```

```

        WriteMessageToEventLog(TEXT("Unhandled
exception in object DllMain"));
        return FALSE;
    }

    return TRUE;    // OK
}

////////////////////////////////////
// Used to determine whether the DLL can be unloaded by OLE

STDAPI DllCanUnloadNow(void)
{
    return (_Module.GetLockCount()==0) ? S_OK : S_FALSE;
}

////////////////////////////////////
// Returns a class factory to create an object of the requested type

STDAPI DllGetClassObject(REFCLSID rclsid, REFIID riid, LPVOID*
ppv)
{
    return _Module.GetClassObject(rclsid, riid, ppv);
}

////////////////////////////////////
// DllRegisterServer - Adds entries to the system registry

STDAPI DllRegisterServer(void)
{
    // registers object, typelib and all interfaces in typelib
    return _Module.RegisterServer(TRUE);
}

////////////////////////////////////
// DllUnregisterServer - Removes entries from the system registry

STDAPI DllUnregisterServer(void)
{
    _Module.UnregisterServer();
    return S_OK;
}

static void WriteMessageToEventLog(LPTSTR lpszMsg)
{
    TCHAR szMsg[256];
    HANDLE hEventSource;
    LPTSTR lpszStrings[2];

    // Use event logging to log the error.
    //
    hEventSource = RegisterEventSource(NULL,
TEXT("tpcc_com_all.dll"));

    _sprintf(szMsg, TEXT("Error in COM+ TPC-C Component: "));
    lpszStrings[0] = szMsg;
    lpszStrings[1] = lpszMsg;

    if (hEventSource != NULL)
    {
        ReportEvent(hEventSource, // handle of event source
EVENTLOG_ERROR_TYPE, // event type
0, // event category
0, // event ID
NULL, // current user's SID
2, // strings in lpszStrings
0, // no bytes of raw data

```

```

(LPCTSTR *)lpszStrings, // array of error strings
NULL); // no raw data

(VOID) DeregisterEventSource(hEventSource);
}

inline void ReleaseInterface(IUnknown *pUnk)
{
    if (pUnk)
    {
        pUnk->Release();
        pUnk = NULL;
    }
}

/* FUNCTION: CCOMPONENT_ERR::ErrorText
*
*/
char* CCOMPONENT_ERR::ErrorText(void)
{
    static SERRORMSG errorMsgs[] =
    {
        { ERR_MISSING_REGISTRY_ENTRIES,
"Required entries missing from registry."
},
        { ERR_LOADDLL_FAILED,
"Load of DLL failed. DLL="
},
        { ERR_GETPROCADDR_FAILED,
"Could not map proc in DLL. GetProcAddr error. DLL="
},
        { ERR_UNKNOWN_DB_PROTOCOL,
"Unknown database protocol specified in registry."
},
        { 0,
""
}
    };

    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)
        sprintf( szTmp+strlen(szTmp), " Error=%d",
m_SystemErr );

    m_szErrorText = new char[strlen(szTmp)+1];
    strcpy( m_szErrorText, szTmp );
    return m_szErrorText;
}

```

```

CTPCC_Common::CTPCC_Common()
{
    m_pTxn = NULL;
    m_bCanBePooled = TRUE;
}

CTPCC_Common::~CTPCC_Common()
{
    if (m_pTxn)
        delete m_pTxn;
}

HRESULT CTPCC_Common::CallSetComplete()
{
    IObjectContext* pObjectContext = NULL;

    // get our object context
    HRESULT hr = CoGetObjectContext( IID_IObjectContext,
(void **)&pObjectContext );
    pObjectContext->SetComplete();
    ReleaseInterface(pObjectContext);
    return hr;
}

//
// called by the ctor activator
//
STDMETHODIMP CTPCC_Common::Construct(IDispatch * pUnk)
{
    // Code to access construction string, if needed later...
    // if (!pUnk)
    //     return E_UNEXPECTED;
    // IObjectConstructString * pString = NULL;
    // HRESULT hr =
pUnk->QueryInterface(IID_IObjectConstructString, (void
**)&pString);
    // pString->Release();
    try
    {
        if (Reg.eDB_Protocol == ODBC)
            m_pTxn = pCTPCC_ODBC_new(
Reg.szDbServer, Reg.szDbUser, Reg.szDbPassword,
szMyComputerName, Reg.szDbName );
        else if (Reg.eDB_Protocol == DBLIB)
            m_pTxn = pCTPCC_DBLIB_new(
Reg.szDbServer, Reg.szDbUser, Reg.szDbPassword,
szMyComputerName, Reg.szDbName );
    }
    catch (CBaseErr *e)
    {
        WriteMessageToEventLog(e->ErrorText());
        delete e;
        return E_FAIL;
    }
    catch (...)
    {
        WriteMessageToEventLog(TEXT("Unhandled
exception in object ::Construct"));
        return E_FAIL;
    }

    return S_OK;
}

```

```

HRESULT CTPCC_Common::NewOrder(VARIANT txn_in,
VARIANT* txn_out)
{
    PNEW_ORDER_DATA pNewOrder;
    COM_DATA *pData;
    try
    {
        pData = (COM_DATA*)txn_in.parray->pvData;
        pNewOrder = m_pTxn->BuffAddr_NewOrder();

        memcpy(pNewOrder, &pData->u.NewOrder,
sizeof(NEW_ORDER_DATA));

        VariantInit(txn_out);
        txn_out->vt = VT_SAFEARRAY;
        txn_out->parray = SafeArrayCreateVector(
VT_UI1,
txn_in.parray->rgsabound->cElements,
txn_in.parray->rgsabound->cElements);
        pData = (COM_DATA*)
txn_out->parray->pvData;

        m_pTxn->NewOrder(TRUE);

        memcpy(&pData->u.NewOrder, pNewOrder,
sizeof(NEW_ORDER_DATA));

        pData->retval = ERR_SUCCESS;
        pData->error = 0;
        return S_OK;
    }
    catch (CBaseErr *e)
    {
        // check for lost database connection; if yes,
component is toast
        if ( ((e->ErrorType() == ERR_TYPE_DBLIB)
&& (e->ErrorNum() == 10005)) ||
((e->ErrorType() ==
ERR_TYPE_ODBC) && (e->ErrorNum() == 10054)) )
            m_bCanBePooled = FALSE;

        pData->retval = e->ErrorType();
        pData->error = e->ErrorNum();
        delete e;
        return E_FAIL;
    }
    catch (...)
    {
        WriteMessageToEventLog(TEXT("Unhandled
exception."));

        pData->retval = ERR_TYPE_LOGIC;
        pData->error = 0;
        m_bCanBePooled = FALSE;
        return E_FAIL;
    }
}

HRESULT CTPCC_Common::Payment(VARIANT txn_in,
VARIANT* txn_out)
{
    PPAYMENT_DATA pPayment;
    COM_DATA *pData;
    try
    {
        pData = (COM_DATA*)txn_in.parray->pvData;

```

```

        pPayment = m_pTxn->BuffAddr_Payment();

        memcpy(pPayment, &pData->u.Payment,
sizeof(PAYMENT_DATA));

        VariantInit(txn_out);
        txn_out->vt = VT_SAFEARRAY;
        txn_out->parray = SafeArrayCreateVector(
VT_UI1,

txn_in.parray->rgsabound->cElements,

txn_in.parray->rgsabound->cElements);
        pData = (COM_DATA*)
txn_out->parray->pvData;

        m_pTxn->Payment(TRUE);

        memcpy( &pData->u.Payment, pPayment,
sizeof(PAYMENT_DATA));

        pData->retval = ERR_SUCCESS;
        pData->error = 0;
        return S_OK;
    }
    catch (CBaseErr *e)
    {
        // check for lost database connection; if yes,
component is toast
        if ( ((e->ErrorType() == ERR_TYPE_DBLIB)
&& (e->ErrorNum() == 10005)) ||
            ((e->ErrorType() ==
ERR_TYPE_ODBC) && (e->ErrorNum() == 10054)) )
            m_bCanBePooled = FALSE;

        pData->retval = e->ErrorType();
        pData->error = e->ErrorNum();
        delete e;
        return E_FAIL;
    }
    catch (...)
    {
        WriteMessageToEventLog(TEXT("Unhandled
exception."));

        pData->retval = ERR_TYPE_LOGIC;
        pData->error = 0;
        m_bCanBePooled = FALSE;
        return E_FAIL;
    }
}

HRESULT CTPCC_Common::StockLevel(VARIANT txn_in,
VARIANT* txn_out)
{
    PSTOCK_LEVEL_DATA    pStockLevel;
    COM_DATA              *pData;

    try
    {
        pData = (COM_DATA*)txn_in.parray->pvData;
        pStockLevel =
m_pTxn->BuffAddr_StockLevel();

        memcpy(pStockLevel, &pData->u.StockLevel,
sizeof(STOCK_LEVEL_DATA));

        VariantInit(txn_out);

```

```

        txn_out->vt = VT_SAFEARRAY;
        txn_out->parray = SafeArrayCreateVector(
VT_UI1,

txn_in.parray->rgsabound->cElements,

txn_in.parray->rgsabound->cElements);
        pData =
(COM_DATA*)txn_out->parray->pvData;

        m_pTxn->StockLevel();

        memcpy( &pData->u.StockLevel, pStockLevel,
sizeof(STOCK_LEVEL_DATA));

        pData->retval = ERR_SUCCESS;
        pData->error = 0;
        return S_OK;
    }
    catch (CBaseErr *e)
    {
        // check for lost database connection; if yes,
component is toast
        if ( ((e->ErrorType() == ERR_TYPE_DBLIB)
&& (e->ErrorNum() == 10005)) ||
            ((e->ErrorType() ==
ERR_TYPE_ODBC) && (e->ErrorNum() == 10054)) )
            m_bCanBePooled = FALSE;

        pData->retval = e->ErrorType();
        pData->error = e->ErrorNum();
        delete e;
        return E_FAIL;
    }
    catch (...)
    {
        WriteMessageToEventLog(TEXT("Unhandled
exception."));

        pData->retval = ERR_TYPE_LOGIC;
        pData->error = 0;
        m_bCanBePooled = FALSE;
        return E_FAIL;
    }
}

HRESULT CTPCC_Common::OrderStatus(VARIANT txn_in,
VARIANT* txn_out)
{
    PORDER_STATUS_DATA    pOrderStatus;
    COM_DATA              *pData;
    try
    {
        pData = (COM_DATA*)txn_in.parray->pvData;
        pOrderStatus =
m_pTxn->BuffAddr_OrderStatus();

        memcpy(pOrderStatus, &pData->u.OrderStatus,
sizeof(ORDER_STATUS_DATA));

        VariantInit(txn_out);
        txn_out->vt = VT_SAFEARRAY;
        txn_out->parray = SafeArrayCreateVector(
VT_UI1,

txn_in.parray->rgsabound->cElements,

```



```

txn_in.parray->rgsabound->cElements);
    pData =
(COM_DATA*)txn_out->parray->pvData;

    m_pTxn->OrderStatus();

    memcpy( &pData->u.OrderStatus, pOrderStatus,
sizeof(ORDER_STATUS_DATA));

    pData->retval = ERR_SUCCESS;
    pData->error = 0;
    return S_OK;
}
catch (CBaseErr *e)
{
    // check for lost database connection; if yes,
component is toast
    if ( ((e->ErrorType() == ERR_TYPE_DBLIB)
&& (e->ErrorNum() == 10005)) ||
        ((e->ErrorType() ==
ERR_TYPE_ODBC) && (e->ErrorNum() == 10054)) )
        m_bCanBePooled = FALSE;

    pData->retval = e->ErrorType();
    pData->error = e->ErrorNum();
    delete e;
    return E_FAIL;
}
catch (...)
{
    WriteMessageToEventLog(TEXT("Unhandled
exception."));

    pData->retval = ERR_TYPE_LOGIC;
    pData->error = 0;
    m_bCanBePooled = FALSE;
    return E_FAIL;
}
}

```

## **tpcc\_com\_all.def**

: tpcc\_com\_all.def : Declares the module parameters.

```

LIBRARY "tpcc_com_all.dll"

EXPORTS
    DllCanUnloadNow @1 PRIVATE
    DllGetClassObject @2 PRIVATE
    DllRegisterServer @3 PRIVATE
    DllUnregisterServer @4 PRIVATE

```

## **tpcc\_com\_all.h**

/\* this ALWAYS GENERATED file contains the definitions for the interfaces \*/

```

/* File created by MIDL compiler version 5.01.0164 */
/* at Fri Jan 19 11:14:55 2001
*/
/* Compiler settings for .\src\tpcc_com_all.idl:
    Oicf (OptLev=i2), W1, Zp8, env=Win32, ms_ext, c_ext
    error checks: allocation ref bounds_check enum stub_data
*/
//@@@MIDL_FILE_HEADING( )

```

/\* verify that the <rpcndr.h> version is high enough to compile this file\*/

```

#ifndef __REQUIRED_RPCNDR_H_VERSION__
#define __REQUIRED_RPCNDR_H_VERSION__ 440
#endif

```

```

#include "rpc.h"
#include "rpcndr.h"

```

```

#ifndef __tpcc_com_all_h__
#define __tpcc_com_all_h__

```

```

#ifdef __cplusplus
extern "C" {
#endif

```

/\* Forward Declarations \*/

```

#ifndef __TPCC_FWD_DEFINED__
#define __TPCC_FWD_DEFINED__

```

```

#ifdef __cplusplus
typedef class TPCC TPCC;
#else
typedef struct TPCC TPCC;
#endif /* __cplusplus */

```

```

#endif /* __TPCC_FWD_DEFINED__ */

```

```

#ifndef __NewOrder_FWD_DEFINED__
#define __NewOrder_FWD_DEFINED__

```

```

#ifdef __cplusplus
typedef class NewOrder NewOrder;
#else
typedef struct NewOrder NewOrder;
#endif /* __cplusplus */

```

```

#endif /* __NewOrder_FWD_DEFINED__ */

```

```

#ifndef __OrderStatus_FWD_DEFINED__
#define __OrderStatus_FWD_DEFINED__

```

```

#ifdef __cplusplus
typedef class OrderStatus OrderStatus;
#else
typedef struct OrderStatus OrderStatus;
#endif /* __cplusplus */

```

```

#endif /* __OrderStatus_FWD_DEFINED__ */

```

```

#ifndef __Payment_FWD_DEFINED__
#define __Payment_FWD_DEFINED__

```

```

#ifdef __cplusplus
typedef class Payment Payment;
#else
typedef struct Payment Payment;
#endif /* __cplusplus */

```

```

#endif /* __Payment_FWD_DEFINED__ */

```

```

#ifdef __StockLevel_FWD_DEFINED__
#define __StockLevel_FWD_DEFINED__

#ifdef __cplusplus
typedef class StockLevel StockLevel;
#else
typedef struct StockLevel StockLevel;
#endif /* __cplusplus */

#ifdef __StockLevel_FWD_DEFINED__ */

/* header files for imported files */
#include "oidl.h"
#include "ocidl.h"
#include "tpcc_com_ps.h"

void __RPC_FAR * __RPC_USER MIDL_user_allocate(size_t);
void __RPC_USER MIDL_user_free( void __RPC_FAR *);

/* interface __MIDL_itf_tpcc_com_all_0000 */
/* [local] */

extern RPC_IF_HANDLE
__MIDL_itf_tpcc_com_all_0000_v0_0_c_ifspec;
extern RPC_IF_HANDLE
__MIDL_itf_tpcc_com_all_0000_v0_0_s_ifspec;

#ifdef __TPCCLib_LIBRARY_DEFINED__
#define __TPCCLib_LIBRARY_DEFINED__

/* library TPCCLib */
/* [helpstring][version][uuid] */

EXTERN_C const IID LIBID_TPCCLib;

EXTERN_C const CLSID CLSID_TPCC;

#ifdef __cplusplus
class
DECLSPEC_UUID("122A3128-2520-11D3-BA71-00C04FBFE08B")
TPCC;
#endif

EXTERN_C const CLSID CLSID_NewOrder;

#ifdef __cplusplus
class
DECLSPEC_UUID("975BAABF-84A7-11D2-BA47-00C04FBFE08B")
NewOrder;
#endif

EXTERN_C const CLSID CLSID_OrderStatus;

#ifdef __cplusplus

```

```

class
DECLSPEC_UUID("266836AD-A50D-11D2-BA4E-00C04FBFE08B")
OrderStatus;
#endif

EXTERN_C const CLSID CLSID_Payment;

#ifdef __cplusplus
class
DECLSPEC_UUID("CD02F7EF-A4FA-11D2-BA4E-00C04FBFE08B")
Payment;
#endif

EXTERN_C const CLSID CLSID_StockLevel;

#ifdef __cplusplus
class
DECLSPEC_UUID("2668369E-A50D-11D2-BA4E-00C04FBFE08B")
StockLevel;
#endif /* __TPCCLib_LIBRARY_DEFINED__ */

/* Additional Prototypes for ALL interfaces */
/* end of Additional Prototypes */

#ifdef __cplusplus
}
#endif

#endif

tpcc_com_all.idl

/* FILE: TPCC.IDL
* Microsoft TPC-C Kit Ver.
4.20.000
* Copyright Microsoft,
1999
* All Rights Reserved
* not yet audited
* PURPOSE: IDL source for TPCC.dll. This file is
processed by the MIDL tool to
* produce the type library
(TPCC.tlb) and marshalling code.
*
* Change history:
* 4.20.000 - first version
*/

interface TPCC;
interface NewOrder;
interface OrderStatus;
interface Payment;
interface StockLevel;

```

```

import "oidl.idl";
import "ocidl.idl";
import "..\tpcc_com_ps\src\tpcc_com_ps.idl";

[
    uuid(122A3117-2520-11D3-BA71-00C04FBFE08B),
    version(1.0),
    helpstring("TPC-C 1.0 Type Library")
]
library TPCCLib
{
    importlib("stdole32.tlb");
    importlib("stdole2.tlb");

    [
        uuid(122A3128-2520-11D3-BA71-00C04FBFE08B),
        helpstring("All Txns Class")
    ]
    coclass TPCC
    {
        [default] interface ITPCC;
    };

    [
        uuid(975BAABF-84A7-11D2-BA47-00C04FBFE08B),
        helpstring("NewOrder Class")
    ]
    coclass NewOrder
    {
        [default] interface ITPCC;
    };

    [
        uuid(266836AD-A50D-11D2-BA4E-00C04FBFE08B),
        helpstring("OrderStatus Class")
    ]
    coclass OrderStatus
    {
        [default] interface ITPCC;
    };

    [
        uuid(CD02F7EF-A4FA-11D2-BA4E-00C04FBFE08B),
        helpstring("Payment Class")
    ]
    coclass Payment
    {
        [default] interface ITPCC;
    };

    [
        uuid(2668369E-A50D-11D2-BA4E-00C04FBFE08B),
        helpstring("StockLevel Class")
    ]
    coclass StockLevel
    {
        [default] interface ITPCC;
    };
}

```

```

};
};

tpcc_com_all.rc

//Microsoft Developer Studio generated resource script.
//
#include "resource.h"

#define APSTUDIO_READONLY_SYMBOLS
//
// Generated from the TEXTINCLUDE 2 resource.
//
#include "winres.h"

//
// English (U.S.) resources
//
#ifdef AFX_RESOURCE_DLL || defined(AFX_TARG_ENU)
#ifdef _WIN32
LANGUAGE LANG_ENGLISH, SUBLANG_ENGLISH_US
#pragma code_page(1252)
#endif // _WIN32

#ifdef APSTUDIO_INVOKED
//
// TEXTINCLUDE
//
1 TEXTINCLUDE DISCARDABLE
BEGIN
    "resource.h\0"
END

2 TEXTINCLUDE DISCARDABLE
BEGIN
    "#include ""winres.h""\r\n"
    "\0"
END

3 TEXTINCLUDE DISCARDABLE
BEGIN
    "1 TYPELIB ""tpcc_com_all.tlb""\r\n"
    "\0"
END

#endif // APSTUDIO_INVOKED

#ifdef _MAC
//
// Version
//
VS_VERSION_INFO VERSIONINFO
FILEVERSION 1,0,0,1
PRODUCTVERSION 1,0,0,1
FILEFLAGSMASK 0x3fL
#ifdef _DEBUG

```

```

FILEFLAGS0x1L
#else
FILEFLAGS0x0L
#endif
FILEOS 0x4L
FILETYPE 0x2L
FILESUBTYPE 0x0L
BEGIN
    BLOCK "StringFileInfo"
    BEGIN
        BLOCK "040904B0"
        BEGIN
            VALUE "CompanyName", "\0"
            VALUE "FileDescription", "tpcc_com_all Module\0"
            VALUE "FileVersion", "1, 0, 0, 1\0"
            VALUE "InternalName", "TPCCNEWORDER\0"
            VALUE "LegalCopyright", "Copyright 1997\0"
            VALUE "OriginalFilename", "tpcc_com_all.DLL\0"
            VALUE "ProductName", "tpcc_com_all Module\0"
            VALUE "ProductVersion", "1, 0, 0, 1\0"
            VALUE "OLESelfRegister", "\0"
        END
    END
    BLOCK "VarFileInfo"
    BEGIN
        VALUE "Translation", 0x409, 1200
    END
END

#endif // !_MAC

////////////////////////////////////
//
// REGISTRY
//

IDR_TPCC          REGISTRY DISCARDABLE
"tpcc_com_all.rgs"
IDR_NEWORDER     REGISTRY DISCARDABLE
"tpcc_com_no.rgs"
IDR_ORDERSTATUS  REGISTRY DISCARDABLE
"tpcc_com_os.rgs"
IDR_PAYMENT      REGISTRY DISCARDABLE
"tpcc_com_pay.rgs"
IDR_STOCKLEVEL   REGISTRY DISCARDABLE
"tpcc_com_sl.rgs"

////////////////////////////////////
//
// String Table
//

STRINGTABLEDISCARDABLE
BEGIN
    IDS_PROJNAME     "tpcc_com_all"
END

#endif // English (U.S.) resources

////////////////////////////////////

#ifndef APSTUDIO_INVOKED
////////////////////////////////////
//
// Generated from the TEXTINCLUDE3 resource.
//

```

```

1 TYPELIB "tpcc_com_all.tlb"

////////////////////////////////////
#endif // not APSTUDIO_INVOKED

tpcc_com_all.rgs

HKCR
{
    TPCC.AllTxns.1 = s 'All Txns Class'
    {
        CLSID = s
'122A3128-2520-11D3-BA71-00C04FBFE08B}'
    }
    TPCC.AllTxns = s 'TPCC Class'
    {
        CurVer = s 'TPCC.AllTxns.1'
    }
    NoRemove CLSID
    {
        ForceRemove
'122A3128-2520-11D3-BA71-00C04FBFE08B' = s 'TPCC Class'
    }
    ProgID = s 'TPCC.AllTxns.1'
    VersionIndependentProgID = s
'TPCC.AllTxns'
    InprocServer32 = s '%MODULE%'
    {
        val ThreadingModel = s
'Both'
    }
}

tpcc_com_all.i.c

/* this file contains the actual definitions of */
/* the IIDs and CLSIDs */

/* link this file in with the server and any clients */

/* File created by MIDL compiler version 5.01.0164 */
/* at Fri Jan 19 11:14:55 2001
*/
/* Compiler settings for .\src\tpcc_com_all.idl:
    Oicf (OptLev=i2), W1, Zp8, env=Win32, ms_ext, c_ext
    error checks: allocation ref bounds_check enum stub_data
*/
//@@@MIDL_FILE_HEADING( )
#ifdef __cplusplus
extern "C"{
#endif

#ifdef __IID_DEFINED__
#define __IID_DEFINED__

typedef struct _IID
{
    unsigned long x;
    unsigned short s1;
    unsigned short s2;
    unsigned char c[8];

```

```

} IID;

#endif // __IID_DEFINED__

#ifndef CLSID_DEFINED
#define CLSID_DEFINED
typedef IID CLSID;
#endif // CLSID_DEFINED

const IID LIBID_TPCCLib =
{0x122A3117,0x2520,0x11D3,{0xBA,0x71,0x00,0xC0,0x4F,0xBF,0xE0,0x8B}};

const CLSID CLSID_TPCC =
{0x122A3128,0x2520,0x11D3,{0xBA,0x71,0x00,0xC0,0x4F,0xBF,0xE0,0x8B}};

const CLSID CLSID_NewOrder =
{0x975BAABF,0x84A7,0x11D2,{0xBA,0x47,0x00,0xC0,0x4F,0xBF,0xE0,0x8B}};

const CLSID CLSID_OrderStatus =
{0x266836AD,0xA50D,0x11D2,{0xBA,0x4E,0x00,0xC0,0x4F,0xBF,0xE0,0x8B}};

const CLSID CLSID_Payment =
{0xCD02F7EF,0xA4FA,0x11D2,{0xBA,0x4E,0x00,0xC0,0x4F,0xBF,0xE0,0x8B}};

const CLSID CLSID_StockLevel =
{0x2668369E,0xA50D,0x11D2,{0xBA,0x4E,0x00,0xC0,0x4F,0xBF,0xE0,0x8B}};

#ifdef __cplusplus
}
#endif

```

### **tpcc\_com\_no.rgs**

```

HKCR
{
    TPCC.NewOrder.1 = s 'NewOrder Class'
    {
        CLSID = s
        '{975BAABF-84A7-11D2-BA47-00C04FBFE08B}'
    }
    TPCC.NewOrder = s 'NewOrder Class'
    {
        CurVer = s 'TPCC.NewOrder.1'
    }
    NoRemove CLSID
    {
        ForceRemove
        {975BAABF-84A7-11D2-BA47-00C04FBFE08B} = s 'NewOrder
        Class'
    }
    ProgID = s 'TPCC.NewOrder.1'
    VersionIndependentProgID = s
    'TPCC.NewOrder'
    InprocServer32 = s '%MODULE%'
}

```

```

{
    val ThreadingModel = s
}
}
}
}
}

tpcc_com_os.rgs

HKCR
{
    TPCC.OrderStatus.1 = s 'OrderStatus Class'
    {
        CLSID = s
        '{266836AD-A50D-11D2-BA4E-00C04FBFE08B}'
    }
    TPCC.OrderStatus = s 'OrderStatus Class'
    {
        CurVer = s 'TPCC.OrderStatus.1'
    }
    NoRemove CLSID
    {
        ForceRemove
        {266836AD-A50D-11D2-BA4E-00C04FBFE08B} = s 'OrderStatus
        Class'
    }
    ProgID = s 'TPCC.OrderStatus.1'
    VersionIndependentProgID = s
    'TPCC.OrderStatus'
    InprocServer32 = s '%MODULE%'
    {
        val ThreadingModel = s
    }
}
}
}
}
}

```

### **tpcc\_com\_pay.rgs**

```

HKCR
{
    TPCC.Payment.1 = s 'Payment Class'
    {
        CLSID = s
        '{CD02F7EF-A4FA-11D2-BA4E-00C04FBFE08B}'
    }
    TPCC.Payment = s 'Payment Class'
    {
        CurVer = s 'TPCC.Payment.1'
    }
    NoRemove CLSID
    {
        ForceRemove
        {CD02F7EF-A4FA-11D2-BA4E-00C04FBFE08B} = s 'Payment
        Class'
    }
    ProgID = s 'TPCC.Payment.1'
    VersionIndependentProgID = s
    'TPCC.Payment'
    InprocServer32 = s '%MODULE%'
    {
        val ThreadingModel = s
    }
}
}
}
}
}

```

```

    }
}

tpcc_com_ps.def

```

```
LIBRARY "tpcc_com_ps"
```

```
DESCRIPTION 'Proxy/Stub DLL'
```

```
EXPORTS
```

```

    DllGetClassObject    @1    PRIVATE
    DllCanUnloadNow     @2    PRIVATE
    GetProxyDllInfo     @3    PRIVATE
    DllRegisterServer   @4    PRIVATE
    DllUnregisterServer @5    PRIVATE

```

## tpcc\_com\_ps.h

```
/* this ALWAYS GENERATED file contains the definitions for the
interfaces */
```

```
/* File created by MIDL compiler version 5.01.0164 */
```

```
/* at Fri Jan 19 11:14:46 2001
```

```
*/
```

```
/* Compiler settings for .\src\tpcc_com_ps.idl:
```

```

    Oicf (OptLev=i2), W1, Zp8, env=Win32, ms_ext, c_ext
    error checks: allocation ref bounds_check enum stub_data
*/
```

```
//@@MIDL_FILE_HEADING( )
```

```
/* verify that the <rpcndr.h> version is high enough to compile this
file*/
```

```

#ifndef __REQUIRED_RPCNDR_H_VERSION__
#define __REQUIRED_RPCNDR_H_VERSION__ 440
#endif

```

```

#include "rpc.h"
#include "rpcndr.h"

```

```

#ifndef __RPCNDR_H_VERSION__
#error this stub requires an updated version of <rpcndr.h>
#endif // __RPCNDR_H_VERSION__

```

```

#ifndef COM_NO_WINDOWS_H
#include "windows.h"
#include "ole2.h"
#endif /*COM_NO_WINDOWS_H*/

```

```

#ifndef __tpcc_com_ps_h__
#define __tpcc_com_ps_h__

```

```

#ifdef __cplusplus
extern "C"{
#endif

```

```
/* Forward Declarations */
```

```

#ifndef __ITPCC_FWD_DEFINED__
#define __ITPCC_FWD_DEFINED__
typedef interface ITPCC ITPCC;
#endif /* __ITPCC_FWD_DEFINED__ */

```

```
/* header files for imported files */
```

```

#include "oidl.h"
#include "ocidl.h"

```

```

void __RPC_FAR * __RPC_USER MIDL_user_allocate(size_t);
void __RPC_USER MIDL_user_free( void __RPC_FAR * );

```

```

/* interface __MIDL_itf_tpcc_com_ps_0000 */
/* [local] */

```

```

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;

```

```

#ifndef __ITPCC_INTERFACE_DEFINED__
#define __ITPCC_INTERFACE_DEFINED__

```

```

/* interface ITPCC */
/* [unique][helpstring][uuid][oleautomation][object] */

```

```
EXTERN_C const IID IID_ITPCC;
```

```
#if defined(__cplusplus) && !defined(CINTERFACE)
```

```

MIDL_INTERFACE("FEEE6AA2-84B1-11d2-BA47-00C04FBFE08B")

```

```
ITPCC : public IUnknown
```

```
{
```

```
public:
```

```

    virtual HRESULT __stdcall NewOrder(
        /* [in] */ VARIANT txn_in,
        /* [out] */ VARIANT __RPC_FAR *txn_out) = 0;

```

```

    virtual HRESULT __stdcall Payment(
        /* [in] */ VARIANT txn_in,
        /* [out] */ VARIANT __RPC_FAR *txn_out) = 0;

```

```

    virtual HRESULT __stdcall Delivery(
        /* [in] */ VARIANT txn_in) = 0;

```

```

    virtual HRESULT __stdcall StockLevel(
        /* [in] */ VARIANT txn_in,
        /* [out] */ VARIANT __RPC_FAR *txn_out) = 0;

```

```

    virtual HRESULT __stdcall OrderStatus(
        /* [in] */ VARIANT txn_in,
        /* [out] */ VARIANT __RPC_FAR *txn_out) = 0;

```

```

    virtual HRESULT __stdcall CallSetComplete( void) = 0;

```

```
};
```

```
#else /* C style interface */
```

```
typedef struct ITPCCVtbl
```

```
{
```

```

    BEGIN_INTERFACE

```

```

    HRESULT ( STDMETHODCALLTYPE __RPC_FAR
    *QueryInterface)(

```

```

ITPCC __RPC_FAR * This,
/* [in] */ REFIID riid,
/* [iid_is][out] */ void __RPC_FAR * __RPC_FAR
*ppvObject);

ULONG ( STDMETHODCALLTYPE __RPC_FAR *AddRef)(
ITPCC __RPC_FAR * This);

ULONG ( STDMETHODCALLTYPE __RPC_FAR *Release)(
ITPCC __RPC_FAR * This);

HRESULT ( STDMETHODCALLTYPE __RPC_FAR *NewOrder)(
ITPCC __RPC_FAR * This,
/* [in] */ VARIANT txn_in,
/* [out] */ VARIANT __RPC_FAR *txn_out);

HRESULT ( STDMETHODCALLTYPE __RPC_FAR *Payment)(
ITPCC __RPC_FAR * This,
/* [in] */ VARIANT txn_in,
/* [out] */ VARIANT __RPC_FAR *txn_out);

HRESULT ( STDMETHODCALLTYPE __RPC_FAR *Delivery)(
ITPCC __RPC_FAR * This,
/* [in] */ VARIANT txn_in);

HRESULT ( STDMETHODCALLTYPE __RPC_FAR *StockLevel)(
ITPCC __RPC_FAR * This,
/* [in] */ VARIANT txn_in,
/* [out] */ VARIANT __RPC_FAR *txn_out);

HRESULT ( STDMETHODCALLTYPE __RPC_FAR *OrderStatus)(
ITPCC __RPC_FAR * This,
/* [in] */ VARIANT txn_in,
/* [out] */ VARIANT __RPC_FAR *txn_out);

HRESULT ( STDMETHODCALLTYPE __RPC_FAR *CallSetComplete)(
ITPCC __RPC_FAR * This);

END_INTERFACE
} ITPCCVtbl;

interface ITPCC
{
CONST_VTBL struct ITPCCVtbl __RPC_FAR *lpVtbl;
};

#ifdef COBJMACROS

#define ITPCC_QueryInterface(This,riid,ppvObject) \
(This)->lpVtbl->QueryInterface(This,riid,ppvObject)

#define ITPCC_AddRef(This) \
(This)->lpVtbl->AddRef(This)

#define ITPCC_Release(This) \
(This)->lpVtbl->Release(This)

#define ITPCC_NewOrder(This,txn_in,txn_out) \
(This)->lpVtbl->NewOrder(This,txn_in,txn_out)

#define ITPCC_Payment(This,txn_in,txn_out) \
(This)->lpVtbl->Payment(This,txn_in,txn_out)

#define ITPCC_Delivery(This,txn_in) \

```

```

(This)->lpVtbl->Delivery(This,txn_in)

#define ITPCC_StockLevel(This,txn_in,txn_out) \
(This)->lpVtbl->StockLevel(This,txn_in,txn_out)

#define ITPCC_OrderStatus(This,txn_in,txn_out) \
(This)->lpVtbl->OrderStatus(This,txn_in,txn_out)

#define ITPCC_CallSetComplete(This) \
(This)->lpVtbl->CallSetComplete(This)

#endif /* COBJMACROS */

#endif /* C style interface */

HRESULT STDMETHODCALLTYPE 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,
IRpcChannelBuffer *_pRpcChannelBuffer,
PRPC_MESSAGE _pRpcMessage,
DWORD *_pdwStubPhase);

HRESULT STDMETHODCALLTYPE 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 STDMETHODCALLTYPE ITPCC_Delivery_Proxy(
ITPCC __RPC_FAR * This,
/* [in] */ VARIANT txn_in);

void __RPC_STUB ITPCC_Delivery_Stub(
IRpcStubBuffer *This,
IRpcChannelBuffer *_pRpcChannelBuffer,
PRPC_MESSAGE _pRpcMessage,
DWORD *_pdwStubPhase);

HRESULT STDMETHODCALLTYPE 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,
    IRpcChannelBuffer *_pRpcChannelBuffer,
    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);

#ifdef __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 * );
void __RPC_USER VARIANT_UserFree( unsigned
    long __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 "oidl.idl";
import "ocidl.idl";

[
    object,
    oleautomation,
    uuid(FEEE6AA2-84B1-11d2-BA47-00C04FBFE08B),
    helpstring("ITPCC Interface"),
    pointer_default(unique)
]
interface ITPCC : IUnknown
{
    HRESULT __stdcall NewOrder
    (
    [in]
    VARIANT txn_in,
    [out] VARIANT *txn_out
    );

    HRESULT __stdcall Payment
    (
    [in]
    VARIANT txn_in,
    [out] VARIANT *txn_out
    );

    HRESULT __stdcall Delivery
    (
    [in]
    VARIANT txn_in
    );

    HRESULT __stdcall StockLevel
    (
    [in]
    VARIANT txn_in,
    [out] VARIANT *txn_out
    );

    HRESULT __stdcall OrderStatus
    (
    [in]
    VARIANT txn_in,
    [out] VARIANT *txn_out
    );

    HRESULT __stdcall CallSetComplete
    (
    );

}; // interface ITPCC

tpcc_com_ps.i.c

/* this file contains the actual definitions of */

```



```

/* the IIDs and CLSIDs */

/* link this file in with the server and any clients */

/* File created by MIDL compiler version 5.01.0164 */
/* at Fri Jan 19 11:14:46 2001
*/
/* Compiler settings for .\src\tpcc_com_ps.idl:
Oicf (OptLev=i2), W1, Zp8, env=Win32, ms_ext, c_ext
error checks: allocation ref bounds_check enum stub_data
*/
//@@@MIDL_FILE_HEADING( )
#ifdef __cplusplus
extern "C"{
#endif

#ifdef __IID_DEFINED__
#define __IID_DEFINED__

typedef struct _IID
{
    unsigned long x;
    unsigned short s1;
    unsigned short s2;
    unsigned char c[8];
} IID;

#endif // __IID_DEFINED__

#ifdef CLSID_DEFINED
#define CLSID_DEFINED
typedef IID CLSID;
#endif // CLSID_DEFINED

const IID IID_ITPCC=
{0xFEEE6AA2,0x84B1,0x11d2,{0xBA,0x47,0x00,0xC0,0x4F,0xBF,0xE0,0x8B}};

#ifdef __cplusplus
}
#endif

tpcc_com_ps_p.c

/* this ALWAYS GENERATED file contains the proxy stub code */

/* File created by MIDL compiler version 5.01.0164 */
/* at Fri Jan 19 11:14:46 2001
*/
/* Compiler settings for .\src\tpcc_com_ps.idl:
Oicf (OptLev=i2), W1, Zp8, env=Win32, ms_ext, c_ext
error checks: allocation ref bounds_check enum stub_data
*/
//@@@MIDL_FILE_HEADING( )

#define USE_STUBLESS_PROXY

/* verify that the <rpcproxy.h> version is high enough to compile this
file*/
#ifdef __REDQ_RPCPROXY_H_VERSION__
#define __REQUIRED_RPCPROXY_H_VERSION__ 440

```

```

#endif

#include "rpcproxy.h"
#ifdef __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 997
#define PROC_FORMAT_STRING_SIZE 187

typedef struct _MIDL_TYPE_FORMAT_STRING
{
    short Pad;
    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,
GUID={0x00000000,0x0000,0x0000,{0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00}} */

/* Object interface: IUnknown, ver. 0.0,
GUID={0x00000000,0x0000,0x0000,{0xC0,0x00,0x00,0x00,0x00,0x00,0x00,0x00}} */

/* Object interface: ITPCC, ver. 0.0,
GUID={0xFEEE6AA2,0x84B1,0x11d2,{0xBA,0x47,0x00,0xC0,0x4F,0xBF,0xE0,0x8B}} */

extern const MIDL_STUB_DESC Object_StubDesc;

extern const MIDL_SERVER_INFO ITPCC_ServerInfo;

#pragma code_seg(".orpc")
extern const USER_MARSHAL_ROUTINE_QUADRUPLE
UserMarshalRoutines[1];

static const MIDL_STUB_DESC Object_StubDesc =
{
    0,
    NdrOleAllocate,
    NdrOleFree,
    0,
    0,
    0,
    0,

```

```

0,
0,
__MIDL_TypeFormatString.Format,
1, /* -error bounds_check flag */
0x20000, /* Ndr library version */
0,
0x50100a4, /* MIDL Version 5.1.164 */
0,
UserMarshalRoutines,
0, /* notify & notify_flag routine table */
1, /* Flags */
0, /* Reserved3 */
0, /* Reserved4 */
0 /* Reserved5 */
};

static const unsigned short ITPCC_FormatStringOffsetTable[] =
{
0,
34,
68,
96,
130,
164
};

static const MIDL_SERVER_INFO ITPCC_ServerInfo =
{
&Object_StubDesc,
0,
__MIDL_ProcFormatString.Format,
&ITPCC_FormatStringOffsetTable[-3],
0,
0,
0,
0
};

static const MIDL_STUBLESS_PROXY_INFO ITPCC_ProxyInfo =
{
&Object_StubDesc,
__MIDL_ProcFormatString.Format,
&ITPCC_FormatStringOffsetTable[-3],
0,
0,
0
};

CINTERFACE_PROXY_VTABLE(9)_ITPCCProxyVtbl =
{
&ITPCC_ProxyInfo,
&IID_ITPCC,
IUnknown_QueryInterface_Proxy,
IUnknown_AddRef_Proxy,
IUnknown_Release_Proxy,
(void *)-1 /* ITPCC::NewOrder */,
(void *)-1 /* ITPCC::Payment */,
(void *)-1 /* ITPCC::Delivery */,
(void *)-1 /* ITPCC::StockLevel */,
(void *)-1 /* ITPCC::OrderStatus */,
(void *)-1 /* ITPCC::CallSetComplete */
};

const CInterfaceStubVtbl_ITPCCStubVtbl =
{
&IID_ITPCC,
&ITPCC_ServerInfo,
9,
0, /* pure interpreted */
CStdStubBuffer_METHODS
};

#pragma data_seg(".rdata")

static const USER_MARSHAL_ROUTINE_QUADRUPLE
UserMarshalRoutines[1] =
{
{
VARIANT_UserSize
,VARIANT_UserMarshal
,VARIANT_UserUnmarshal
,VARIANT_UserFree
}
};

#if !defined(__RPC_WIN32__)
#error Invalid build platform for this stub.
#endif

#if !(TARGET_IS_NT40_OR_LATER)
#error You need a Windows NT 4.0 or later to run this stub because it
uses these features:
#error -Oif or -Oicf, [wire_marshall] or [user_marshall] attribute, more
than 32 methods in the interface.
#error However, your C/C++ compilation flags indicate you intend to
run this app on earlier systems.
#error This app will die there with the
RPC_X_WRONG_STUB_VERSION error.
#endif

static const MIDL_PROC_FORMAT_STRING
__MIDL_ProcFormatString =
{
0,
{
/* Procedure NewOrder */
FC_AUTO_HANDLE /* 0x33, /*
0x6c, /* Old Flags:
object, Oi2 */
/* 2 */ NdrFcLong( 0x0 ), /* 0 */
/* 6 */ NdrFcShort( 0x3 ), /* 3 */
#ifdef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 8 */ NdrFcShort( 0x1c ), /* x86 Stack size/offset = 28 */
#else
NdrFcShort( 0x20 ), /* MIPS &
PPC Stack size/offset = 32 */
#endif
#else
NdrFcShort( 0x28 ), /* Alpha Stack
size/offset = 40 */
#endif
/* 10 */ NdrFcShort( 0x0 ), /* 0 */
/* 12 */ NdrFcShort( 0x8 ), /* 8 */
/* 14 */ 0x7, /* Oi2 Flags: srv must size, clt must
size, has return, */
0x3, /* 3 */
/* Parameter txn_in */
}
}
};

```

```

/* 16 */  NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by
val, */
#ifdef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 18 */  NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
#else
NdrFcShort( 0x8 ), /* MIPS &
PPC Stack size/offset = 8 */
#endif
#else
NdrFcShort( 0x8 ), /* Alpha Stack
size/offset = 8 */
#endif
/* 20 */  NdrFcShort( 0x3c8 ), /* Type Offset=968 */

/* Parameter txn_out */

/* 22 */  NdrFcShort( 0x4113 ), /* Flags: must size, must
free, out, simple ref, srv alloc size=16 */
#ifdef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 24 */  NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
#else
NdrFcShort( 0x18 ), /* MIPS &
PPC Stack size/offset = 24 */
#endif
#else
NdrFcShort( 0x18 ), /* Alpha Stack
size/offset = 24 */
#endif
/* 26 */  NdrFcShort( 0x3da ), /* Type Offset=986 */

/* Return value */

/* 28 */  NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifdef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 30 */  NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
#else
NdrFcShort( 0x1c ), /* MIPS &
PPC Stack size/offset = 28 */
#endif
#else
NdrFcShort( 0x20 ), /* Alpha Stack
size/offset = 32 */
#endif
/* 32 */  0x8, /* FC_LONG */
0x0, /* 0 */

/* Procedure Payment */

/* 34 */  0x33, /* FC_AUTO_HANDLE */
0x6c, /* Old Flags:
object, Oi2 */
/* 36 */  NdrFcLong( 0x0 ), /* 0 */
/* 40 */  NdrFcShort( 0x4 ), /* 4 */
#ifdef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 42 */  NdrFcShort( 0x1c ), /* x86 Stack size/offset = 28 */
#else
NdrFcShort( 0x20 ), /* MIPS &
PPC Stack size/offset = 32 */
#endif
#else
NdrFcShort( 0x28 ), /* Alpha Stack
size/offset = 40 */
#endif

/* 44 */  NdrFcShort( 0x0 ), /* 0 */
/* 46 */  NdrFcShort( 0x8 ), /* 8 */
/* 48 */  0x7, /* Oi2 Flags: srv must size, clt must
size, has return, */
0x3, /* 3 */

/* Parameter txn_in */

/* 50 */  NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by
val, */
#ifdef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 52 */  NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
#else
NdrFcShort( 0x8 ), /* MIPS &
PPC Stack size/offset = 8 */
#endif
#else
NdrFcShort( 0x8 ), /* Alpha Stack
size/offset = 8 */
#endif
/* 54 */  NdrFcShort( 0x3c8 ), /* Type Offset=968 */

/* Parameter txn_out */

/* 56 */  NdrFcShort( 0x4113 ), /* Flags: must size, must
free, out, simple ref, srv alloc size=16 */
#ifdef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 58 */  NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
#else
NdrFcShort( 0x18 ), /* MIPS &
PPC Stack size/offset = 24 */
#endif
#else
NdrFcShort( 0x18 ), /* Alpha Stack
size/offset = 24 */
#endif
/* 60 */  NdrFcShort( 0x3da ), /* Type Offset=986 */

/* Return value */

/* 62 */  NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifdef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 64 */  NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
#else
NdrFcShort( 0x1c ), /* MIPS &
PPC Stack size/offset = 28 */
#endif
#else
NdrFcShort( 0x20 ), /* Alpha Stack
size/offset = 32 */
#endif
/* 66 */  0x8, /* FC_LONG */
0x0, /* 0 */

/* Procedure Delivery */

/* 68 */  0x33, /* FC_AUTO_HANDLE */
0x6c, /* Old Flags:
object, Oi2 */
/* 70 */  NdrFcLong( 0x0 ), /* 0 */
/* 74 */  NdrFcShort( 0x5 ), /* 5 */
#ifdef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 76 */  NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
#endif
#endif

```

```

PPC Stack size/offset = 28 */
#endif
#else
    NdrFcShort( 0x1c ), /* MIPS &
/* 110 */ 0x7, /* Oi2 Flags: srv must size, clt must
size, has return, */
    0x3, /* 3 */

    /* Parameter txn_in */

/* 112 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by
val, */
#ifndef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 114 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
#else
    NdrFcShort( 0x8 ), /* MIPS &
PPC Stack size/offset = 8 */
#endif
#else
    NdrFcShort( 0x8 ), /* Alpha Stack
size/offset = 8 */
#endif
/* 116 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */

/* Parameter txn_out */

/* 118 */ NdrFcShort( 0x4113 ), /* Flags: must size, must
free, out, simple ref, srv alloc size=16 */
#ifndef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 120 */ NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
#else
    NdrFcShort( 0x18 ), /* MIPS &
PPC Stack size/offset = 24 */
#endif
#else
    NdrFcShort( 0x18 ), /* Alpha Stack
size/offset = 24 */
#endif
/* 122 */ NdrFcShort( 0x3da ), /* Type Offset=986 */

/* Return value */

/* 124 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifndef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 126 */ NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
#else
    NdrFcShort( 0x1c ), /* MIPS &
PPC Stack size/offset = 28 */
#endif
#else
    NdrFcShort( 0x20 ), /* Alpha Stack
size/offset = 32 */
#endif
/* 128 */ 0x8, /* FC_LONG */
0x0, /* 0 */

/* Procedure StockLevel */

/* 96 */ 0x33, /* FC_AUTO_HANDLE */
0x6c, /* Old Flags:
object, Oi2 */
/* 98 */ NdrFcLong( 0x0 ), /* 0 */
/* 102 */ NdrFcShort( 0x6 ), /* 6 */
#endif
#if !defined(_ALPHA_)
#if !defined(_MIPS_) && !defined(_PPC_)
/* 104 */ NdrFcShort( 0x1c ), /* x86 Stack size/offset = 28 */
#else
    NdrFcShort( 0x20 ), /* MIPS &
PPC Stack size/offset = 32 */
#endif
#else
    NdrFcShort( 0x28 ), /* Alpha Stack
size/offset = 40 */
#endif
/* 106 */ NdrFcShort( 0x0 ), /* 0 */
/* 108 */ NdrFcShort( 0x8 ), /* 8 */

```

```

#endif
#else
    NdrFcShort( 0x28 ), /* Alpha Stack
size/offset = 40 */
#endif
/* 140 */ NdrFcShort( 0x0 ), /* 0 */
/* 142 */ NdrFcShort( 0x8 ), /* 8 */
/* 144 */ 0x7, /* Oi2 Flags: srv must size, clt must
size, has return, */
    0x3, /* 3 */
    /* Parameter txn_in */
/* 146 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by
val, */
#ifndef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 148 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
#else
    NdrFcShort( 0x8 ), /* MIPS &
PPC Stack size/offset = 8 */
#endif
#else
    NdrFcShort( 0x8 ), /* Alpha Stack
size/offset = 8 */
#endif
/* 150 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */
    /* Parameter txn_out */
/* 152 */ NdrFcShort( 0x4113 ), /* Flags: must size, must
free, out, simple ref, srv alloc size=16 */
#ifndef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 154 */ NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
#else
    NdrFcShort( 0x18 ), /* MIPS &
PPC Stack size/offset = 24 */
#endif
#else
    NdrFcShort( 0x18 ), /* Alpha Stack
size/offset = 24 */
#endif
/* 156 */ NdrFcShort( 0x3da ), /* Type Offset=986 */
    /* Return value */
/* 158 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifndef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 160 */ NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
#else
    NdrFcShort( 0x1c ), /* MIPS &
PPC Stack size/offset = 28 */
#endif
#else
    NdrFcShort( 0x20 ), /* Alpha Stack
size/offset = 32 */
#endif
/* 162 */ 0x8, /* FC_LONG */
    0x0, /* 0 */
    /* Procedure CallSetComplete */
/* 164 */ 0x33, /* FC_AUTO_HANDLE */
    0x6c, /* Old Flags:
object, Oi2 */
/* 166 */ NdrFcLong( 0x0 ), /* 0 */
/* 170 */ NdrFcShort( 0x8 ), /* 8 */
#ifndef _ALPHA_
/* 172 */ NdrFcShort( 0x8 ), /* x86, MIPS, PPC Stack size/offset
= 8 */
#else
    NdrFcShort( 0x10 ), /* Alpha Stack
size/offset = 16 */
#endif
/* 174 */ NdrFcShort( 0x0 ), /* 0 */
/* 176 */ NdrFcShort( 0x8 ), /* 8 */
/* 178 */ 0x4, /* Oi2 Flags: has return, */
    0x1, /* 1 */
    /* Return value */
/* 180 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifndef _ALPHA_
/* 182 */ NdrFcShort( 0x4 ), /* x86, MIPS, PPC Stack size/offset
= 4 */
#else
    NdrFcShort( 0x8 ), /* Alpha Stack
size/offset = 8 */
#endif
/* 184 */ 0x8, /* FC_LONG */
    0x0, /* 0 */
    0x0
    }
};
static const MIDL_TYPE_FORMAT_STRING
__MIDL_TypeFormatString =
{
    0,
    {
        NdrFcShort( 0x0 ), /* 0 */
        /* 2 */
        0x12, 0x0, /* FC_UP */
        /* 4 */ NdrFcShort( 0x3b0 ), /* Offset= 944 (948) */
        /* 6 */
        0x2b, /*
FC_NON_ENCAPSULATED_UNION */
        0x9, /* FC_ULONG */
        /*
*/
        /* 8 */ 0x7, /* Corr desc: FC_USHORT */
        0x0, /* */
        /* 10 */ NdrFcShort( 0xffff ), /* -8 */
        /* 12 */ NdrFcShort( 0x2 ), /* Offset= 2 (14) */
        /* 14 */ NdrFcShort( 0x10 ), /* 16 */
        /* 16 */ NdrFcShort( 0x2b ), /* 43 */
        /* 18 */ NdrFcLong( 0x3 ), /* 3 */
        /* 22 */ NdrFcShort( 0x8008 ), /* Simple arm type:
FC_LONG */
        /* 24 */ NdrFcLong( 0x11 ), /* 17 */
        /* 28 */ NdrFcShort( 0x8001 ), /* Simple arm type:
FC_BYTE */
        /* 30 */ NdrFcLong( 0x2 ), /* 2 */
        /* 34 */ NdrFcShort( 0x8006 ), /* Simple arm type:
FC_SHORT */
        /* 36 */ NdrFcLong( 0x4 ), /* 4 */
        /* 40 */ NdrFcShort( 0x800a ), /* Simple arm type:
FC_FLOAT */
        /* 42 */ NdrFcLong( 0x5 ), /* 5 */
        /* 46 */ NdrFcShort( 0x800c ), /* Simple arm type:
FC_DOUBLE */
        /* 48 */ NdrFcLong( 0xb ), /* 11 */
        /* 52 */ NdrFcShort( 0x8006 ), /* Simple arm type:
FC_SHORT */
    }
}

```

```

/* 54 */ NdrFcLong( 0xa ), /* 10 */
/* 58 */ NdrFcShort( 0x8008 ), /* Simple arm type:
FC_LONG */
/* 60 */ NdrFcLong( 0x6 ), /* 6 */
/* 64 */ NdrFcShort( 0xd6 ), /* Offset= 214 (278) */
/* 66 */ NdrFcLong( 0x7 ), /* 7 */
/* 70 */ NdrFcShort( 0x800c ), /* Simple arm type:
FC_DOUBLE */
/* 72 */ NdrFcLong( 0x8 ), /* 8 */
/* 76 */ NdrFcShort( 0xd0 ), /* Offset= 208 (284) */
/* 78 */ NdrFcLong( 0xd ), /* 13 */
/* 82 */ NdrFcShort( 0xe2 ), /* Offset= 226 (308) */
/* 84 */ NdrFcLong( 0x9 ), /* 9 */
/* 88 */ NdrFcShort( 0xee ), /* Offset= 238 (326) */
/* 90 */ NdrFcLong( 0x2000 ), /* 8192 */
/* 94 */ NdrFcShort( 0xfa ), /* Offset= 250 (344) */
/* 96 */ NdrFcLong( 0x24 ), /* 36 */
/* 100 */ NdrFcShort( 0x308 ), /* Offset= 776 (876) */
/* 102 */ NdrFcLong( 0x4024 ), /* 16420 */
/* 106 */ NdrFcShort( 0x302 ), /* Offset= 770 (876) */
/* 108 */ NdrFcLong( 0x4011 ), /* 16401 */
/* 112 */ NdrFcShort( 0x300 ), /* Offset= 768 (880) */
/* 114 */ NdrFcLong( 0x4002 ), /* 16386 */
/* 118 */ NdrFcShort( 0x2fe ), /* Offset= 766 (884) */
/* 120 */ NdrFcLong( 0x4003 ), /* 16387 */
/* 124 */ NdrFcShort( 0x2fc ), /* Offset= 764 (888) */
/* 126 */ NdrFcLong( 0x4004 ), /* 16388 */
/* 130 */ NdrFcShort( 0x2fa ), /* Offset= 762 (892) */
/* 132 */ NdrFcLong( 0x4005 ), /* 16389 */
/* 136 */ NdrFcShort( 0x2f8 ), /* Offset= 760 (896) */
/* 138 */ NdrFcLong( 0x400b ), /* 16395 */
/* 142 */ NdrFcShort( 0x2e6 ), /* Offset= 742 (884) */
/* 144 */ NdrFcLong( 0x400a ), /* 16394 */
/* 148 */ NdrFcShort( 0x2e4 ), /* Offset= 740 (888) */
/* 150 */ NdrFcLong( 0x4006 ), /* 16390 */
/* 154 */ NdrFcShort( 0x2ea ), /* Offset= 746 (900) */
/* 156 */ NdrFcLong( 0x4007 ), /* 16391 */
/* 160 */ NdrFcShort( 0x2e0 ), /* Offset= 736 (896) */
/* 162 */ NdrFcLong( 0x4008 ), /* 16392 */
/* 166 */ NdrFcShort( 0x2e2 ), /* Offset= 738 (904) */
/* 168 */ NdrFcLong( 0x400d ), /* 16397 */
/* 172 */ NdrFcShort( 0x2e0 ), /* Offset= 736 (908) */
/* 174 */ NdrFcLong( 0x4009 ), /* 16393 */
/* 178 */ NdrFcShort( 0x2de ), /* Offset= 734 (912) */
/* 180 */ NdrFcLong( 0x6000 ), /* 24576 */
/* 184 */ NdrFcShort( 0x2dc ), /* Offset= 732 (916) */
/* 186 */ NdrFcLong( 0x400c ), /* 16396 */
/* 190 */ NdrFcShort( 0x2da ), /* Offset= 730 (920) */
/* 192 */ NdrFcLong( 0x10 ), /* 16 */
/* 196 */ NdrFcShort( 0x8002 ), /* Simple arm type:
FC_CHAR */
/* 198 */ NdrFcLong( 0x12 ), /* 18 */
/* 202 */ NdrFcShort( 0x8006 ), /* Simple arm type:
FC_SHORT */
/* 204 */ NdrFcLong( 0x13 ), /* 19 */
/* 208 */ NdrFcShort( 0x8008 ), /* Simple arm type:
FC_LONG */
/* 210 */ NdrFcLong( 0x16 ), /* 22 */
/* 214 */ NdrFcShort( 0x8008 ), /* Simple arm type:
FC_LONG */
/* 216 */ NdrFcLong( 0x17 ), /* 23 */
/* 220 */ NdrFcShort( 0x8008 ), /* Simple arm type:
FC_LONG */
/* 222 */ NdrFcLong( 0xe ), /* 14 */
/* 226 */ NdrFcShort( 0x2be ), /* Offset= 702 (928) */
/* 228 */ NdrFcLong( 0x400e ), /* 16398 */
/* 232 */ NdrFcShort( 0x2c4 ), /* Offset= 708 (940) */
/* 234 */ NdrFcLong( 0x4010 ), /* 16400 */
/* 238 */ NdrFcShort( 0x2c2 ), /* Offset= 706 (944) */
/* 240 */ NdrFcLong( 0x4012 ), /* 16402 */
/* 244 */ NdrFcShort( 0x280 ), /* Offset= 640 (884) */
/* 246 */ NdrFcLong( 0x4013 ), /* 16403 */
/* 250 */ NdrFcShort( 0x27e ), /* Offset= 638 (888) */
/* 252 */ NdrFcLong( 0x4016 ), /* 16406 */
/* 256 */ NdrFcShort( 0x278 ), /* Offset= 632 (888) */
/* 258 */ NdrFcLong( 0x4017 ), /* 16407 */
/* 262 */ NdrFcShort( 0x272 ), /* Offset= 626 (888) */
/* 264 */ NdrFcLong( 0x0 ), /* 0 */
/* 268 */ NdrFcShort( 0x0 ), /* Offset= 0 (268) */
/* 270 */ NdrFcLong( 0x1 ), /* 1 */
/* 274 */ NdrFcShort( 0x0 ), /* Offset= 0 (274) */
/* 276 */ NdrFcShort( 0xffffffff ), /* Offset= -1 (275) */
/* 278 */
0x15, /*
FC_STRUCT */
0x7, /* 7 */
/* 280 */ NdrFcShort( 0x8 ), /* 8 */
/* 282 */ 0xb, /* FC_HYPER */
0x5b, /* FC_END */
/* 284 */
0x12, 0x0, /* FC_UP */
/* 286 */ NdrFcShort( 0xc ), /* Offset= 12 (298) */
/* 288 */
0x1b, /*
FC_CARRAY */
0x1, /* 1 */
/* 290 */ NdrFcShort( 0x2 ), /* 2 */
/* 292 */ 0x9, /* Corr desc: FC_ULONG */
0x0, /*
/* 294 */ NdrFcShort( 0xffc ), /* -4 */
/* 296 */ 0x6, /* FC_SHORT */
0x5b, /* FC_END */
/* 298 */
0x17, /*
FC_CSTRUCT */
0x3, /* 3 */
/* 300 */ NdrFcShort( 0x8 ), /* 8 */
/* 302 */ NdrFcShort( 0xffffffff2 ), /* Offset= -14 (288) */
/* 304 */ 0x8, /* FC_LONG */
0x8, /* FC_LONG
*/
/* 306 */ 0x5c, /* FC_PAD */
0x5b, /* FC_END */
/* 308 */
0x2f, /* FC_IP */
0x5a, /*
FC_CONSTANT_IID */
/* 310 */ NdrFcLong( 0x0 ), /* 0 */
/* 314 */ NdrFcShort( 0x0 ), /* 0 */
/* 316 */ NdrFcShort( 0x0 ), /* 0 */
/* 318 */ 0xc0, /* 192 */
0x0, /* 0 */
/* 320 */ 0x0, /* 0 */
0x0, /* 0 */
/* 322 */ 0x0, /* 0 */
0x0, /* 0 */
/* 324 */ 0x0, /* 0 */
0x46, /* 70 */
/* 326 */
0x2f, /* FC_IP */
0x5a, /*
FC_CONSTANT_IID */
/* 328 */ NdrFcLong( 0x20400 ), /* 132096 */
/* 332 */ NdrFcShort( 0x0 ), /* 0 */
/* 334 */ NdrFcShort( 0x0 ), /* 0 */
/* 336 */ 0xc0, /* 192 */

```

/* 338 */	0x0,	/* 0 */	/*	0x8,	/* FC_LONG
/* 340 */	0x0,	/* 0 */	/* 448 */	0x5c,	/* FC_PAD */
/* 342 */	0x0,	/* 0 */	/* 450 */	0x5b,	/* FC_END */
/* 344 */	0x46,	/* 70 */	FC_PSTRUCT */	0x16,	/*
/* 346 */	NdrFcShort( 0x2 ),	/* FC_UP */	/* 452 */	0x3,	/* 3 */
/* 348 */	/* Offset= 2 (348) */	/*	/* 454 */	0x5c,	/* FC_PP */
/* 350 */	NdrFcShort( 0x1fc ),	/* FC_UP */	/* 456 */	0x4b,	/* FC_PAD */
/* 352 */	/* Offset= 508 (858) */	/*	FC_NO_REPEAT */	0x5c,	/*
FC_ENCAPSULATED_UNION */	0x2a,	/*	/* 458 */	0x5c,	/* FC_PAD */
/* 354 */	NdrFcShort( 0x18 ),	/* 24 */	/* 460 */	NdrFcShort( 0x4 ),	/* 4 */
/* 356 */	NdrFcShort( 0xa ),	/* 10 */	/* 462 */	NdrFcShort( 0x4 ),	/* 4 */
/* 358 */	NdrFcLong( 0x8 ),	/* 8 */	/* 464 */	0x11, 0x0, /* FC_RP */	/*
/* 362 */	NdrFcShort( 0x58 ),	/* Offset= 88 (450) */	/* 466 */	NdrFcShort( 0xfffffd4 ),	/* Offset= -44 (420) */
/* 364 */	NdrFcLong( 0xd ),	/* 13 */	/*	0x5b,	/* FC_END */
/* 368 */	NdrFcShort( 0x78 ),	/* Offset= 120 (488) */	/* 468 */	0x8,	/* FC_LONG
/* 370 */	NdrFcLong( 0x9 ),	/* 9 */	/*	/*	/*
/* 374 */	NdrFcShort( 0x94 ),	/* Offset= 148 (522) */	/* 470 */	0x8,	/* FC_LONG */
/* 376 */	NdrFcLong( 0xc ),	/* 12 */	/*	/*	/*
/* 380 */	NdrFcShort( 0xbc ),	/* Offset= 188 (568) */	/* 472 */	0x8,	/* FC_LONG */
/* 382 */	NdrFcLong( 0x24 ),	/* 36 */	/* 474 */	0x5b,	/* FC_END */
/* 386 */	NdrFcShort( 0x114 ),	/* Offset= 276 (662) */	/*	/*	/*
/* 388 */	NdrFcLong( 0x800d ),	/* 32781 */	FC_BOGUS_ARRAY */	0x21,	/*
/* 392 */	NdrFcShort( 0x130 ),	/* Offset= 304 (696) */	/* 476 */	0x3,	/* 3 */
/* 394 */	NdrFcLong( 0x10 ),	/* 16 */	/* 478 */	NdrFcShort( 0x0 ),	/* 0 */
/* 398 */	NdrFcShort( 0x148 ),	/* Offset= 328 (726) */	/* 482 */	0x19,	/* Corr desc: field pointer,
/* 400 */	NdrFcLong( 0x2 ),	/* 2 */	/* 484 */	FC_ULONG */	/*
/* 404 */	NdrFcShort( 0x160 ),	/* Offset= 352 (756) */	/* 486 */	0x0,	/* */
/* 406 */	NdrFcLong( 0x3 ),	/* 3 */	/* 488 */	NdrFcShort( 0x0 ),	/* 0 */
/* 410 */	NdrFcShort( 0x178 ),	/* Offset= 376 (786) */	/* 490 */	NdrFcShort( 0x0 ),	/* 0 */
/* 412 */	NdrFcLong( 0x14 ),	/* 20 */	/* 492 */	NdrFcLong( 0xffffffff ),	/* -1 */
/* 416 */	NdrFcShort( 0x190 ),	/* Offset= 400 (816) */	/* 494 */	0x4c,	/* FC_EMBEDDED_COMPLEX */
/* 418 */	NdrFcShort( 0xffffffff ),	/* Offset= -1 (417) */	/* 496 */	0x0,	/* 0 */
/* 420 */	0x1b,	/*	/* 498 */	NdrFcShort( 0xfffff50 ),	/* Offset= -176 (308) */
FC_CARRAY */	0x3,	/* 3 */	/* 500 */	0x5c,	/* FC_PAD */
/* 422 */	NdrFcShort( 0x4 ),	/* 4 */	/* 502 */	0x5b,	/* FC_END */
/* 424 */	0x19,	/* Corr desc: field pointer,	/* 504 */	0x1a,	/*
FC_ULONG */	0x0,	/* */	FC_BOGUS_STRUCT */	0x3,	/* 3 */
/* 426 */	NdrFcShort( 0x0 ),	/* 0 */	/* 490 */	NdrFcShort( 0x8 ),	/* 8 */
/* 428 */	0x4b,	/* FC_PP */	/* 492 */	NdrFcShort( 0x0 ),	/* 0 */
/* 430 */	0x5c,	/* FC_PAD */	/* 494 */	NdrFcShort( 0x6 ),	/* Offset= 6 (500) */
FC_VARIABLE_REPEAT */	0x48,	/*	/* 496 */	0x8,	/* FC_LONG */
FC_FIXED_OFFSET */	0x49,	/*	/* 498 */	0x36,	/*
/* 432 */	NdrFcShort( 0x4 ),	/* 4 */	/* 500 */	FC_POINTER */	/*
/* 434 */	NdrFcShort( 0x0 ),	/* 0 */	/* 502 */	0x5c,	/* FC_PAD */
/* 436 */	NdrFcShort( 0x1 ),	/* 1 */	/* 504 */	0x5b,	/* FC_END */
/* 438 */	NdrFcShort( 0x0 ),	/* 0 */	/*	0x11, 0x0, /* FC_RP */	/*
/* 440 */	NdrFcShort( 0x0 ),	/* 0 */	/* 502 */	NdrFcShort( 0xfffffe0 ),	/* Offset= -32 (470) */
/* 442 */	0x12, 0x0, /* FC_UP */	/*	/* 504 */	/*	/*
/* 444 */	NdrFcShort( 0xfffff6e ),	/* Offset= -146 (298) */	FC_BOGUS_ARRAY */	0x21,	/*
/* 446 */	0x5b,	/* FC_END */	/* 506 */	NdrFcShort( 0x0 ),	/* 0 */
			/* 508 */	0x3,	/* 3 */
			/* 510 */	NdrFcShort( 0x0 ),	/* 0 */
			/* 512 */	0x19,	/* Corr desc: field pointer,
			/* 514 */	FC_ULONG */	/*
			/* 516 */	0x0,	/* */
			/* 518 */	NdrFcShort( 0x0 ),	/* 0 */
			/* 520 */	NdrFcLong( 0xffffffff ),	/* -1 */
			/* 522 */	0x4c,	/* FC_EMBEDDED_COMPLEX */

	0x0,	/* 0 */		FC_CONSTANT_IID */	0x5a,	/*
/* 518 */	NdrFcShort( 0xfffff40 ),	/* Offset= -192 (326) */		/* 586 */	NdrFcLong( 0x2f ),	/* 47 */
/* 520 */	0x5c,	/* FC_PAD */		/* 590 */	NdrFcShort( 0x0 ),	/* 0 */
	0x5b,	/* FC_END */		/* 592 */	NdrFcShort( 0x0 ),	/* 0 */
/* 522 */				/* 594 */	0xc0,	/* 192 */
FC_BOGUS_STRUCT */	0x1a,	/*		/* 596 */	0x0,	/* 0 */
	0x3,	/* 3 */		/* 598 */	0x0,	/* 0 */
/* 524 */	NdrFcShort( 0x8 ),	/* 8 */		/* 600 */	0x0,	/* 0 */
/* 526 */	NdrFcShort( 0x0 ),	/* 0 */		/* 602 */	0x46,	/* 70 */
/* 528 */	NdrFcShort( 0x6 ),	/* Offset= 6 (534) */				
/* 530 */	0x8,	/* FC_LONG */				
	0x36,	/*				
FC_POINTER */						
/* 532 */	0x5c,	/* FC_PAD */				
	0x5b,	/* FC_END */				
/* 534 */						
	0x11, 0x0,	/* FC_RP */		/* 604 */	NdrFcShort( 0x1 ),	/* 1 */
/* 536 */	NdrFcShort( 0xfffffe0 ),	/* Offset= -32 (504) */		/* 606 */	0x19,	/* Corr desc: field pointer,
/* 538 */						
	0x1b,	/*		FC_ULONG */		
FC_CARRY */				/* 608 */	NdrFcShort( 0x4 ),	/* 4 */
	0x3,	/* 3 */		/* 610 */	0x1,	/* FC_BYTE */
/* 540 */	NdrFcShort( 0x4 ),	/* 4 */			0x5b,	/* FC_END */
/* 542 */	0x19,	/* Corr desc: field pointer,		/* 612 */		
FC_ULONG */					0x1a,	/*
	0x0,	/* */		FC_BOGUS_STRUCT */		
/* 544 */	NdrFcShort( 0x0 ),	/* 0 */		/* 614 */	NdrFcShort( 0x10 ),	/* 16 */
/* 546 */				/* 616 */	NdrFcShort( 0x0 ),	/* 0 */
	0x4b,	/* FC_PP */		/* 618 */	NdrFcShort( 0xa ),	/* Offset= 10 (628) */
	0x5c,	/* FC_PAD */		/* 620 */	0x8,	/* FC_LONG */
/* 548 */					0x8,	/* FC_LONG
	0x48,	/*				
FC_VARIABLE_REPEAT */				/* 622 */	0x4c,	/* FC_EMBEDDED_COMPLEX */
	0x49,	/*			0x0,	/* 0 */
FC_FIXED_OFFSET */				/* 624 */	NdrFcShort( 0xfffffd8 ),	/* Offset= -40 (584) */
/* 550 */	NdrFcShort( 0x4 ),	/* 4 */		/* 626 */	0x36,	/* FC_POINTER */
/* 552 */	NdrFcShort( 0x0 ),	/* 0 */			0x5b,	/* FC_END */
/* 554 */	NdrFcShort( 0x1 ),	/* 1 */		/* 628 */		
/* 556 */	NdrFcShort( 0x0 ),	/* 0 */			0x12, 0x0,	/* FC_UP */
/* 558 */	NdrFcShort( 0x0 ),	/* 0 */		/* 630 */	NdrFcShort( 0xfffffe4 ),	/* Offset= -28 (602) */
/* 560 */	0x12, 0x0,	/* FC_UP */		/* 632 */		
/* 562 */	NdrFcShort( 0xffffdce ),	/* Offset= -562 (0) */			0x1b,	/*
/* 564 */				FC_CARRY */		
	0x5b,	/* FC_END */			0x3,	/* 3 */
	0x8,	/* FC_LONG		/* 634 */	NdrFcShort( 0x4 ),	/* 4 */
/* 566 */	0x5c,	/* FC_PAD */		/* 636 */	0x19,	/* Corr desc: field pointer,
	0x5b,	/* FC_END */				
/* 568 */				FC_ULONG */		
	0x1a,	/*		/* 638 */	NdrFcShort( 0x0 ),	/* 0 */
FC_BOGUS_STRUCT */				/* 640 */		
	0x3,	/* 3 */			0x4b,	/* FC_PP */
/* 570 */	NdrFcShort( 0x8 ),	/* 8 */		/* 642 */	0x5c,	/* FC_PAD */
/* 572 */	NdrFcShort( 0x0 ),	/* 0 */			0x48,	/*
/* 574 */	NdrFcShort( 0x6 ),	/* Offset= 6 (580) */		FC_VARIABLE_REPEAT */		
/* 576 */	0x8,	/* FC_LONG */			0x49,	/*
	0x36,	/*		FC_FIXED_OFFSET */		
FC_POINTER */				/* 644 */	NdrFcShort( 0x4 ),	/* 4 */
/* 578 */	0x5c,	/* FC_PAD */		/* 646 */	NdrFcShort( 0x0 ),	/* 0 */
	0x5b,	/* FC_END */		/* 648 */	NdrFcShort( 0x1 ),	/* 1 */
/* 580 */				/* 650 */	NdrFcShort( 0x0 ),	/* 0 */
	0x11, 0x0,	/* FC_RP */		/* 652 */	NdrFcShort( 0x0 ),	/* 0 */
/* 582 */	NdrFcShort( 0xfffffd4 ),	/* Offset= -44 (538) */		/* 654 */	0x12, 0x0,	/* FC_UP */
/* 584 */				/* 656 */	NdrFcShort( 0xfffffd4 ),	/* Offset= -44 (612) */
	0x2f,	/* FC_IP */		/* 658 */		
					0x5b,	/* FC_END */



0x8,	/* FC_LONG	/* 722 */ NdrFcShort( 0x0 ),	/* 0 */
/* 660 */ 0x5c,	/* FC_PAD */	/* 724 */ 0x1,	/* FC_BYTE */
/* 662 */	0x5b,	/* 726 */	0x5b,
FC_BOGUS_STRUCT *	/* FC_END */	FC_PSTRUCT *	/* FC_END */
0x1a,	/*	0x16,	/*
0x3,	/* 3 */	0x3,	/* 3 */
/* 664 */ NdrFcShort( 0x8 ),	/* 8 */	/* 728 */ NdrFcShort( 0x8 ),	/* 8 */
/* 666 */ NdrFcShort( 0x0 ),	/* 0 */	/* 730 */	0x4b,
/* 668 */ NdrFcShort( 0x6 ),	/* Offset= 6 (674) */	/* 732 */	0x5c,
/* 670 */ 0x8,	/* FC_LONG */	/* 734 */	/* FC_PP */
FC_POINTER */	0x36,	FC_NO_REPEAT */	/* FC_PAD */
/* 672 */ 0x5c,	/* FC_PAD */	0x5c,	/* FC_PAD */
/* 674 */	0x5b,	/* 734 */ NdrFcShort( 0x4 ),	/* 4 */
0x11, 0x0, /* FC_RP */	/* FC_END */	/* 736 */ NdrFcShort( 0x4 ),	/* 4 */
/* 676 */ NdrFcShort( 0xfffffd4 ),	/* Offset= -44 (632) */	/* 738 */ 0x12, 0x0, /* FC_UP */	/*
/* 678 */	0x1d,	/* 740 */ NdrFcShort( 0xfffffe8 ),	/* Offset= -24 (716) */
FC_SMFARRAY *	/*	/* 742 */	0x5b,
0x0,	/* 0 */	0x8,	/* FC_END */
/* 680 */ NdrFcShort( 0x8 ),	/* 8 */	/*	0x8,
/* 682 */ 0x2,	/* FC_CHAR */	/* 744 */ 0x8,	/* FC_LONG */
/* 684 */	0x5b,	/* 746 */	0x5b,
FC_STRUCT */	/* FC_END */	FC_CARRAY *	/* FC_END */
0x15,	/*	0x1b,	/*
/* 686 */ NdrFcShort( 0x10 ),	/* 3 */	0x1,	/* 1 */
/* 688 */ 0x8,	/* 16 */	/* 748 */ NdrFcShort( 0x2 ),	/* 2 */
0x6,	/* FC_LONG */	/* 750 */ 0x19,	/* Corr desc: field pointer,
/*	/* FC_SHORT	FC_ULONG */	/*
/* 690 */ 0x6,	/* FC_SHORT */	0x0,	/* */
FC_EMBEDDED_COMPLEX *	/*	/* 752 */ NdrFcShort( 0x0 ),	/* 0 */
/* 692 */ 0x0,	/*	/* 754 */ 0x6,	/* FC_SHORT */
Offset= -15 (678) */	NdrFcShort( 0xffffffff ),	/* 756 */	0x5b,
/* 696 */	/* FC_END */	FC_PSTRUCT *	/* FC_END */
0x1a,	/*	0x16,	/*
FC_BOGUS_STRUCT *	0x5b,	/* 758 */ NdrFcShort( 0x8 ),	/* 3 */
0x3,	/* 3 */	/* 760 */	0x3,
/* 698 */ NdrFcShort( 0x18 ),	/* 24 */	/* 762 */	0x4b,
/* 700 */ NdrFcShort( 0x0 ),	/* 0 */	0x5c,	/* FC_PP */
/* 702 */ NdrFcShort( 0xa ),	/* Offset= 10 (712) */	0x46,	/* FC_PAD */
/* 704 */ 0x8,	/* FC_LONG */	FC_NO_REPEAT *	/*
FC_POINTER */	0x36,	0x5c,	/* FC_PAD */
/* 706 */ 0x4c,	/* FC_EMBEDDED_COMPLEX */	/* 764 */ NdrFcShort( 0x4 ),	/* 4 */
/* 708 */ NdrFcShort( 0xfffffe8 ),	/* 0 */	/* 766 */ NdrFcShort( 0x4 ),	/* 4 */
/* 710 */ 0x5c,	/* FC_PAD */	/* 768 */ 0x12, 0x0, /* FC_UP */	/*
/* 712 */	0x5b,	/* 770 */ NdrFcShort( 0xfffffe8 ),	/* Offset= -24 (746) */
0x11, 0x0, /* FC_RP */	/* FC_END */	/* 772 */	0x5b,
/* 714 */ NdrFcShort( 0xfffff0c ),	/* Offset= -244 (470) */	0x8,	/* FC_LONG
/* 716 */	0x8,	/*	/* FC_END */
FC_CARRAY *	0x1b,	/* 774 */ 0x8,	/* FC_LONG */
0x0,	/*	/* 776 */	0x5b,
/* 718 */ NdrFcShort( 0x1 ),	/* 0 */	FC_CARRAY *	/*
/* 720 */ 0x19,	/* 1 */	0x1b,	/*
FC_ULONG *	/* Corr desc: field pointer,	FC_CARRAY *	/* 3 */
0x0,	/*	/* 778 */ NdrFcShort( 0x4 ),	/* 4 */
0x0,	/*	/* 780 */ 0x19,	/* Corr desc: field pointer,
0x0,	/*	FC_ULONG *	/*
0x0,	/*	0x0,	/*

/* 782 */ NdrFcShort( 0x0 ),	/* 0 */	/* 842 */ 0x5c,	/* FC_PAD */
/* 784 */ 0x8,	/* FC_LONG */		0x5b,
	0x5b,	/* FC_END */	/* FC_END */
/* 786 */		/* 844 */	
	0x16,		0x1b,
FC_PSTRUCT */		FC_CARRAY */	/*
	0x3,		0x3,
/* 788 */ NdrFcShort( 0x8 ),	/* 3 */	/* 846 */ NdrFcShort( 0x8 ),	/* 3 */
/* 790 */	/* 8 */	/* 848 */ 0x7,	/* 8 */
	0x4b,	/* Corr desc: FC_USHORT */	0x0,
/* 792 */	0x5c,	/* FC_PP */	/*
	0x46,	/* FC_PAD */	/* 850 */ NdrFcShort( 0xffd8 ),
FC_NO_REPEAT */		/* FC_PAD */	/* -40 */
	0x5c,	/* FC_PAD */	/* 852 */ 0x4c,
/* 794 */ NdrFcShort( 0x4 ),	/* 4 */		/* FC_EMBEDDED_COMPLEX */
/* 796 */ NdrFcShort( 0x4 ),	/* 4 */	/* 854 */ NdrFcShort( 0xfffffee ),	/* 0 */
/* 798 */ 0x12, 0x0, /* FC_UP */		/* Offset= -18 (836) */	/*
/* 800 */ NdrFcShort( 0xfffffe8 ),	/* Offset= -24 (776) */	/* 856 */ 0x5c,	/* FC_PAD */
/* 802 */			0x5b,
	0x5b,	/* FC_END */	/* FC_END */
	0x8,	/* FC_LONG	/*
/*			0x1a,
/* 804 */ 0x8,	/* FC_LONG */		/*
/* 806 */	0x5b,	/* FC_END */	FC_BOGUS_STRUCT */
	0x1b,	/*	0x3,
FC_CARRAY */			/* 3 */
	0x7,	/* 7 */	/* 860 */ NdrFcShort( 0x28 ),
/* 808 */ NdrFcShort( 0x8 ),	/* 8 */		/* 40 */
/* 810 */ 0x19,	/* Corr desc: field pointer,	/* 862 */ NdrFcShort( 0xfffffee ),	/* Offset= -18 (844) */
FC_ULONG */		/* 864 */ NdrFcShort( 0x0 ),	/* Offset= 0 (864) */
	0x0,	/* FC_SHORT */	/*
/* 812 */ NdrFcShort( 0x0 ),	/*	/* 866 */ 0x6,	0x6,
/* 814 */ 0xb,	/* FC_HYPER */		/* FC_SHORT
/* 816 */	0x5b,	/* FC_END */	/*
	0x16,	/*	/* 868 */ 0x38,
FC_PSTRUCT */			/* FC_ALIGNM4 */
	0x3,	/* 3 */	0x8,
/* 818 */ NdrFcShort( 0x8 ),	/* 8 */		/* FC_LONG
/* 820 */	0x4b,	/* FC_PP */	/*
	0x5c,	/* FC_PAD */	/* 870 */ 0x8,
/* 822 */	0x46,	/*	/* FC_LONG */
FC_NO_REPEAT */			0x4c,
	0x5c,	/* FC_PAD */	/*
/* 824 */ NdrFcShort( 0x4 ),	/* 4 */		FC_EMBEDDED_COMPLEX */
/* 826 */ NdrFcShort( 0x4 ),	/* 4 */	/* 872 */ 0x0,	/* 0 */
/* 828 */ 0x12, 0x0, /* FC_UP */			NdrFcShort( 0xffffdf7 ),
/* 830 */ NdrFcShort( 0xfffffe8 ),	/* Offset= -24 (806) */	Offset= -521 (352) */	/*
/* 832 */	0x5b,	/* FC_END */	0x5b,
	0x8,	/* FC_LONG	/* FC_END */
/*			0x12, 0x0, /* FC_UP */
/* 834 */ 0x8,	/* FC_LONG */	/* 876 */	/*
/* 836 */	0x5b,	/* FC_END */	/* 878 */ NdrFcShort( 0xfffffef6 ),
	0x15,	/*	/* Offset= -266 (612) */
FC_STRUCT */			/* 880 */
	0x3,	/* 3 */	0x12, 0x8, /* FC_UP
/* 838 */ NdrFcShort( 0x8 ),	/* 8 */		/* FC_UP
/* 840 */ 0x8,	/* FC_LONG */	[simple_pointer] */	/* FC_BYTE */
/*	0x8,	/* FC_LONG	/*
		/* 882 */ 0x1,	0x5c,
			/* FC_PAD */
		/* 884 */	0x12, 0x8, /* FC_UP
		[simple_pointer] */	/* FC_UP
		/* 886 */ 0x6,	/* FC_SHORT */
			/* FC_PAD */
		/* 888 */	0x5c,
		[simple_pointer] */	/* FC_PAD */
		/* 890 */ 0x8,	0x12, 0x8, /* FC_UP
		/* 892 */	/* FC_UP
		[simple_pointer] */	/* FC_UP
		/* 894 */ 0xa,	/* FC_FLOAT */
			0x5c,
		/* 896 */	/* FC_PAD */
		[simple_pointer] */	0x12, 0x8, /* FC_UP
		/* 898 */ 0xc,	/* FC_UP
			/* FC_DOUBLE */
		/* 900 */	0x5c,
		/* 902 */ NdrFcShort( 0xffffd90 ),	/* FC_PAD */
		/* Offset= -624 (278) */	/*
		/* 904 */	0x12, 0x0, /* FC_UP */
			/* FC_UP */
			0x12, 0x10,

```

/* 906 */ NdrFcShort( 0xffffd92 ), /* Offset= -622 (284) */
/* 908 */
                                0x12, 0x10, /* FC_UP */
/* 910 */ NdrFcShort( 0xffffda6 ), /* Offset= -602 (308) */
/* 912 */
                                0x12, 0x10, /* FC_UP */
/* 914 */ NdrFcShort( 0xffffdb4 ), /* Offset= -588 (326) */
/* 916 */
                                0x12, 0x10, /* FC_UP */
/* 918 */ NdrFcShort( 0xffffdc2 ), /* Offset= -574 (344) */
/* 920 */
                                0x12, 0x10, /* FC_UP */
/* 922 */ NdrFcShort( 0x2 ), /* Offset= 2 (924) */
/* 924 */
                                0x12, 0x0, /* FC_UP */
/* 926 */ NdrFcShort( 0xffffc62 ), /* Offset= -926 (0) */
/* 928 */
                                0x15, /*
FC_STRUCT */
                                0x7, /* 7 */
/* 930 */ NdrFcShort( 0x10 ), /* 16 */
/* 932 */ 0x6, /* FC_SHORT */
                                0x1, /* FC_BYTE
*/
/* 934 */ 0x1, /* FC_BYTE */
FC_ALIGNM4 */
                                0x38, /*
/* 936 */ 0x8, /* FC_LONG */
                                0x39, /*
FC_ALIGNM8 */
/* 938 */ 0xb, /* FC_HYPER */
                                0x5b, /* FC_END */
/* 940 */
                                0x12, 0x0, /* FC_UP */
/* 942 */ NdrFcShort( 0xfffff2 ), /* Offset= -14 (928) */
/* 944 */
                                0x12, 0x8, /* FC_UP
[simple_pointer] */
/* 946 */ 0x2, /* FC_CHAR */
                                0x5c, /* FC_PAD */
/* 948 */
                                0x1a, /*
FC_BOGUS_STRUCT */
                                0x7, /* 7 */
/* 950 */ NdrFcShort( 0x20 ), /* 32 */
/* 952 */ NdrFcShort( 0x0 ), /* 0 */
/* 954 */ NdrFcShort( 0x0 ), /* Offset= 0 (954) */
/* 956 */ 0x8, /* FC_LONG */
                                0x8, /* FC_LONG
*/
/* 958 */ 0x6, /* FC_SHORT */
                                0x6, /* FC_SHORT
*/
/* 960 */ 0x6, /* FC_SHORT */
                                0x6, /* FC_SHORT
*/
/* 962 */ 0x4c, /* FC_EMBEDDED_COMPLEX */
                                0x0, /* 0 */
/* 964 */ NdrFcShort( 0xffffc42 ), /* Offset= -958 (6) */
/* 966 */ 0x5c, /* FC_PAD */
                                0x5b, /* FC_END */
/* 968 */ 0xb4, /* FC_USER_MARSHAL */
                                0x83, /* 131 */
/* 970 */ NdrFcShort( 0x0 ), /* 0 */
/* 972 */ NdrFcShort( 0x10 ), /* 16 */
/* 974 */ NdrFcShort( 0x0 ), /* 0 */
/* 976 */ NdrFcShort( 0xffffc32 ), /* Offset= -974 (2) */
/* 978 */
                                0x11, 0x4, /* FC_RP
[allocated_on_stack] */
/* 980 */ NdrFcShort( 0x6 ), /* Offset= 6 (986) */
/* 982 */
                                0x13, 0x0, /* FC_OP */
/* 984 */ NdrFcShort( 0xfffff4c ), /* Offset= -36 (948) */
/* 986 */ 0xb4, /* FC_USER_MARSHAL */
                                0x83, /* 131 */
/* 988 */ NdrFcShort( 0x0 ), /* 0 */
/* 990 */ NdrFcShort( 0x10 ), /* 16 */
/* 992 */ NdrFcShort( 0x0 ), /* 0 */
/* 994 */ NdrFcShort( 0xfffff4 ), /* 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,
2,
0, /* table of [async_uuid] interfaces */
0, /* Filler1 */
0, /* Filler2 */
0 /* Filler3 */
};

```

## tpcc\_com\_rem.cpp

```
/* FILE: TPCC_COM.CPP
 * Microsoft TPC-C Kit Ver.
4.20.000
 * Copyright Microsoft,
1999
 * All Rights Reserved
 *
 * not yet audited
 *
 * PURPOSE: Source file for TPC-C COM+ class
implementation.
 * Contact: Charles Levine (clevine@microsoft.com)
 *
 * Change history:
 * 4.20.000 - first version
 */

// needed for CoInitializeEx
#define _WIN32_WINNT 0x0400

#include <windows.h>

// need to declare functions for export
#define DllDecl __declspec( dllexport )

#include "..\..\common\src\trans.h" //tpckit
transaction header contains definitions of structures specific to TPC-C
#include "..\..\common\src\error.h"
#include "..\..\common\src\txn_base.h"
#include "tpcc_com_rem.h"

#include "..\..\tpcc_com_remote_ps\src\tpcc_com_remote_ps_i.c"
#include "..\..\tpcc_com_remote\src\tpcc_com_remote_i.c"

// wrapper routine for class constructor
__declspec(dllexport) CTPCC_COM_REM*
CTPCC_COM_REM_new(BOOL bSinglePool)
{
    return new CTPCC_COM_REM(bSinglePool);
}

CTPCC_COM_REM::CTPCC_COM_REM(BOOL bSinglePool)
{
    HRESULT hr = NULL;
    long lRet = 0;
    ULONG ulTmpSize = 0;

    m_bSinglePool = bSinglePool;

    m_pNewOrder = NULL;
    m_pPayment = NULL;
    m_pStockLevel = NULL;
    m_pOrderStatus = NULL;
}

/*
(COM_DATA*)CoTaskMemAlloc(sizeof(COM_DATA));
if (!m_pTxn)
    throw new CCOMERR( E_FAIL );
*/

ulTmpSize = (ULONG) sizeof(COM_DATA);
VariantInit(&m_vTxn);
m_vTxn.vt = VT_SAFEARRAY;

m_vTxn.parray = SafeArrayCreateVector(
```

```
VT_UI1,
ulTmpSize,
ulTmpSize);
if (!m_vTxn.parray)
    throw new CCOMERR( E_FAIL );

memset((void*)m_vTxn.parray->pvData,0,ulTmpSize);

m_pTxn = (COM_DATA*)m_vTxn.parray->pvData;

hr = CoInitializeEx(NULL, COINIT_MULTITHREADED);
if (FAILED(hr))
{
    throw new CCOMERR( hr );
}

// create components
if (m_bSinglePool)
{
    hr = CoCreateInstance(CLSID_TPCCrem,
NULL, CLSCTX_SERVER, IID_ITPCCrem, (void
***)&m_pNewOrder);
if (FAILED(hr))
    throw new CCOMERR(hr);

// all txns will use same component
m_pPayment = m_pNewOrder;
m_pStockLevel = m_pNewOrder;
m_pOrderStatus = m_pNewOrder;
}
else
{
// use different components for each txn

hr = CoCreateInstance(CLSID_NewOrder,
NULL, CLSCTX_SERVER, IID_ITPCCrem, (void
***)&m_pNewOrder);
if (FAILED(hr))
    throw new CCOMERR(hr);

hr = CoCreateInstance(CLSID_Payment, NULL,
CLSCTX_SERVER, IID_ITPCCrem, (void ***)&m_pPayment);
if (FAILED(hr))
    throw new CCOMERR(hr);

hr = CoCreateInstance(CLSID_StockLevel,
NULL, CLSCTX_SERVER, IID_ITPCCrem, (void
***)&m_pStockLevel);
if (FAILED(hr))
    throw new CCOMERR(hr);

hr = CoCreateInstance(CLSID_OrderStatus,
NULL, CLSCTX_SERVER, IID_ITPCCrem, (void
***)&m_pOrderStatus);
if (FAILED(hr))
    throw new CCOMERR(hr);
}

// call setcomplete to release each component back into pool
hr = m_pNewOrder->CallSetComplete();
if (FAILED(hr))
    throw new CCOMERR(hr);

if (!m_bSinglePool)
{
```

```

        hr = m_pPayment->CallSetComplete();
        if (FAILED(hr))
            throw new CCOMERR(hr);

        hr = m_pStockLevel->CallSetComplete();
        if (FAILED(hr))
            throw new CCOMERR(hr);

        hr = m_pOrderStatus->CallSetComplete();
        if (FAILED(hr))
            throw new CCOMERR(hr);
    }
}

CTPCC_COM_REM::~~CTPCC_COM_REM()
{
    if (m_pTxn)
        SafeArrayDestroy(m_vTxn.parray);

    ReleaseInterface(m_pNewOrder);
    if (!m_bSinglePool)
    {
        ReleaseInterface(m_pPayment);
        ReleaseInterface(m_pStockLevel);
        ReleaseInterface(m_pOrderStatus);
    }
    CoUninitialize();
}

void CTPCC_COM_REM::NewOrder()
{
    VARIANT vTxn_out;

    HRESULT hr = m_pNewOrder->NewOrder(m_vTxn,
&vTxn_out);
    if (FAILED(hr))
        throw new CCOMERR( hr );
    memcpy(m_pTxn, (void
*)vTxn_out.parray->pvData,vTxn_out.parray->rgsabound[0].cElement
s);
    SafeArrayDestroy(vTxn_out.parray);

    if ( m_pTxn->ErrorType != ERR_SUCCESS )
        throw new CCOMERR( m_pTxn->ErrorType,
m_pTxn->error );
}

void CTPCC_COM_REM::Payment()
{
    VARIANT vTxn_out;

    HRESULT hr = m_pPayment->Payment(m_vTxn,
&vTxn_out);
    if (FAILED(hr))
        throw new CCOMERR( hr );
    memcpy(m_pTxn, (void
*)vTxn_out.parray->pvData,vTxn_out.parray->rgsabound[0].cElement
s);
    SafeArrayDestroy(vTxn_out.parray);

    if ( m_pTxn->ErrorType != ERR_SUCCESS )
        throw new CCOMERR( m_pTxn->ErrorType,
m_pTxn->error );
}

void CTPCC_COM_REM::StockLevel()
{

```

```

    VARIANT vTxn_out;

    HRESULT hr = m_pStockLevel->StockLevel(m_vTxn,
&vTxn_out);
    if (FAILED(hr))
        throw new CCOMERR( hr );
    memcpy(m_pTxn, (void
*)vTxn_out.parray->pvData,vTxn_out.parray->rgsabound[0].cElement
s);
    SafeArrayDestroy(vTxn_out.parray);

    if ( m_pTxn->ErrorType != ERR_SUCCESS )
        throw new CCOMERR( m_pTxn->ErrorType,
m_pTxn->error );
}

void CTPCC_COM_REM::OrderStatus()
{
    VARIANT vTxn_out;

    HRESULT hr = m_pOrderStatus->OrderStatus(m_vTxn,
&vTxn_out);
    if (FAILED(hr))
        throw new CCOMERR( hr );
    memcpy(m_pTxn, (void
*)vTxn_out.parray->pvData,vTxn_out.parray->rgsabound[0].cElement
s);
    SafeArrayDestroy(vTxn_out.parray);

    if ( m_pTxn->ErrorType != ERR_SUCCESS )
        throw new CCOMERR( m_pTxn->ErrorType,
m_pTxn->error );
}

```

### **tpcc\_com\_rem.h**

```

/*      FILE:          TPCC_COM.H
*                               Microsoft TPC-C Kit Ver.
4.20.000
*                               Copyright Microsoft,
1999
*                               All Rights Reserved
*                               not yet audited
*
*      PURPOSE:       Header file for TPC-C COM+ class
implementation.
*
*      Change history:
*                               4.20.000 - first version
*/

#pragma once

#include <stdio.h>
#include "..\..\tpcc_com_ps\src\tpcc_com_ps.h"
#include "..\..\tpcc_com_remote_ps\src\tpcc_com_remote_ps.h"

// need to declare functions for import, unless define has already been
created
// by the DLL's .cpp module for export.
#ifdef DllDecl
#define DllDecl __declspec( dllimport )
#endif

class CCOMERR : public CBaseErr

```

```

{
private:
    char m_szErrorText[64];

public:
    // use this interface for genuine COM errors
    CCOMERR( HRESULT hr )
    {
        m_hr = hr;
        m_iErrorType = 0;
        m_iError = 0;
    }

    // use this interface to impersonate a non-COM
error type
    CCOMERR( int iErrorType, int iError )
    {
        m_iErrorType = iErrorType;
        m_iError = iError;
        m_hr = S_OK;
    }

    int m_hr;
    int m_iErrorType;
    int m_iError;

    // A CCOMERR class can impersonate another
class, which happens if the error
// was not actually a COM Services error, but
was simply transmitted back via COM.
    int ErrorType()
    {
        if (m_iErrorType == 0)
            return ERR_TYPE_COM;
        else
            return m_iErrorType;
    }

    int ErrorNum() {return m_hr;}

    char *ErrorText()
    {
        if (m_hr == S_OK)
            sprintf( m_szErrorText,
"Error: Class %d, error # %d", m_iErrorType, m_iError);
        else
            sprintf( m_szErrorText,
"Error: COM HRESULT %x", m_hr );
        return m_szErrorText;
    }
};

class DllDecl CTPCC_COM : public CTPCC_BASE
{
private:
    BOOL m_bSinglePool;

    // COM Interface pointers
    ITPCC* m_pNewOrder;
    ITPCC* m_pPayment;
    ITPCC* m_pStockLevel;
    ITPCC* m_pOrderStatus;

    struct COM_DATA
    {
        int ErrorType;

```

```

        int error;
        union
        {
            NEW_ORDER_DATA
            PAYMENT_DATA
            DELIVERY_DATA
            STOCK_LEVEL_DATA
            ORDER_STATUS_DATA OrderStatus;
        } u;
    } *m_pTxn;

    VARIANT m_vTxn;

public:
    CTPCC_COM(BOOL bSinglePool);
    ~CTPCC_COM(void);

    inline PNEW_ORDER_DATA
    BuffAddr_NewOrder() { return
&m_pTxn->u.NewOrder; };
    inline PPAYMENT_DATA
    BuffAddr_Payment() { return &m_pTxn->u.Payment; };
    inline PDELIVERY_DATA
    BuffAddr_Delivery() { return &m_pTxn->u.Delivery; };
    inline PSTOCK_LEVEL_DATA
    BuffAddr_StockLevel() { return &m_pTxn->u.StockLevel; };
    inline PORDER_STATUS_DATA
    BuffAddr_OrderStatus() { return &m_pTxn->u.OrderStatus;
};

    void NewOrder ();
    void Payment ();
    void StockLevel ();
    void OrderStatus ();
    void Delivery () { throw new
CCOMERR(E_NOTIMPL); } // not supported
};

inline void ReleaseInterface(IUnknown *pUnk)
{
    if (pUnk)
    {
        pUnk->Release();
        pUnk = NULL;
    }
}

// wrapper routine for class constructor
extern "C" __declspec(dllexport) CTPCC_COM*
CTPCC_COM_new(BOOL);

typedef CTPCC_COM* (TYPE_CTPCC_COM)(BOOL);

class DllDecl CTPCC_COM_REM : public CTPCC_BASE
{
private:
    BOOL m_bSinglePool;

    // COM Interface pointers
    ITPCCrem* m_pNewOrder;

```

```

m_pPayment;      ITPCCrem*
m_pStockLevel;  ITPCCrem*
m_pOrderStatus; ITPCCrem*

struct COM_DATA
{
    int ErrorType;
    int error;
    union
    {
        NEW_ORDER_DATA
            NewOrder;
        PAYMENT_DATA
            Payment;
        DELIVERY_DATA
            Delivery;
        STOCK_LEVEL_DATA
            StockLevel;
        ORDER_STATUS_DATA
            OrderStatus;
    } u;
    *m_pTxn;

    VARIANT m_vTxn;
public:
    CTPCC_COM_REM(BOOL bSinglePool);
    ~CTPCC_COM_REM(void);

    inline PNEW_ORDER_DATA
    BuffAddr_NewOrder() { return
    &m_pTxn->u.NewOrder; };
    inline PPAYMENT_DATA
    BuffAddr_Payment() { return &m_pTxn->u.Payment; };
    inline PDELIVERY_DATA
    BuffAddr_Delivery() { return &m_pTxn->u.Delivery; };
    inline PSTOCK_LEVEL_DATA
    BuffAddr_StockLevel() { return &m_pTxn->u.StockLevel; };
    inline PORDER_STATUS_DATA
    BuffAddr_OrderStatus() { return &m_pTxn->u.OrderStatus;
};

    void NewOrder      ();
    void Payment       ();
    void StockLevel    ();
    void OrderStatus   ();
    void Delivery      () { throw new
CCOMERR(E_NOTIMPL); } // not supported
};

// wrapper routine for class constructor
extern "C" __declspec(dllexport) CTPCC_COM_REM*
CTPCC_COM_REMOTE_new(BOOL);

typedef CTPCC_COM_REM* (TYPE_CTPCC_COM_REM)(BOOL);

tpcc_com_remote.cpp

/*      FILE:      TPCC_COM_REMOTE.CPP
*
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
class.
*      Contact:      Charles Levine (clevine@microsoft.com)
*
*      Change history:
*      4.20.000 - updated rev number to match kit
*/

#define STRICT
#define _WIN32_WINNT 0x0400
#define _ATL_APARTMENT_THREADED

#include <stdio.h>
#include <atbase.h>
//You may derive a class from CComModule and use it if you want to
override
//something, but do not change the name of _Module
extern CComModule _Module;

#include <atlcom.h>
#include <initguid.h>
#include <transact.h>
#include <atimpl.cpp>
#include <comsvcs.h>

#include <sqltypes.h>
#include <sql.h>
#include <sqlext.h>

#include "tpcc_com_remote_ps.h"
#include "..\common\src\trans.h"
//tpckit transaction header contains definations of structures
specific to TPC-C
#include "..\common\src\txn_base.h"
#include "..\common\src\error.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 "resource.h"
#include "tpcc_com_remote.h"
#include "tpcc_com_remote_i.c"
#include "Methods.h"
#include "..\tpcc_com_remote_ps\src\tpcc_com_remote_ps_i.c"
#include "..\common\src\ReadRegistry.cpp"

CComModule _Module;

BEGIN_OBJECT_MAP(ObjectMap)
    OBJECT_ENTRY(CLSID_TPCCrem, CTPCCrem)
END_OBJECT_MAP()

// configuration settings from registry
TPCCREGISTRYDATA Reg;
char
szMyComputerName[MAX_COMPUTERNAME_LENGTH+1];

```

```

static HINSTANCE hLibInstanceDb = NULL;

TYPE_CTPCC_DBLIB      *pCTPCC_DBLIB_new;
TYPE_CTPCC_ODBC      *pCTPCC_ODBC_new;

////////////////////////////////////
// DLL Entry Point

extern "C"
BOOL WINAPI DllMain(HINSTANCE hInstance, DWORD
dwReason, LPVOID /**lpReserved*/)
{
    char szDllName[128];

    try
    {
        if (dwReason == DLL_PROCESS_ATTACH)
        {
            _Module.Init(ObjectMap, hInstance);

DisableThreadLibraryCalls(hInstance);

            DWORD dwSize =
MAX_COMPUTERNAME_LENGTH+1;

            GetComputerName(szMyComputerName, &dwSize);
            szMyComputerName[dwSize] = 0;

            if ( ReadTPCCRegistrySettings(
&Reg ) )
                throw new
CCOMPONENT_ERR( ERR_MISSING_REGISTRY_ENTRIES);

            if (Reg.eDB_Protocol == DBLIB)
            {
                strcpy( szDllName,
Reg.szPath );
                strcat( szDllName,
"tpcc_dblib.dll");
                hLibInstanceDb =
LoadLibrary( szDllName );
                if (hLibInstanceDb ==
NULL)
                    throw new
CCOMPONENT_ERR( ERR_LOADDLL_FAILED,szDllName,
GetLastError());

                // get function pointer to
wrapper for class constructor
                pCTPCC_DBLIB_new =
(TYPE_CTPCC_DBLIB*)
GetProcAddress(hLibInstanceDb,"CTPCC_DBLIB_new");
                if (pCTPCC_DBLIB_new
== NULL)
                    throw new
CCOMPONENT_ERR( ERR_GETPROCADDR_FAILED,
szDllName, GetLastError());

                ODBC)
            else if (Reg.eDB_Protocol ==
{
                strcpy( szDllName,
Reg.szPath );
                strcat( szDllName,
"tpcc_odbc.dll");

```

```

                hLibInstanceDb =
                if (hLibInstanceDb ==
                throw new
CCOMPONENT_ERR( ERR_LOADDLL_FAILED,szDllName,
GetLastError());

                // get function pointer to
wrapper for class constructor
                pCTPCC_ODBC_new =
(TYPE_CTPCC_ODBC*)
GetProcAddress(hLibInstanceDb,"CTPCC_ODBC_new");
                if (pCTPCC_ODBC_new
== NULL)
                    throw new
CCOMPONENT_ERR( ERR_GETPROCADDR_FAILED,
szDllName, GetLastError());
            }
            else
                throw new
CCOMPONENT_ERR( ERR_UNKNOWN_DB_PROTOCOL);
        }
        else if (dwReason ==
DLL_PROCESS_DETACH)
            _Module.Term();
    }
    catch (CBaseErr *e)
    {
        WriteMessageToEventLog(e->ErrorText());
        delete e;
        return FALSE;
    }
    catch (...)
    {
        WriteMessageToEventLog(TEXT("Unhandled
exception in object DllMain"));
        return FALSE;
    }
    return TRUE;    // OK
}

////////////////////////////////////
// Used to determine whether the DLL can be unloaded by OLE

STDAPI DllCanUnloadNow(void)
{
    return (_Module.GetLockCount()==0) ? S_OK : S_FALSE;
}

////////////////////////////////////
// Returns a class factory to create an object of the requested type

STDAPI DllGetClassObject(REFCLSID rclsid, REFIID riid, LPVOID*
ppv)
{
    return _Module.GetClassObject(rclsid, riid, ppv);
}

////////////////////////////////////
// DllRegisterServer - Adds entries to the system registry

STDAPI DllRegisterServer(void)
{

```



```

// registers object, typelib and all interfaces in typelib
return _Module.RegisterServer(TRUE);
}

////////////////////////////////////
// DllUnregisterServer - Removes entries from the system registry

STDAPI DllUnregisterServer(void)
{
    _Module.UnregisterServer();
    return S_OK;
}

static void WriteMessageToEventLog(LPTSTR lpszMsg)
{
    TCHAR szMsg[256];
    HANDLE hEventSource;
    LPTSTR lpszStrings[2];

    // Use event logging to log the error.
    //
    hEventSource = RegisterEventSource(NULL,
    TEXT("tpcc_com_remote.dll"));

    _stprintf(szMsg, TEXT("Error in COM+ TPC-C Component: "));
    lpszStrings[0] = szMsg;
    lpszStrings[1] = lpszMsg;

    if (hEventSource != NULL)
    {
        ReportEvent(hEventSource, // handle of event source
        EVENTLOG_ERROR_TYPE, // event type
        0, // event category
        0, // event ID
        NULL, // current user's SID
        2, // strings in lpszStrings
        0, // no bytes of raw data
        (LPCTSTR *)lpszStrings, // array of error strings
        NULL); // no raw data

        (VOID) DeregisterEventSource(hEventSource);
    }
}

inline void ReleaseInterface(IUnknown *pUnk)
{
    if (pUnk)
    {
        pUnk->Release();
        pUnk = NULL;
    }
}

/* FUNCTION: CCOMPONENT_ERR::ErrorText
*
*/

char* CCOMPONENT_ERR::ErrorText(void)
{
    static SERRORMSG errorMsgs[] =
    {
        { ERR_MISSING_REGISTRY_ENTRIES,
        "Required entries missing from registry."
        },
        { ERR_LOADDLL_FAILED,
        "Load of DLL failed. DLL="
        },
    },

```

```

        { ERR_GETPROCADDR_FAILED,
        "Could not map proc in DLL. GetProcAddress error. DLL="
        },
        { ERR_UNKNOWN_DB_PROTOCOL,
        "Unknown database protocol specified in registry."
        },
        { 0,
        ""
        }
    };

    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;
}

CTPCC_Common_Remote::CTPCC_Common_Remote()
{
    m_pTxn = NULL;
    m_bCanBePooled = TRUE;
}

CTPCC_Common_Remote::~CTPCC_Common_Remote()
{
    if (m_pTxn)
        delete m_pTxn;
}

HRESULT CTPCC_Common_Remote::CallSetComplete()
{
    IObjectContext* pObjectContext = NULL;

    // get our object context
    HRESULT hr = CoGetObjectContext( IID_IObjectContext,
    (void **)&pObjectContext );
    pObjectContext->SetComplete();
    ReleaseInterface(pObjectContext);
    return hr;
}

//
// called by the ctor activator
//

```

```

STDMETHODIMP CTPCC_Common_Remote::Construct(IDispatch *
pUnk)
{
    // Code to access construction string, if needed later...
    // if (!pUnk)
    //     return E_UNEXPECTED;
    // IObjectConstructString * pString = NULL;
    // HRESULT hr =
pUnk->QueryInterface(IID_IObjectConstructString, (void
***)&pString);
    // pString->Release();
    try
    {
        if (Reg.eDB_Protocol == ODBC)
            m_pTxn = pCTPCC_ODBC_new(
Reg.szDbServer, Reg.szDbUser, Reg.szDbPassword,
szMyComputerName, Reg.szDbName);
        else if (Reg.eDB_Protocol == DBLIB)
            m_pTxn = pCTPCC_DBLIB_new(
Reg.szDbServer, Reg.szDbUser, Reg.szDbPassword,
szMyComputerName, Reg.szDbName);
    }
    catch (CBaseErr *e)
    {
        WriteMessageToEventLog(e->ErrorText());
        delete e;
        return E_FAIL;
    }
    catch (...)
    {
        WriteMessageToEventLog(TEXT("Unhandled
exception in object ::Construct"));
        return E_FAIL;
    }
    return S_OK;
}

HRESULT CTPCC_Common_Remote::NewOrder(VARIANT txn_in,
VARIANT* txn_out)
{
    PNEW_ORDER_DATA    pNewOrder;
    COM_DATA            *pData;
    try
    {
        pData = (COM_DATA*)txn_in.parray->pvData;
        pNewOrder = m_pTxn->BuffAddr_NewOrder();

        memcpy(pNewOrder, &pData->u.NewOrder,
sizeof(NEW_ORDER_DATA));

        VariantInit(txn_out);
        txn_out->vt = VT_SAFEARRAY;
        txn_out->parray = SafeArrayCreateVector(
VT_UI1,
txn_in.parray->rgsabound->cElements,
txn_in.parray->rgsabound->cElements);
        pData = (COM_DATA*)
txn_out->parray->pvData;

        m_pTxn->Payment(FALSE);

        memcpy( &pData->u.Payment, pNewOrder,
sizeof(PAYMENT_DATA));

        pData->retval = ERR_SUCCESS;
        pData->error = 0;
        return S_OK;
    }
    catch (CBaseErr *e)
    {
        // check for lost database connection; if yes,
component is toast
    }
}

pData->retval = ERR_SUCCESS;
pData->error = 0;
return S_OK;
}
catch (CBaseErr *e)
{
    // check for lost database connection; if yes,
component is toast
    if ( ((e->ErrorType() == ERR_TYPE_DBLIB)
&& (e->ErrorNum() == 10005)) ||
        ((e->ErrorType() ==
ERR_TYPE_ODBC) && (e->ErrorNum() == 10054)) )
        m_bCanBePooled = FALSE;

    pData->retval = e->ErrorType();
    pData->error = e->ErrorNum();
    delete e;
    return E_FAIL;
}
catch (...)
{
    WriteMessageToEventLog(TEXT("Unhandled
exception.));
    pData->retval = ERR_TYPE_LOGIC;
    pData->error = 0;
    m_bCanBePooled = FALSE;
    return E_FAIL;
}
}

HRESULT CTPCC_Common_Remote::Payment(VARIANT txn_in,
VARIANT* txn_out)
{
    PPAYMENT_DATA pPayment;
    COM_DATA      *pData;
    try
    {
        pData = (COM_DATA*)txn_in.parray->pvData;
        pPayment = m_pTxn->BuffAddr_Payment();

        memcpy(pPayment, &pData->u.Payment,
sizeof(PAYMENT_DATA));

        VariantInit(txn_out);
        txn_out->vt = VT_SAFEARRAY;
        txn_out->parray = SafeArrayCreateVector(
VT_UI1,
txn_in.parray->rgsabound->cElements,
txn_in.parray->rgsabound->cElements);
        pData = (COM_DATA*)
txn_out->parray->pvData;

        m_pTxn->Payment(FALSE);

        memcpy( &pData->u.Payment, pPayment,
sizeof(PAYMENT_DATA));

        pData->retval = ERR_SUCCESS;
        pData->error = 0;
        return S_OK;
    }
    catch (CBaseErr *e)
    {
        // check for lost database connection; if yes,
component is toast
    }
}

```

```

        if ( ((e->ErrorType() == ERR_TYPE_DBLIB)
&& (e->ErrorNum() == 10005)) ||
        ((e->ErrorType() ==
ERR_TYPE_ODBC) && (e->ErrorNum() == 10054)) )
            m_bCanBePooled = FALSE;

        pData->retval = e->ErrorType();
        pData->error = e->ErrorNum();
        delete e;
        return E_FAIL;
    }
    catch (...)
    {
        WriteMessageToEventLog(TEXT("Unhandled
exception.));

        pData->retval = ERR_TYPE_LOGIC;
        pData->error = 0;
        m_bCanBePooled = FALSE;
        return E_FAIL;
    }
}

HRESULT CTPCC_Common_Remote::StockLevel(VARIANT txn_in,
VARIANT* txn_out)
{
    PSTOCK_LEVEL_DATA    pStockLevel;
    COM_DATA              *pData;

    try
    {
        pData = (COM_DATA*)txn_in.parray->pvData;
        pStockLevel =
m_pTxn->BuffAddr_StockLevel();

        memcpy(pStockLevel, &pData->u.StockLevel,
sizeof(STOCK_LEVEL_DATA));

        VariantInit(txn_out);
        txn_out->vt = VT_SAFEARRAY;
        txn_out->parray = SafeArrayCreateVector(
VT_UI1,
txn_in.parray->rgsabound->cElements,
txn_in.parray->rgsabound->cElements);
        pData =
(COM_DATA*)txn_out->parray->pvData;

        m_pTxn->StockLevel();

        memcpy( &pData->u.StockLevel, pStockLevel,
sizeof(STOCK_LEVEL_DATA));

        pData->retval = ERR_SUCCESS;
        pData->error = 0;
        return S_OK;
    }
    catch (CBaseErr *e)
    {
        // check for lost database connection; if yes,
component is toast
        if ( ((e->ErrorType() == ERR_TYPE_DBLIB)
&& (e->ErrorNum() == 10005)) ||
        ((e->ErrorType() ==
ERR_TYPE_ODBC) && (e->ErrorNum() == 10054)) )
            m_bCanBePooled = FALSE;
    }
}

```

```

        pData->retval = e->ErrorType();
        pData->error = e->ErrorNum();
        delete e;
        return E_FAIL;
    }
    catch (...)
    {
        WriteMessageToEventLog(TEXT("Unhandled
exception.));

        pData->retval = ERR_TYPE_LOGIC;
        pData->error = 0;
        m_bCanBePooled = FALSE;
        return E_FAIL;
    }
}

HRESULT CTPCC_Common_Remote::OrderStatus(VARIANT
txn_in, VARIANT* txn_out)
{
    PORDER_STATUS_DATA    pOrderStatus;
    COM_DATA              *pData;
    try
    {
        pData = (COM_DATA*)txn_in.parray->pvData;
        pOrderStatus =
m_pTxn->BuffAddr_OrderStatus();

        memcpy(pOrderStatus, &pData->u.OrderStatus,
sizeof(ORDER_STATUS_DATA));

        VariantInit(txn_out);
        txn_out->vt = VT_SAFEARRAY;
        txn_out->parray = SafeArrayCreateVector(
VT_UI1,
txn_in.parray->rgsabound->cElements,
txn_in.parray->rgsabound->cElements);
        pData =
(COM_DATA*)txn_out->parray->pvData;

        m_pTxn->OrderStatus();

        memcpy( &pData->u.OrderStatus, pOrderStatus,
sizeof(ORDER_STATUS_DATA));

        pData->retval = ERR_SUCCESS;
        pData->error = 0;
        return S_OK;
    }
    catch (CBaseErr *e)
    {
        // check for lost database connection; if yes,
component is toast
        if ( ((e->ErrorType() == ERR_TYPE_DBLIB)
&& (e->ErrorNum() == 10005)) ||
        ((e->ErrorType() ==
ERR_TYPE_ODBC) && (e->ErrorNum() == 10054)) )
            m_bCanBePooled = FALSE;
    }
}

```

```

        {
            WriteMessageToEventLog(TEXT("Unhandled
exception."));
            pData->retval = ERR_TYPE_LOGIC;
            pData->error = 0;
            m_bCanBePooled = FALSE;
            return E_FAIL;
        }
    }
}

```

## **tpcc\_com\_remote.def**

: tpcc\_com\_remote.def : Declares the module parameters.

```
LIBRARY "tpcc_com_remote.dll"
```

```
EXPORTS
```

```

    DllCanUnloadNow @1 PRIVATE
    DllGetClassObject @2 PRIVATE
    DllRegisterServer @3 PRIVATE
    DllUnregisterServer @4 PRIVATE

```

## **tpcc\_com\_remote.h**

/\* this ALWAYS GENERATED file contains the definitions for the interfaces \*/

/\* File created by MIDL compiler version 5.01.0164 \*/

/\* at Fri Jan 19 11:15:15 2001

\*/

/\* Compiler settings for .\src\tpcc\_com\_remote.idl:  
 Oicf (OptLev=i2), W1, Zp8, env=Win32, ms\_ext, c\_ext  
 error checks: allocation ref bounds\_check enum stub\_data  
 \*/

//@@MIDL\_FILE\_HEADING( )

/\* verify that the <rpcndr.h> version is high enough to compile this file\*/

```

#ifdef __REQUIRED_RPCNDR_H_VERSION__
#define __REQUIRED_RPCNDR_H_VERSION__ 440
#endif

```

```

#include "rpc.h"
#include "rpcndr.h"

```

```

#ifdef __tpcc_com_remote_h__
#define __tpcc_com_remote_h__

```

```

#ifdef __cplusplus
extern "C"{
#endif

```

/\* Forward Declarations \*/

```

#ifdef __TPCCrem_FWD_DEFINED__
#define __TPCCrem_FWD_DEFINED__

```

```

#ifdef __cplusplus
typedef class TPCCrem TPCCrem;
#else
typedef struct TPCCrem TPCCrem;
#endif /* __cplusplus */

```

```
#endif /* __TPCCrem_FWD_DEFINED__ */
```

/\* header files for imported files \*/

```

#include "oidl.h"
#include "ocidl.h"
#include "tpcc_com_remote_ps.h"

```

```

void __RPC_FAR * __RPC_USER MIDL_user_allocate(size_t);
void __RPC_USER MIDL_user_free( void __RPC_FAR * );

```

```

/* interface __MIDL_itf_tpcc_com_remote_0000 */
/* [local] */

```

```

extern RPC_IF_HANDLE
__MIDL_itf_tpcc_com_remote_0000_v0_0_c_ifspec;
extern RPC_IF_HANDLE
__MIDL_itf_tpcc_com_remote_0000_v0_0_s_ifspec;

```

```

#ifdef __TPCCremLib_LIBRARY_DEFINED__
#define __TPCCremLib_LIBRARY_DEFINED__

```

```

/* library TPCCremLib */
/* [helpstring][version][uuid] */

```

```
EXTERN_C const IID LIBID_TPCCremLib;
```

```
EXTERN_C const CLSID CLSID_TPCCrem;
```

```
#ifdef __cplusplus
```

```

class
DECLSPEC_UUID("63EC25AB-828A-4ed7-8C6C-B46D29889594")
TPCCrem;
#endif
#endif /* __TPCCremLib_LIBRARY_DEFINED__ */

```

/\* Additional Prototypes for ALL interfaces \*/

/\* end of Additional Prototypes \*/

```

#ifdef __cplusplus
}
#endif

```

```
#endif
```

## **tpcc\_com\_remote.idl**

```

/* FILE: TPCC.IDL
 * Microsoft TPC-C Kit Ver.
4.20.000
 * Copyright Microsoft,
1999
 * All Rights Reserved
 * not yet audited
 *
 * PURPOSE: IDL source for TPCC.dll. This file is
processed by the MIDL tool to
 * produce the type library
(TPCC.tlb) and marshalling code.

```

```

*
* Change history:
*           4.20.000 - first version
*/

interface TPCCrem;

import "oidl.idl";
import "ocidl.idl";
import "..\tpcc_com_remote_ps\src\tpcc_com_remote_ps.idl";

[
    uuid(B207575F-7A88-489a-9383-859E2D771537),
    version(1.0),
    helpstring("Remote TPC-C 1.0 Type Library")
]
library TPCCremLib
{
    importlib("stdole32.tlb");
    importlib("stdole2.tlb");

    [
        uuid(63EC25AB-828A-4ed7-8C6C-B46D29889594),
        helpstring("Remote All Txns Class")
    ]
    coclass TPCCrem
    {
        [default] interface ITPCCrem;
    };
};

tpcc_com_remote.rc

//Microsoft Developer Studio generated resource script.
//
#include "resource.h"

#define APSTUDIO_READONLY_SYMBOLS
//
// Generated from the TEXTINCLUDE2 resource.
//
#include "winres.h"

//
// English (U.S.) resources

#if !defined(AFX_RESOURCE_DLL) || defined(AFX_TARG_ENU)
#ifndef _WIN32
LANGUAGE LANG_ENGLISH, SUBLANG_ENGLISH_US
#pragma code_page(1252)
#endif
#endif

#ifndef APSTUDIO_INVOKED
//
// TEXTINCLUDE
//

```

```

1 TEXTINCLUDE DISCARDABLE
BEGIN
    "resource.h\0"
END

2 TEXTINCLUDE DISCARDABLE
BEGIN
    "#include ""winres.h""\r\n"
    "\0"
END

3 TEXTINCLUDE DISCARDABLE
BEGIN
    "1 TYPELIB ""tpcc_com_remote.tlb""\r\n"
    "\0"
END

#endif // APSTUDIO_INVOKED

#ifdef _MAC
//
// Version
//
VS_VERSION_INFO VERSIONINFO
FILEVERSION 1,0,0,1
PRODUCTVERSION 1,0,0,1
FILEFLAGSMASK 0x3fL
#ifdef _DEBUG
FILEFLAGS 0x1L
#else
FILEFLAGS 0x0L
#endif
FILEOS 0x4L
FILETYPE 0x2L
FILESUBTYPE 0x0L
BEGIN
    BLOCK "StringFileInfo"
    BEGIN
        BLOCK "040904B0"
        BEGIN
            VALUE "CompanyName", "\0"
            VALUE "FileDescription", "tpcc_com_remote Module\0"
            VALUE "FileVersion", "1, 0, 0, 1\0"
            VALUE "InternalName", "TPCCNEWORDER\0"
            VALUE "LegalCopyright", "Copyright 1997\0"
            VALUE "OriginalFilename", "tpcc_com_remote.DLL\0"
            VALUE "ProductName", "tpcc_com_remote Module\0"
            VALUE "ProductVersion", "1, 0, 0, 1\0"
            VALUE "OLESelfRegister", "\0"
        END
    END
    BLOCK "VarFileInfo"
    BEGIN
        VALUE "Translation", 0x409, 1200
    END
END

#endif // !_MAC

//
// REGISTRY
//

```

```
IDR_TPCCrem          REGISTRY DISCARDABLE
"tpcc_com_remote.rgs"
```

```
////////////////////////////////////
//
// String Table
//
```

```
STRINGTABLEDISCARDABLE
BEGIN
    IDS_PROJNAME          "tpcc_com_remote"
END
```

```
#endif // English (U.S.) resources
////////////////////////////////////
```

```
#ifndef APSTUDIO_INVOKED
////////////////////////////////////
//
// Generated from the TEXTINCLUDE3 resource.
//
1 TYPELIB "tpcc_com_remote.tlb"
```

```
////////////////////////////////////
#endif // not APSTUDIO_INVOKED
```

### ***tpcc\_com\_remote.rgs***

```
HKCR
{
    TPCC.AllRemoteTxns.1 = s 'All RemoteTxns Class'
    {
        CLSID = s
        '{63EC25AB-828A-4ed7-8C6C-B46D29889594}'
    }
    TPCC.AllRemoteTxns = s 'TPCC Remote Class'
    {
        CurVer = s 'TPCC.AllRemoteTxns.1'
    }
    NoRemove CLSID
    {
        ForceRemove
        {63EC25AB-828A-4ed7-8C6C-B46D29889594} = s 'TPCC Remote
        Class'
    }
    Progid = s 'TPCC.AllRemoteTxns.1'
    VersionIndependentProgid = s
    'TPCC.AllRemoteTxns'
    InprocServer32 = s '%MODULE%'
    {
        val ThreadingModel = s
        'Both'
    }
}
}
```

### ***tpcc\_com\_remote.i.c***

```
/* this file contains the actual definitions of */
/* the IIDs and CLSIDs */
```

```
/* link this file in with the server and any clients */
```

```
/* File created by MIDL compiler version 5.01.0164 */
/* at Fri Jan 19 11:15:15 2001
*/
/* Compiler settings for .\src\tpcc_com_remote.idl:
    Oicf (OptLev=i2), W1, Zp8, env=Win32, ms_ext, c_ext
    error checks: allocation ref bounds_check enum stub_data
*/
```

```
//@@MIDL_FILE_HEADING( )
#ifdef __cplusplus
extern "C"{
#endif
```

```
#ifndef __IID_DEFINED__
#define __IID_DEFINED__
```

```
typedef struct _IID
{
    unsigned long x;
    unsigned short s1;
    unsigned short s2;
    unsigned char c[8];
} IID;
```

```
#endif // __IID_DEFINED__
```

```
#ifndef CLSID_DEFINED
#define CLSID_DEFINED
typedef IID CLSID;
#endif // CLSID_DEFINED
```

```
const IID LIBID_TPCCremLib =
{0xB207575F,0x7A88,0x489a,{0x93,0x83,0x85,0x9E,0x2D,0x77,0x1
5,0x37}};
```

```
const CLSID CLSID_TPCCrem =
{0x63EC25AB,0x828A,0x4ed7,{0x8C,0x6C,0xB4,0x6D,0x29,0x88,0
x95,0x94}};
```

```
#ifdef __cplusplus
}
#endif
```

### ***tpcc\_com\_remote\_Methods.h***

```
/* FILE: METHODS.H
* Microsoft TPC-C Kit Ver.
4.20.000
* Copyright Microsoft,
1999
* All Rights Reserved
* not yet audited
* PURPOSE: Header file for COM components.
* Change history:
* 4.20.000 - first version
*/
```

```
enum COMPONENT_ERROR
```

```

{
    ERR_MISSING_REGISTRY_ENTRIES= 1,
    ERR_LOADDLL_FAILED,
    ERR_GETPROCADDR_FAILED,
    ERR_UNKNOWN_DB_PROTOCOL
};

class CCOMPONENT_ERR : public CBaseErr
{
public:
    CCOMPONENT_ERR(COMPONENT_ERROR
Err)
    {
        m_Error = Err;
        m_szTextDetail = NULL;
        m_SystemErr = 0;
        m_szErrorText = NULL;
    };

    CCOMPONENT_ERR(COMPONENT_ERROR
Err, char *szTextDetail, DWORD dwSystemErr)
    {
        m_Error = Err;
        m_szTextDetail = new
char[strlen(szTextDetail)+1];
        strcpy( m_szTextDetail, szTextDetail
);

        m_SystemErr = dwSystemErr;
        m_szErrorText = NULL;
    };

    ~CCOMPONENT_ERR()
    {
        if (m_szTextDetail != NULL)
            delete [] m_szTextDetail;
        if (m_szErrorText != NULL)
            delete [] m_szErrorText;
    };

    COMPONENT_ERROR      m_Error;
    char                  *m_szTextDetail;
    char                  *m_szErrorText;
    DWORD                 m_SystemErr;

    int ErrorType() {return
ERR_TYPE_COMPONENT;};
    int ErrorNum() {return m_Error;};
    char *ErrorText();
};

static void WriteMessageToEventLog(LPTSTR lpszMsg);

////////////////////////////////////
// CTPCC_Common
class CTPCC_Common_Remote :
public ITPCCrem,
public IObjectControl,
public IObjectConstruct,
public CComObjectRootEx<CComSingleThreadModel>
{
public:
    BEGIN_COM_MAP(CTPCC_Common_Remote)
        COM_INTERFACE_ENTRY(ITPCCrem)
        COM_INTERFACE_ENTRY(IObjectControl)

```

```

        COM_INTERFACE_ENTRY(IObjectConstruct)
    END_COM_MAP()

    CTPCC_Common_Remote();
    ~CTPCC_Common_Remote();

// ITPCCrem
public:
    HRESULT __stdcall NewOrder(          VARIANT
    txn_in, VARIANT* txn_out);
    HRESULT __stdcall Payment(          VARIANT
    txn_in, VARIANT* txn_out);
    HRESULT __stdcall Delivery(         VARIANT
    txn_in) {return E_NOTIMPL;};
    HRESULT __stdcall StockLevel(VARIANT txn_in,
    VARIANT* txn_out);
    HRESULT __stdcall OrderStatus(      VARIANT
    txn_in, VARIANT* txn_out);

    HRESULT __stdcall CallSetComplete();

// IObjectControl
    STDMETHODIMP_(BOOL) CanBePooled() { return
m_bCanBePooled; }
    STDMETHODIMP Activate() { return S_OK; } //
we don't support COM Services transactions (no enlistment)
    STDMETHODIMP_(void) Deactivate() { /* nothing to do
*/ }

// IObjectConstruct
    STDMETHODIMP Construct(IDispatch * pUnk);

// helper methods
private:
    BOOL      m_bCanBePooled;
    CTPCC_BASE *m_pTxn;

    struct COM_DATA
    {
        int retVal;
        int error;
        union
        {
            NEW_ORDER_DATA
            PAYMENT_DATA
            DELIVERY_DATA
            STOCK_LEVEL_DATA
            ORDER_STATUS_DATA
        } u;
    };
};

////////////////////////////////////
// CTPCC
class CTPCCrem :
public CTPCC_Common_Remote,
public CComCoClass<CTPCCrem, &CLSID_TPCCrem>
{
public:
    DECLARE_REGISTRY_RESOURCEID(IDR_TPCCrem)
    BEGIN_COM_MAP(CTPCCrem)

```

```

        COM_INTERFACE_ENTRY2(IUnknown,
        CComObjectRootEx)

COM_INTERFACE_ENTRY_CHAIN(CTPCC_Common_Remote)
END_COM_MAP()

};

```

### **tpcc\_com\_remote\_ps.def**

```
LIBRARY "tpcc_com_remote_ps"
```

```
DESCRIPTION 'Proxy/Stub DLL'
```

```
EXPORTS
```

```

    DllGetClassObject    @1    PRIVATE
    DllCanUnloadNow     @2    PRIVATE
    GetProxyDllInfo     @3    PRIVATE
    DllRegisterServer   @4    PRIVATE
    DllUnregisterServer @5    PRIVATE

```

### **tpcc\_com\_remote\_ps.h**

```
/* this ALWAYS GENERATED file contains the definitions for the
interfaces */
```

```

/* File created by MIDL compiler version 5.01.0164 */
/* at Fri Jan 19 11:15:07 2001
*/
/* Compiler settings for .\src\tpcc_com_remote_ps.idl:
    Oicf (OptLev=i2), W1, Zp8, env=Win32, ms_ext, c_ext
    error checks: allocation ref bounds_check enum stub_data
*/
//@@@MIDL_FILE_HEADING( )

```

```
/* verify that the <rpcndr.h> version is high enough to compile this
file*/
```

```

#ifndef __REQUIRED_RPCNDR_H_VERSION__
#define __REQUIRED_RPCNDR_H_VERSION__ 440
#endif

```

```

#include "rpc.h"
#include "rpcndr.h"

```

```

#ifndef __RPCNDR_H_VERSION__
#error this stub requires an updated version of <rpcndr.h>
#endif // __RPCNDR_H_VERSION__

```

```

#ifndef COM_NO_WINDOWS_H
#include "windows.h"
#include "ole2.h"
#endif /*COM_NO_WINDOWS_H*/

```

```

#ifndef __tpcc_com_remote_ps_h__
#define __tpcc_com_remote_ps_h__

```

```

#ifdef __cplusplus
extern "C"{
#endif

```

```
/* Forward Declarations */
```

```

#ifndef __ITPCCrem_FWD_DEFINED__
#define __ITPCCrem_FWD_DEFINED__

```

```

typedef interface ITPCCrem ITPCCrem;
#endif /* __ITPCCrem_FWD_DEFINED__ */

```

```

/* header files for imported files */
#include "oidl.h"
#include "ocidl.h"

```

```

void __RPC_FAR * __RPC_USER MIDL_user_allocate(size_t);
void __RPC_USER MIDL_user_free( void __RPC_FAR * );

```

```

/* interface __MIDL_itf_tpcc_com_remote_ps_0000 */
/* [local] */

```

```

extern RPC_IF_HANDLE
__MIDL_itf_tpcc_com_remote_ps_0000_v0_0_c_ifspec;
extern RPC_IF_HANDLE
__MIDL_itf_tpcc_com_remote_ps_0000_v0_0_s_ifspec;

```

```

#ifndef __ITPCCrem_INTERFACE_DEFINED__
#define __ITPCCrem_INTERFACE_DEFINED__

```

```

/* interface ITPCCrem */
/* [unique][helpstring][uuid][oleautomation][object] */

```

```
EXTERN_C const IID IID_ITPCCrem;
```

```
#if defined(__cplusplus) && !defined(CINTERFACE)
```

```
MIDL_INTERFACE("4899AD2E-D521-4d59-824A-4381438FD840")
```

```

ITPCCrem : public IUnknown
{
public:
    virtual HRESULT __stdcall NewOrder(
        /* [in] */ VARIANT txn_in,
        /* [out] */ VARIANT __RPC_FAR *txn_out) = 0;

    virtual HRESULT __stdcall Payment(
        /* [in] */ VARIANT txn_in,
        /* [out] */ VARIANT __RPC_FAR *txn_out) = 0;

    virtual HRESULT __stdcall Delivery(
        /* [in] */ VARIANT txn_in) = 0;

    virtual HRESULT __stdcall StockLevel(
        /* [in] */ VARIANT txn_in,
        /* [out] */ VARIANT __RPC_FAR *txn_out) = 0;

    virtual HRESULT __stdcall OrderStatus(
        /* [in] */ VARIANT txn_in,
        /* [out] */ VARIANT __RPC_FAR *txn_out) = 0;

    virtual HRESULT __stdcall CallSetComplete( void) = 0;
};

```

```
#else /* C style interface */
```

```

typedef struct ITPCCremVtbl
{
    BEGIN_INTERFACE

```



```

HRESULT ( STDMETHODCALLTYPE __RPC_FAR
*QueryInterface)(
    ITPCCrem __RPC_FAR * This,
    /* [in] */ REFIID riid,
    /* [iid_is][out] */ void __RPC_FAR * __RPC_FAR
*ppvObject);

ULONG ( STDMETHODCALLTYPE __RPC_FAR *AddRef)(
    ITPCCrem __RPC_FAR * This);

ULONG ( STDMETHODCALLTYPE __RPC_FAR *Release)(
    ITPCCrem __RPC_FAR * This);

HRESULT ( _stdcall __RPC_FAR *NewOrder)(
    ITPCCrem __RPC_FAR * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT __RPC_FAR *txn_out);

HRESULT ( _stdcall __RPC_FAR *Payment)(
    ITPCCrem __RPC_FAR * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT __RPC_FAR *txn_out);

HRESULT ( _stdcall __RPC_FAR *Delivery)(
    ITPCCrem __RPC_FAR * This,
    /* [in] */ VARIANT txn_in);

HRESULT ( _stdcall __RPC_FAR *StockLevel)(
    ITPCCrem __RPC_FAR * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT __RPC_FAR *txn_out);

HRESULT ( _stdcall __RPC_FAR *OrderStatus)(
    ITPCCrem __RPC_FAR * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT __RPC_FAR *txn_out);

HRESULT ( _stdcall __RPC_FAR *CallSetComplete)(
    ITPCCrem __RPC_FAR * This);

END_INTERFACE
} ITPCCremVtbl;

interface ITPCCrem
{
    CONST_VTBL struct ITPCCremVtbl __RPC_FAR *IpVtbl;
};

#ifdef COBJMACROS

#define ITPCCrem_QueryInterface(This,riid,ppvObject) \
    (This)->IpVtbl -> QueryInterface(This,riid,ppvObject)

#define ITPCCrem_AddRef(This) \
    (This)->IpVtbl -> AddRef(This)

#define ITPCCrem_Release(This) \
    (This)->IpVtbl -> Release(This)

#define ITPCCrem_NewOrder(This,txn_in,txn_out) \
    (This)->IpVtbl -> NewOrder(This,txn_in,txn_out)

#define ITPCCrem_Payment(This,txn_in,txn_out) \
    (This)->IpVtbl -> Payment(This,txn_in,txn_out)

```

```

#define ITPCCrem_Delivery(This,txn_in) \
    (This)->IpVtbl -> Delivery(This,txn_in)

#define ITPCCrem_StockLevel(This,txn_in,txn_out) \
    (This)->IpVtbl -> StockLevel(This,txn_in,txn_out)

#define ITPCCrem_OrderStatus(This,txn_in,txn_out) \
    (This)->IpVtbl -> OrderStatus(This,txn_in,txn_out)

#define ITPCCrem_CallSetComplete(This) \
    (This)->IpVtbl -> CallSetComplete(This)

#endif /* COBJMACROS */

#ifdef /* C style interface */

HRESULT _stdcall ITPCCrem_NewOrder_Proxy(
    ITPCCrem __RPC_FAR * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT __RPC_FAR *txn_out);

void __RPC_STUB ITPCCrem_NewOrder_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer * _pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,
    DWORD *_pdwStubPhase);

HRESULT _stdcall ITPCCrem_Payment_Proxy(
    ITPCCrem __RPC_FAR * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT __RPC_FAR *txn_out);

void __RPC_STUB ITPCCrem_Payment_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer * _pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,
    DWORD *_pdwStubPhase);

HRESULT _stdcall ITPCCrem_Delivery_Proxy(
    ITPCCrem __RPC_FAR * This,
    /* [in] */ VARIANT txn_in);

void __RPC_STUB ITPCCrem_Delivery_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer * _pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,
    DWORD *_pdwStubPhase);

HRESULT _stdcall ITPCCrem_StockLevel_Proxy(
    ITPCCrem __RPC_FAR * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT __RPC_FAR *txn_out);

void __RPC_STUB ITPCCrem_StockLevel_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer * _pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,

```

```

DWORD *_pdwStubPhase);

HRESULT __stdcall ITPCCrem_OrderStatus_Proxy(
    ITPCCrem __RPC_FAR * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT __RPC_FAR *txn_out);

void __RPC_STUB ITPCCrem_OrderStatus_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *_pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,
    DWORD *_pdwStubPhase);

HRESULT __stdcall ITPCCrem_CallSetComplete_Proxy(
    ITPCCrem __RPC_FAR * This);

void __RPC_STUB ITPCCrem_CallSetComplete_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *_pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,
    DWORD *_pdwStubPhase);

#endif /* __ITPCCrem_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 * );
void __RPC_USER VARIANT_UserFree( unsigned
long __RPC_FAR *, VARIANT __RPC_FAR * );

/* end of Additional Prototypes */

#ifdef __cplusplus
}
#endif

#endif

tpcc_com_remote_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 ITPCCrem;
import "oidl.idl";
import "ocidl.idl";

[
    object,
    oleautomation,
    uuid(4899AD2E-D521-4d59-824A-4381438FD840),
    helpstring("ITPCC Interface"),
    pointer_default(unique)
]
interface ITPCCrem : IUnknown
{
    HRESULT __stdcall NewOrder
    (
    [in]
    VARIANT txn_in,
    [out] VARIANT *txn_out
    );

    HRESULT __stdcall Payment
    (
    [in]
    VARIANT txn_in,
    [out] VARIANT *txn_out
    );

    HRESULT __stdcall Delivery
    (
    [in]
    VARIANT txn_in
    );

    HRESULT __stdcall StockLevel
    (
    [in]
    VARIANT txn_in,
    [out] VARIANT *txn_out
    );

    HRESULT __stdcall OrderStatus
    (
    [in]
    VARIANT txn_in,
    [out] VARIANT *txn_out
    );

    HRESULT __stdcall CallSetComplete
    (
    );

}; // interface ITPCC

tpcc_com_remote_ps_i.c

```

```

/* this file contains the actual definitions of */
/* the IIDs and CLSIDs */

/* link this file in with the server and any clients */

/* File created by MIDL compiler version 5.01.0164 */
/* at Fri Jan 19 11:15:07 2001
*/
/* Compiler settings for .\src\tpcc_com_remote_ps.idl:
Oicf (OptLev=i2), W1, Zp8, env=Win32, ms_ext, c_ext
error checks: allocation ref bounds_check enum stub_data
*/
//@@@MIDL_FILE_HEADING( )
#ifdef __cplusplus
extern "C"{
#endif

#ifdef __IID_DEFINED__
#define __IID_DEFINED__

typedef struct _IID
{
    unsigned long x;
    unsigned short s1;
    unsigned short s2;
    unsigned char c[8];
} IID;

#endif // __IID_DEFINED__

#ifdef CLSID_DEFINED
#define CLSID_DEFINED
typedef IID CLSID;
#endif // CLSID_DEFINED

const IID IID_ITPCCrem =
{0x4899AD2E,0xD521,0x4d59,{0x82,0x4A,0x43,0x81,0x43,0x8F,0x
D8,0x40}};

#ifdef __cplusplus
}
#endif

tpcc_com_remote_ps_p.c

/* this ALWAYS GENERATED file contains the proxy stub code */

/* File created by MIDL compiler version 5.01.0164 */
/* at Fri Jan 19 11:15:07 2001
*/
/* Compiler settings for .\src\tpcc_com_remote_ps.idl:
Oicf (OptLev=i2), W1, Zp8, env=Win32, ms_ext, c_ext
error checks: allocation ref bounds_check enum stub_data
*/
//@@@MIDL_FILE_HEADING( )

#define USE_STUBLESS_PROXY

/* verify that the <rpcproxy.h> version is high enough to compile this
file*/
#endif __REDQ_RPCPROXY_H_VERSION__

```

```

#define __REQUIRED_RPCPROXY_H_VERSION__ 440
#endif

#include "rpcproxy.h"
#ifdef __RPCPROXY_H_VERSION__
#error this stub requires an updated version of <rpcproxy.h>
#endif // __RPCPROXY_H_VERSION__

#include "tpcc_com_remote_ps.h"

#define TYPE_FORMAT_STRING_SIZE 997
#define PROC_FORMAT_STRING_SIZE 187

typedef struct _MIDL_TYPE_FORMAT_STRING
{
    short Pad;
    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_remote_ps_0000, ver.
0.0,
GUID={0x00000000,0x0000,0x0000,{0x00,0x00,0x00,0x00,0x00,0x
00,0x00,0x00}} */

/* Object interface: IUnknown, ver. 0.0,
GUID={0x00000000,0x0000,0x0000,{0xC0,0x00,0x00,0x00,0x00,0x
00,0x00,0x46}} */

/* Object interface: ITPCCrem, ver. 0.0,
GUID={0x4899AD2E,0xD521,0x4d59,{0x82,0x4A,0x43,0x81,0x43,0
x8F,0xD8,0x40}} */

extern const MIDL_STUB_DESC Object_StubDesc;

extern const MIDL_SERVER_INFO ITPCCrem_ServerInfo;

#pragma code_seg(".orpc")
extern const USER_MARSHAL_ROUTINE_QUADRUPLE
UserMarshalRoutines[1];

static const MIDL_STUB_DESC Object_StubDesc =
{
    0,
    NdrOleAllocate,
    NdrOleFree,
    0,

```

```

0,
0,
0,
0,
__MIDL_TypeFormatString.Format,
1, /* -error bounds_check flag */
0x20000, /* Ndr library version */
0,
0x50100a4, /* MIDL Version 5.1.164 */
0,
UserMarshalRoutines,
0, /* notify & notify_flag routine table */
1, /* Flags */
0, /* Reserved3 */
0, /* Reserved4 */
0 /* Reserved5 */
};

static const unsigned short ITPCCrem_FormatStringOffsetTable[] =
{
0,
34,
68,
96,
130,
164
};

static const MIDL_SERVER_INFO ITPCCrem_ServerInfo =
{
&Object_StubDesc,
0,
__MIDL_ProcFormatString.Format,
&ITPCCrem_FormatStringOffsetTable[-3],
0,
0,
0,
0
};

static const MIDL_STUBLESS_PROXY_INFO ITPCCrem_ProxyInfo =
{
&Object_StubDesc,
__MIDL_ProcFormatString.Format,
&ITPCCrem_FormatStringOffsetTable[-3],
0,
0,
0
};

CINTERFACE_PROXY_VTABLE(9)_ITPCCremProxyVtbl =
{
&ITPCCrem_ProxyInfo,
&IID_ITPCCrem,
IUnknown_QueryInterface_Proxy,
IUnknown_AddRef_Proxy,
IUnknown_Release_Proxy,
(void *)-1 /* ITPCCrem::NewOrder */,
(void *)-1 /* ITPCCrem::Payment */,
(void *)-1 /* ITPCCrem::Delivery */,
(void *)-1 /* ITPCCrem::StockLevel */,
(void *)-1 /* ITPCCrem::OrderStatus */,
(void *)-1 /* ITPCCrem::CallSetComplete */
};

const CInterfaceStubVtbl _ITPCCremStubVtbl =
{

```

```

&IID_ITPCCrem,
&ITPCCrem_ServerInfo,
9,
0, /* pure interpreted */
CStdStubBuffer_METHODS
};

#pragma data_seg("rdata")

static const USER_MARSHAL_ROUTINE_QUADRUPLE
UserMarshalRoutines[1] =
{
{
VARIANT_UserSize,
VARIANT_UserMarshal,
VARIANT_UserUnmarshal,
VARIANT_UserFree
}
};

#ifndef __RPC_WIN32__
#error Invalid build platform for this stub.
#endif

#ifndef TARGET_IS_NT40_OR_LATER
#error You need a Windows NT 4.0 or later to run this stub because it
uses these features:
#error -Oif or -Oicf, [wire_marshall] or [user_marshall] attribute, more
than 32 methods in the interface.
#error However, your C/C++ compilation flags indicate you intend to
run this app on earlier systems.
#error This app will die there with the
RPC_X_WRONG_STUB_VERSION error.
#endif

static const MIDL_PROC_FORMAT_STRING
__MIDL_ProcFormatString =
{
0,
{
/* Procedure NewOrder */
FC_AUTO_HANDLE /* 0x33, */ /*
object, Oi2 */ /* 0x6c, */ /* Old Flags:
/* 2 */ NdrFcLong( 0x0 ), /* 0 */
/* 6 */ NdrFcShort( 0x3 ), /* 3 */
#ifdef _ALPHA_
/* 8 */ NdrFcShort( 0x1c ), /* x86 Stack size/offset = 28 */
#else
NdrFcShort( 0x20 ), /* MIPS &
PPC Stack size/offset = 32 */
#endif
#else
NdrFcShort( 0x28 ), /* Alpha Stack
size/offset = 40 */
#endif
/* 10 */ NdrFcShort( 0x0 ), /* 0 */
/* 12 */ NdrFcShort( 0x8 ), /* 8 */
/* 14 */ 0x7, /* Oi2 Flags: srv must size, clt must
size, has return, */

```

```

                                0x3,                /* 3 */
                                /* Parameter txn_in */
/* 16 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by
val, */
#ifndef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 18 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
#else
                                NdrFcShort( 0x8 ), /* MIPS &
PPC Stack size/offset = 8 */
#endif
#else
                                NdrFcShort( 0x8 ), /* Alpha Stack
size/offset = 8 */
#endif
/* 20 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */

                                /* Parameter txn_out */

/* 22 */ NdrFcShort( 0x4113 ), /* Flags: must size, must
free, out, simple ref, srv alloc size=16 */
#ifndef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 24 */ NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
#else
                                NdrFcShort( 0x18 ), /* MIPS &
PPC Stack size/offset = 24 */
#endif
#else
                                NdrFcShort( 0x18 ), /* Alpha Stack
size/offset = 24 */
#endif
/* 26 */ NdrFcShort( 0x3da ), /* Type Offset=986 */

                                /* Return value */

/* 28 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifndef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 30 */ NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
#else
                                NdrFcShort( 0x1c ), /* MIPS &
PPC Stack size/offset = 28 */
#endif
#else
                                NdrFcShort( 0x20 ), /* Alpha Stack
size/offset = 32 */
#endif
/* 32 */ 0x8, /* FC_LONG */
                                0x0, /* 0 */

                                /* Procedure Payment */

/* 34 */ 0x33, /* FC_AUTO_HANDLE */
                                0x6c, /* Old Flags:
object, Oi2 */
/* 36 */ NdrFcLong( 0x0 ), /* 0 */
/* 40 */ NdrFcShort( 0x4 ), /* 4 */
#ifndef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 42 */ NdrFcShort( 0x1c ), /* x86 Stack size/offset = 28 */
#else
                                NdrFcShort( 0x20 ), /* MIPS &
PPC Stack size/offset = 32 */
#endif
#else
                                NdrFcShort( 0x20 ), /* Alpha Stack
size/offset = 40 */
#endif
/* 44 */ NdrFcShort( 0x0 ), /* 0 */
/* 46 */ NdrFcShort( 0x8 ), /* 8 */
/* 48 */ 0x7, /* Oi2 Flags: srv must size, clt must
size, has return, */
                                0x3, /* 3 */

                                /* Parameter txn_in */

/* 50 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by
val, */
#ifndef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 52 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
#else
                                NdrFcShort( 0x8 ), /* MIPS &
PPC Stack size/offset = 8 */
#endif
#else
                                NdrFcShort( 0x8 ), /* Alpha Stack
size/offset = 8 */
#endif
/* 54 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */

                                /* Parameter txn_out */

/* 56 */ NdrFcShort( 0x4113 ), /* Flags: must size, must
free, out, simple ref, srv alloc size=16 */
#ifndef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 58 */ NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
#else
                                NdrFcShort( 0x18 ), /* MIPS &
PPC Stack size/offset = 24 */
#endif
#else
                                NdrFcShort( 0x18 ), /* Alpha Stack
size/offset = 24 */
#endif
/* 60 */ NdrFcShort( 0x3da ), /* Type Offset=986 */

                                /* Return value */

/* 62 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifndef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 64 */ NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
#else
                                NdrFcShort( 0x1c ), /* MIPS &
PPC Stack size/offset = 28 */
#endif
#else
                                NdrFcShort( 0x20 ), /* Alpha Stack
size/offset = 32 */
#endif
/* 66 */ 0x8, /* FC_LONG */
                                0x0, /* 0 */

                                /* Procedure Delivery */

/* 68 */ 0x33, /* FC_AUTO_HANDLE */
                                0x6c, /* Old Flags:
object, Oi2 */
/* 70 */ NdrFcLong( 0x0 ), /* 0 */
/* 74 */ NdrFcShort( 0x5 ), /* 5 */
#ifndef _ALPHA_

```

```

#if !defined(_MIPS_) && !defined(_PPC_)
/* 76 */ NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
#else
NdrFcShort( 0x1c ), /* MIPS &
PPC Stack size/offset = 28 */
#endif
#else
NdrFcShort( 0x20 ), /* Alpha Stack
size/offset = 32 */
#endif
/* 78 */ NdrFcShort( 0x0 ), /* 0 */
/* 80 */ NdrFcShort( 0x8 ), /* 8 */
/* 82 */ 0x6, /* Oi2 Flags: clt must size, has
return, */
0x2, /* 2 */

/* Parameter txn_in */

/* 84 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by
val, */
#ifdef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 86 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
#else
NdrFcShort( 0x8 ), /* MIPS &
PPC Stack size/offset = 8 */
#endif
#else
NdrFcShort( 0x8 ), /* Alpha Stack
size/offset = 8 */
#endif
/* 88 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */

/* Return value */

/* 90 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifdef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 92 */ NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
#else
NdrFcShort( 0x18 ), /* MIPS &
PPC Stack size/offset = 24 */
#endif
#else
NdrFcShort( 0x18 ), /* Alpha Stack
size/offset = 24 */
#endif
/* 94 */ 0x8, /* FC_LONG */
0x0, /* 0 */

/* Procedure StockLevel */

/* 96 */ 0x33, /* FC_AUTO_HANDLE */
0x6c, /* Old Flags:
object, Oi2 */
/* 98 */ NdrFcLong( 0x0 ), /* 0 */
/* 102 */ NdrFcShort( 0x6 ), /* 6 */
#ifdef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 104 */ NdrFcShort( 0x1c ), /* x86 Stack size/offset = 28 */
#else
NdrFcShort( 0x20 ), /* MIPS &
PPC Stack size/offset = 32 */
#endif
#else
NdrFcShort( 0x28 ), /* Alpha Stack
size/offset = 40 */
#endif
/* 106 */ NdrFcShort( 0x0 ), /* 0 */
/* 108 */ NdrFcShort( 0x8 ), /* 8 */
/* 110 */ 0x7, /* Oi2 Flags: srv must size, clt must
size, has return, */
0x3, /* 3 */

/* Parameter txn_in */

/* 112 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by
val, */
#ifdef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 114 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
#else
NdrFcShort( 0x8 ), /* MIPS &
PPC Stack size/offset = 8 */
#endif
#else
NdrFcShort( 0x8 ), /* Alpha Stack
size/offset = 8 */
#endif
/* 116 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */

/* Parameter txn_out */

/* 118 */ NdrFcShort( 0x4113 ), /* Flags: must size, must
free, out, simple ref, srv alloc size=16 */
#ifdef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 120 */ NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
#else
NdrFcShort( 0x18 ), /* MIPS &
PPC Stack size/offset = 24 */
#endif
#else
NdrFcShort( 0x18 ), /* Alpha Stack
size/offset = 24 */
#endif
/* 122 */ NdrFcShort( 0x3da ), /* Type Offset=986 */

/* Return value */

/* 124 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifdef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 126 */ NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
#else
NdrFcShort( 0x1c ), /* MIPS &
PPC Stack size/offset = 28 */
#endif
#else
NdrFcShort( 0x20 ), /* Alpha Stack
size/offset = 32 */
#endif
/* 128 */ 0x8, /* FC_LONG */
0x0, /* 0 */

/* Procedure OrderStatus */

/* 130 */ 0x33, /* FC_AUTO_HANDLE */
0x6c, /* Old Flags:
object, Oi2 */
/* 132 */ NdrFcLong( 0x0 ), /* 0 */
/* 136 */ NdrFcShort( 0x7 ), /* 7 */
#ifdef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 138 */ NdrFcShort( 0x1c ), /* x86 Stack size/offset = 28 */
#endif
#endif

```

```

NdrFcShort( 0x20 ), /* MIPS &
PPC Stack size/offset = 32 */
#endif
#else
NdrFcShort( 0x28 ), /* Alpha Stack
size/offset = 40 */
#endif
/* 140 */ NdrFcShort( 0x0 ), /* 0 */
/* 142 */ NdrFcShort( 0x8 ), /* 8 */
/* 144 */ 0x7, /* Oi2 Flags: srv must size, clt must
size, has return, */
0x3, /* 3 */
/* Parameter txn_in */
/* 146 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by
val, */
#ifndef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 148 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
#else
NdrFcShort( 0x8 ), /* MIPS &
PPC Stack size/offset = 8 */
#endif
#else
NdrFcShort( 0x8 ), /* Alpha Stack
size/offset = 8 */
#endif
/* 150 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */
/* Parameter txn_out */
/* 152 */ NdrFcShort( 0x4113 ), /* Flags: must size, must
free, out, simple ref, srv alloc size=16 */
#ifndef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 154 */ NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
#else
NdrFcShort( 0x18 ), /* MIPS &
PPC Stack size/offset = 24 */
#endif
#else
NdrFcShort( 0x18 ), /* Alpha Stack
size/offset = 24 */
#endif
/* 156 */ NdrFcShort( 0x3da ), /* Type Offset=986 */
/* Return value */
/* 158 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifndef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 160 */ NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
#else
NdrFcShort( 0x1c ), /* MIPS &
PPC Stack size/offset = 28 */
#endif
#else
NdrFcShort( 0x20 ), /* Alpha Stack
size/offset = 32 */
#endif
/* 162 */ 0x8, /* FC_LONG */
0x0, /* 0 */
/* Procedure CallSetComplete */
/* 164 */ 0x33, /* FC_AUTO_HANDLE */
object, Oi2 */
/* 166 */ NdrFcLong( 0x0 ), /* 0 */
/* 170 */ NdrFcShort( 0x8 ), /* 8 */
#ifndef _ALPHA_
/* 172 */ NdrFcShort( 0x8 ), /* x86, MIPS, PPC Stack size/offset
= 8 */
#else
NdrFcShort( 0x10 ), /* Alpha Stack
size/offset = 16 */
#endif
/* 174 */ NdrFcShort( 0x0 ), /* 0 */
/* 176 */ NdrFcShort( 0x8 ), /* 8 */
/* 178 */ 0x4, /* Oi2 Flags: has return, */
0x1, /* 1 */
/* Return value */
/* 180 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifndef _ALPHA_
/* 182 */ NdrFcShort( 0x4 ), /* x86, MIPS, PPC Stack size/offset
= 4 */
#else
NdrFcShort( 0x8 ), /* Alpha Stack
size/offset = 8 */
#endif
/* 184 */ 0x8, /* FC_LONG */
0x0, /* 0 */
0x0
}
};
static const MIDL_TYPE_FORMAT_STRING
_MIDL_TypeFormatString =
{
0,
{
NdrFcShort( 0x0 ), /* 0 */
/* 2 */
0x12, 0x0, /* FC_UP */
/* 4 */ NdrFcShort( 0x3b0 ), /* Offset= 944 (948) */
/* 6 */
0x2b, /*
FC_NON_ENCAPSULATED_UNION */
0x9, /* FC_ULONG
*/
/* 8 */ 0x7, /* Corr desc: FC_USHORT */
0x0, /*
*/
/* 10 */ NdrFcShort( 0xffff8 ), /* -8 */
/* 12 */ NdrFcShort( 0x2 ), /* Offset= 2 (14) */
/* 14 */ NdrFcShort( 0x10 ), /* 16 */
/* 16 */ NdrFcShort( 0x2b ), /* 43 */
/* 18 */ NdrFcLong( 0x3 ), /* 3 */
/* 22 */ NdrFcShort( 0x8008 ), /* Simple arm type:
FC_LONG */
/* 24 */ NdrFcLong( 0x11 ), /* 17 */
/* 28 */ NdrFcShort( 0x8001 ), /* Simple arm type:
FC_BYTE */
/* 30 */ NdrFcLong( 0x2 ), /* 2 */
/* 34 */ NdrFcShort( 0x8006 ), /* Simple arm type:
FC_SHORT */
/* 36 */ NdrFcLong( 0x4 ), /* 4 */
/* 40 */ NdrFcShort( 0x800a ), /* Simple arm type:
FC_FLOAT */
/* 42 */ NdrFcLong( 0x5 ), /* 5 */
/* 46 */ NdrFcShort( 0x800c ), /* Simple arm type:
FC_DOUBLE */
}
}

```

```

/* 48 */ NdrFcLong( 0xb ), /* 11 */
/* 52 */ NdrFcShort( 0x8006 ), /* Simple arm type:
FC_SHORT */
/* 54 */ NdrFcLong( 0xa ), /* 10 */
/* 58 */ NdrFcShort( 0x8008 ), /* Simple arm type:
FC_LONG */
/* 60 */ NdrFcLong( 0x6 ), /* 6 */
/* 64 */ NdrFcShort( 0xd6 ), /* Offset= 214 (278) */
/* 66 */ NdrFcLong( 0x7 ), /* 7 */
/* 70 */ NdrFcShort( 0x800c ), /* Simple arm type:
FC_DOUBLE */
/* 72 */ NdrFcLong( 0x8 ), /* 8 */
/* 76 */ NdrFcShort( 0xd0 ), /* Offset= 208 (284) */
/* 78 */ NdrFcLong( 0xd ), /* 13 */
/* 82 */ NdrFcShort( 0xe2 ), /* Offset= 226 (308) */
/* 84 */ NdrFcLong( 0x9 ), /* 9 */
/* 88 */ NdrFcShort( 0xee ), /* Offset= 238 (326) */
/* 90 */ NdrFcLong( 0x2000 ), /* 8192 */
/* 94 */ NdrFcShort( 0xfa ), /* Offset= 250 (344) */
/* 96 */ NdrFcLong( 0x24 ), /* 36 */
/* 100 */ NdrFcShort( 0x308 ), /* Offset= 776 (876) */
/* 102 */ NdrFcLong( 0x4024 ), /* 16420 */
/* 106 */ NdrFcShort( 0x302 ), /* Offset= 770 (876) */
/* 108 */ NdrFcLong( 0x4011 ), /* 16401 */
/* 112 */ NdrFcShort( 0x300 ), /* Offset= 768 (880) */
/* 114 */ NdrFcLong( 0x4002 ), /* 16386 */
/* 118 */ NdrFcShort( 0x2fe ), /* Offset= 766 (884) */
/* 120 */ NdrFcLong( 0x4003 ), /* 16387 */
/* 124 */ NdrFcShort( 0x2fc ), /* Offset= 764 (888) */
/* 126 */ NdrFcLong( 0x4004 ), /* 16388 */
/* 130 */ NdrFcShort( 0x2fa ), /* Offset= 762 (892) */
/* 132 */ NdrFcLong( 0x4005 ), /* 16389 */
/* 136 */ NdrFcShort( 0x2f8 ), /* Offset= 760 (896) */
/* 138 */ NdrFcLong( 0x400b ), /* 16395 */
/* 142 */ NdrFcShort( 0x2e6 ), /* Offset= 742 (884) */
/* 144 */ NdrFcLong( 0x400a ), /* 16394 */
/* 148 */ NdrFcShort( 0x2e4 ), /* Offset= 740 (888) */
/* 150 */ NdrFcLong( 0x4006 ), /* 16390 */
/* 154 */ NdrFcShort( 0x2ea ), /* Offset= 746 (900) */
/* 156 */ NdrFcLong( 0x4007 ), /* 16391 */
/* 160 */ NdrFcShort( 0x2e0 ), /* Offset= 736 (896) */
/* 162 */ NdrFcLong( 0x4008 ), /* 16392 */
/* 166 */ NdrFcShort( 0x2e2 ), /* Offset= 738 (904) */
/* 168 */ NdrFcLong( 0x400d ), /* 16397 */
/* 172 */ NdrFcShort( 0x2e0 ), /* Offset= 736 (908) */
/* 174 */ NdrFcLong( 0x4009 ), /* 16393 */
/* 178 */ NdrFcShort( 0x2de ), /* Offset= 734 (912) */
/* 180 */ NdrFcLong( 0x6000 ), /* 24576 */
/* 184 */ NdrFcShort( 0x2dc ), /* Offset= 732 (916) */
/* 186 */ NdrFcLong( 0x400c ), /* 16396 */
/* 190 */ NdrFcShort( 0x2da ), /* Offset= 730 (920) */
/* 192 */ NdrFcLong( 0x10 ), /* 16 */
/* 196 */ NdrFcShort( 0x8002 ), /* Simple arm type:
FC_CHAR */
/* 198 */ NdrFcLong( 0x12 ), /* 18 */
/* 202 */ NdrFcShort( 0x8006 ), /* Simple arm type:
FC_SHORT */
/* 204 */ NdrFcLong( 0x13 ), /* 19 */
/* 208 */ NdrFcShort( 0x8008 ), /* Simple arm type:
FC_LONG */
/* 210 */ NdrFcLong( 0x16 ), /* 22 */
/* 214 */ NdrFcShort( 0x8008 ), /* Simple arm type:
FC_LONG */
/* 216 */ NdrFcLong( 0x17 ), /* 23 */
/* 220 */ NdrFcShort( 0x8008 ), /* Simple arm type:
FC_LONG */
/* 222 */ NdrFcLong( 0xe ), /* 14 */
/* 226 */ NdrFcShort( 0x2be ), /* Offset= 702 (928) */
/* 228 */ NdrFcLong( 0x400e ), /* 16398 */
/* 232 */ NdrFcShort( 0x2c4 ), /* Offset= 708 (940) */
/* 234 */ NdrFcLong( 0x4010 ), /* 16400 */
/* 238 */ NdrFcShort( 0x2c2 ), /* Offset= 706 (944) */
/* 240 */ NdrFcLong( 0x4012 ), /* 16402 */
/* 244 */ NdrFcShort( 0x280 ), /* Offset= 640 (884) */
/* 246 */ NdrFcLong( 0x4013 ), /* 16403 */
/* 250 */ NdrFcShort( 0x27e ), /* Offset= 638 (888) */
/* 252 */ NdrFcLong( 0x4016 ), /* 16406 */
/* 256 */ NdrFcShort( 0x278 ), /* Offset= 632 (888) */
/* 258 */ NdrFcLong( 0x4017 ), /* 16407 */
/* 262 */ NdrFcShort( 0x272 ), /* Offset= 626 (888) */
/* 264 */ NdrFcLong( 0x0 ), /* 0 */
/* 268 */ NdrFcShort( 0x0 ), /* Offset= 0 (268) */
/* 270 */ NdrFcLong( 0x1 ), /* 1 */
/* 274 */ NdrFcShort( 0x0 ), /* Offset= 0 (274) */
/* 276 */ NdrFcShort( 0xffffffff ), /* Offset= -1 (275) */
/* 278 */
0x15, /*
FC_STRUCT */
0x7, /* 7 */
/* 280 */ NdrFcShort( 0x8 ), /* 8 */
/* 282 */ 0xb, /* FC_HYPER */
0x5b, /* FC_END */
/* 284 */
0x12, 0x0, /* FC_UP */
/* 286 */ NdrFcShort( 0xc ), /* Offset= 12 (298) */
/* 288 */
0x1b, /*
FC_CARRAY */
0x1, /* 1 */
/* 290 */ NdrFcShort( 0x2 ), /* 2 */
/* 292 */ 0x9, /* Corr desc: FC_ULONG */
0x0, /*
/* 294 */ NdrFcShort( 0xfffc ), /* -4 */
/* 296 */ 0x6, /* FC_SHORT */
0x5b, /* FC_END */
/* 298 */
0x17, /*
FC_CSTRUCT */
0x3, /* 3 */
/* 300 */ NdrFcShort( 0x8 ), /* 8 */
/* 302 */ NdrFcShort( 0xffffffff2 ), /* Offset= -14 (288) */
/* 304 */ 0x8, /* FC_LONG */
0x8, /* FC_LONG */
/* 306 */ 0x5c, /* FC_PAD */
0x5b, /* FC_END */
/* 308 */
0x2f, /* FC_IP */
0x5a, /*
/* 310 */ NdrFcLong( 0x0 ), /* 0 */
/* 314 */ NdrFcShort( 0x0 ), /* 0 */
/* 316 */ NdrFcShort( 0x0 ), /* 0 */
/* 318 */ 0xc0, /* 192 */
0x0, /* 0 */
/* 320 */ 0x0, /* 0 */
0x0, /* 0 */
/* 322 */ 0x0, /* 0 */
0x0, /* 0 */
/* 324 */ 0x0, /* 0 */
0x46, /* 70 */
/* 326 */
0x2f, /* FC_IP */
0x5a, /*
FC_CONSTANT_IID */
/* 328 */ NdrFcLong( 0x20400 ), /* 132096 */

```



```

/* 332 */ NdrFcShort( 0x0 ), /* 0 */
/* 334 */ NdrFcShort( 0x0 ), /* 0 */
/* 336 */ 0xc0, /* 192 */
/* 338 */ 0x0, /* 0 */
/* 340 */ 0x0, /* 0 */
/* 342 */ 0x0, /* 0 */
/* 344 */ /* 70 */
/* 346 */ NdrFcShort( 0x2 ), /* Offset= 2 (348) */
/* 348 */ /* FC_UP */
/* 350 */ NdrFcShort( 0x1fc ), /* Offset= 508 (858) */
/* 352 */ /* FC_UP */
FC_ENCAPSULATED_UNION /*
/* 354 */ NdrFcShort( 0x18 ), /* 24 */
/* 356 */ NdrFcShort( 0xa ), /* 10 */
/* 358 */ NdrFcLong( 0x8 ), /* 8 */
/* 362 */ NdrFcShort( 0x58 ), /* Offset= 88 (450) */
/* 364 */ NdrFcLong( 0xd ), /* 13 */
/* 368 */ NdrFcShort( 0x78 ), /* Offset= 120 (488) */
/* 370 */ NdrFcLong( 0x9 ), /* 9 */
/* 374 */ NdrFcShort( 0x94 ), /* Offset= 148 (522) */
/* 376 */ NdrFcLong( 0xc ), /* 12 */
/* 380 */ NdrFcShort( 0xbc ), /* Offset= 188 (568) */
/* 382 */ NdrFcLong( 0x24 ), /* 36 */
/* 386 */ NdrFcShort( 0x114 ), /* Offset= 276 (662) */
/* 388 */ NdrFcLong( 0x800d ), /* 32781 */
/* 392 */ NdrFcShort( 0x130 ), /* Offset= 304 (696) */
/* 394 */ NdrFcLong( 0x10 ), /* 16 */
/* 398 */ NdrFcShort( 0x148 ), /* Offset= 328 (726) */
/* 400 */ NdrFcLong( 0x2 ), /* 2 */
/* 404 */ NdrFcShort( 0x160 ), /* Offset= 352 (756) */
/* 406 */ NdrFcLong( 0x3 ), /* 3 */
/* 410 */ NdrFcShort( 0x178 ), /* Offset= 376 (786) */
/* 412 */ NdrFcLong( 0x14 ), /* 20 */
/* 416 */ NdrFcShort( 0x190 ), /* Offset= 400 (816) */
/* 418 */ NdrFcShort( 0xfffffff ), /* Offset= -1 (417) */
/* 420 */ /* FC_UP */
FC_CARRAY /*
/* 422 */ NdrFcShort( 0x4 ), /* 4 */
/* 424 */ 0x19, /* Corr desc: field pointer,
FC_ULONG */
/* 426 */ NdrFcShort( 0x0 ), /* 0 */
/* 428 */ /* FC_PP */
/* 430 */ /* FC_PAD */
/* 432 */ /* FC_PP */
FC_VARIABLE_REPEAT /*
/* 434 */ NdrFcShort( 0x4 ), /* 4 */
/* 436 */ NdrFcShort( 0x0 ), /* 0 */
/* 438 */ NdrFcShort( 0x1 ), /* 1 */
/* 440 */ NdrFcShort( 0x0 ), /* 0 */
/* 442 */ 0x12, 0x0, /* FC_UP */
/* 444 */ NdrFcShort( 0xfffff6e ), /* Offset= -146 (298) */
/* 446 */ /* FC_UP */
/* 448 */ 0x5c, /* FC_PAD */
/* 450 */ /* FC_END */
/* 452 */ /* FC_END */
/* 454 */ /* FC_END */
/* 456 */ /* FC_END */
FC_NO_REPEAT /*
/* 458 */ NdrFcShort( 0x4 ), /* 4 */
/* 460 */ NdrFcShort( 0x4 ), /* 4 */
/* 462 */ 0x11, 0x0, /* FC_RP */
/* 464 */ NdrFcShort( 0xfffffd4 ), /* Offset= -44 (420) */
/* 466 */ /* FC_END */
/* 468 */ 0x8, /* FC_LONG */
/* 470 */ /* FC_END */
FC_BOGUS_ARRAY /*
/* 472 */ NdrFcShort( 0x0 ), /* 0 */
/* 474 */ 0x19, /* Corr desc: field pointer,
FC_ULONG */
/* 476 */ NdrFcShort( 0x0 ), /* 0 */
/* 478 */ NdrFcLong( 0xffffffff ), /* -1 */
/* 482 */ 0x4c, /* FC_EMBEDDED_COMPLEX */
/* 484 */ NdrFcShort( 0xfffff50 ), /* Offset= -176 (308) */
/* 486 */ 0x5c, /* FC_PAD */
/* 488 */ /* FC_END */
/* 490 */ /* FC_END */
FC_BOGUS_STRUCT /*
/* 492 */ NdrFcShort( 0x8 ), /* 8 */
/* 494 */ NdrFcShort( 0x0 ), /* 0 */
/* 496 */ NdrFcShort( 0x6 ), /* Offset= 6 (500) */
/* 498 */ 0x8, /* FC_LONG */
/* 500 */ /* FC_END */
/* 502 */ NdrFcShort( 0xfffffe0 ), /* Offset= -32 (470) */
/* 504 */ /* FC_END */
/* 506 */ /* FC_END */
FC_BOGUS_ARRAY /*
/* 508 */ NdrFcShort( 0x0 ), /* 0 */
/* 510 */ 0x19, /* Corr desc: field pointer,
FC_ULONG */
/* 512 */ NdrFcShort( 0x0 ), /* 0 */

```

/* 512 */ NdrFcLong( 0xffffffff ), /* -1 */			0x2f,	/* FC_IP */
/* 516 */ 0x4c,	/* FC_EMBEDDED_COMPLEX */	0x0,	0x5a,	/*
/* 518 */ NdrFcShort( 0xfffff40 ), /* Offset= -192 (326) */		/* 0 */	FC_CONSTANT_IID */	
/* 520 */ 0x5c,	/* FC_PAD */	0x5b,	/* 586 */ NdrFcLong( 0x2f ), /* 47 */	/* 0 */
/* 522 */	/* FC_END */	0x1a,	/* 590 */ NdrFcShort( 0x0 ), /* 0 */	/* 0 */
FC_BOGUS_STRUCT */		/*	/* 592 */ NdrFcShort( 0x0 ), /* 0 */	/* 0 */
		0x3,	/* 594 */ 0xc0,	/* 192 */
/* 524 */ NdrFcShort( 0x8 ), /* 8 */	/* 3 */	/*	/* 596 */ 0x0,	/* 0 */
/* 526 */ NdrFcShort( 0x0 ), /* 0 */		/*	/* 598 */ 0x0,	/* 0 */
/* 528 */ NdrFcShort( 0x6 ), /* Offset= 6 (534) */		/*	/* 600 */ 0x0,	/* 0 */
/* 530 */ 0x8,	/* FC_LONG */	0x36,	/* 602 */	/* 70 */
FC_POINTER */		/*	0x1b,	/*
/* 532 */ 0x5c,	/* FC_PAD */	0x5b,	FC_CARRAY */	
/* 534 */	/* FC_END */	0x11, 0x0,	/* 604 */ NdrFcShort( 0x1 ), /* 1 */	/* 0 */
/* 536 */ NdrFcShort( 0xfffffe0 ), /* Offset= -32 (504) */		/* FC_RP */	/* 606 */ 0x19,	/* Corr desc: field pointer,
/* 538 */		/*	FC_ULONG */	
FC_CARRAY */		0x1b,	/* 608 */ NdrFcShort( 0x4 ), /* 4 */	/* *
/* 540 */ NdrFcShort( 0x4 ), /* 4 */	/* 3 */	/*	/* 610 */ 0x1,	/* FC_BYTE */
/* 542 */ 0x19,	/* Corr desc: field pointer,	0x3,	/* 612 */	0x5b,
FC_ULONG */		/* *	FC_BOGUS_STRUCT */	/* FC_END */
/* 544 */ NdrFcShort( 0x0 ), /* 0 */	/* *	0x0,	0x1a,	/*
/* 546 */		/*	0x3,	/* 3 */
/* 548 */		0x4b,	/* 614 */ NdrFcShort( 0x10 ), /* 16 */	/* *
FC_VARIABLE_REPEAT */		0x5c,	/* 616 */ NdrFcShort( 0x0 ), /* 0 */	/* *
/* 550 */ NdrFcShort( 0x4 ), /* 4 */	/* FC_PP */	0x48,	/* 618 */ NdrFcShort( 0xa ), /* Offset= 10 (628) */	/* *
/* 552 */ NdrFcShort( 0x0 ), /* 0 */	/* FC_PAD */	/*	/* 620 */ 0x8,	/* FC_LONG */
/* 554 */ NdrFcShort( 0x1 ), /* 1 */	/* *	0x49,	/*	/* FC_LONG
/* 556 */ NdrFcShort( 0x0 ), /* 0 */		/*	/* 622 */ 0x4c,	/* FC_EMBEDDED_COMPLEX */
/* 558 */ NdrFcShort( 0x0 ), /* 0 */		/*	/* 624 */ NdrFcShort( 0xfffffd8 ), /* Offset= -40 (584) */	/* *
/* 560 */ 0x12, 0x0, /* FC_UP */		/*	/* 626 */ 0x36,	/* FC_POINTER */
/* 562 */ NdrFcShort( 0xffffdce ), /* Offset= -562 (0) */		/*	/* 628 */	0x5b,
/* 564 */	/* FC_END */	0x5b,	/* 630 */ NdrFcShort( 0xfffffe4 ), /* Offset= -28 (602) */	/* FC_END */
/*		0x8,	/* 632 */	0x12, 0x0, /* FC_UP */
/* 566 */ 0x5c,	/* FC_PAD */	/*	FC_CARRAY */	/*
/* 568 */	/* FC_END */	0x3,	/* 634 */ NdrFcShort( 0x4 ), /* 4 */	/* *
FC_BOGUS_STRUCT */		/*	/* 636 */ 0x19,	/* Corr desc: field pointer,
/* 570 */ NdrFcShort( 0x8 ), /* 8 */	/* 3 */	/*	FC_ULONG */	
/* 572 */ NdrFcShort( 0x0 ), /* 0 */		/*	/* 638 */ NdrFcShort( 0x0 ), /* 0 */	/* *
/* 574 */ NdrFcShort( 0x6 ), /* Offset= 6 (580) */		/*	/* 640 */	0x4b,
/* 576 */ 0x8,	/* FC_LONG */	0x1a,	/* 642 */	0x5c,
FC_POINTER */		/*	FC_VARIABLE_REPEAT */	/*
/* 578 */ 0x5c,	/* FC_PAD */	0x3,	FC_FIXED_OFFSET */	
/* 580 */	/* FC_END */	/*	/* 644 */ NdrFcShort( 0x4 ), /* 4 */	/* *
/* 582 */ NdrFcShort( 0xfffffd4 ), /* Offset= -44 (538) */		/*	/* 646 */ NdrFcShort( 0x0 ), /* 0 */	/* *
/* 584 */		0x11, 0x0,	/* 648 */ NdrFcShort( 0x1 ), /* 1 */	/* *
	/* FC_RP */	/*	/* 650 */ NdrFcShort( 0x0 ), /* 0 */	/* *
	/* Offset= -44 (612) */	0x5b,	/* 652 */ NdrFcShort( 0x0 ), /* 0 */	/* *
		/*	/* 654 */ 0x12, 0x0, /* FC_UP */	/* *
		0x11, 0x0,	/* 656 */ NdrFcShort( 0xfffffd4 ), /* Offset= -44 (612) */	/* *
		/*	/* 658 */	/* *

	0x5b,	/* FC_END */		0x0,	/* */
	0x8,	/* FC_LONG	/* 722 */ NdrFcShort( 0x0 ),	/* 0 */	
*/			/* 724 */ 0x1,	/* FC_BYTE */	
/* 660 */ 0x5c,	/* FC_PAD */		/* 726 */	0x5b,	/* FC_END */
/* 662 */	0x5b,	/* FC_END */		0x16,	/*
FC_BOGUS_STRUCT */	0x1a,	/*	FC_PSTRUCT */	0x3,	/* 3 */
	0x3,	/* 3 */	/* 728 */ NdrFcShort( 0x8 ),	/* 8 */	
/* 664 */ NdrFcShort( 0x8 ),	/* 8 */		/* 730 */	0x4b,	/* FC_PP */
/* 666 */ NdrFcShort( 0x0 ),	/* 0 */			0x5c,	/* FC_PAD */
/* 668 */ NdrFcShort( 0x6 ),	/* Offset= 6 (674) */		/* 732 */		
/* 670 */ 0x8,	/* FC_LONG */			0x46,	/*
FC_POINTER */	0x36,	/*	FC_NO_REPEAT */	0x5c,	/* FC_PAD */
/* 672 */ 0x5c,	/* FC_PAD */		/* 734 */ NdrFcShort( 0x4 ),	/* 4 */	
/* 674 */	0x5b,	/* FC_END */	/* 736 */ NdrFcShort( 0x4 ),	/* 4 */	
	0x11, 0x0, /* FC_RP */		/* 738 */ 0x12, 0x0, /* FC_UP */		
/* 676 */ NdrFcShort( 0xfffffd4 ),	/* Offset= -44 (632) */		/* 740 */ NdrFcShort( 0xfffffe8 ),	/* Offset= -24 (716) */	
/* 678 */			/* 742 */	0x5b,	/* FC_END */
FC_SMFARRAY */	0x1d,	/*		0x8,	/* FC_LONG
	0x0,	/* 0 */	*/		
/* 680 */ NdrFcShort( 0x8 ),	/* 8 */		/* 744 */ 0x8,	/* FC_LONG */	
/* 682 */ 0x2,	/* FC_CHAR */		/* 746 */	0x5b,	/* FC_END */
/* 684 */	0x5b,	/* FC_END */		0x1b,	/*
FC_STRUCT */	0x15,	/*	FC_CARRAY */	0x1,	/* 1 */
	0x3,	/* 3 */	/* 748 */ NdrFcShort( 0x2 ),	/* 2 */	
/* 686 */ NdrFcShort( 0x10 ),	/* 16 */		/* 750 */ 0x19,	/* Corr desc: field pointer,	
/* 688 */ 0x8,	/* FC_LONG */		FC_ULONG */		
*/	0x6,	/* FC_SHORT	/* 752 */ NdrFcShort( 0x0 ),	/* 0 */	
/* 690 */ 0x6,	/* FC_SHORT */		/* 754 */ 0x6,	/* FC_SHORT */	
FC_EMBEDDED_COMPLEX */	0x4c,	/*	/* 756 */	0x5b,	/* FC_END */
/* 692 */ 0x0,	/* 0 */			0x16,	/*
Offset= -15 (678) */	NdrFcShort( 0xfffff1 ),	/*	FC_PSTRUCT */	0x3,	/* 3 */
/* 696 */	0x5b,	/* FC_END */	/* 758 */ NdrFcShort( 0x8 ),	/* 8 */	
FC_BOGUS_STRUCT */	0x1a,	/*	/* 760 */	0x4b,	/* FC_PP */
	0x3,	/* 3 */	/* 762 */	0x5c,	/* FC_PAD */
/* 698 */ NdrFcShort( 0x18 ),	/* 24 */			0x46,	/*
/* 700 */ NdrFcShort( 0x0 ),	/* 0 */		FC_NO_REPEAT */	0x5c,	/* FC_PAD */
/* 702 */ NdrFcShort( 0xa ),	/* Offset= 10 (712) */		/* 764 */ NdrFcShort( 0x4 ),	/* 4 */	
/* 704 */ 0x8,	/* FC_LONG */		/* 766 */ NdrFcShort( 0x4 ),	/* 4 */	
FC_POINTER */	0x36,	/*	/* 768 */ 0x12, 0x0, /* FC_UP */		
/* 706 */ 0x4c,	/* FC_EMBEDDED_COMPLEX */		/* 770 */ NdrFcShort( 0xfffffe8 ),	/* Offset= -24 (746) */	
	0x0,	/* 0 */	/* 772 */	0x5b,	/* FC_END */
/* 708 */ NdrFcShort( 0xfffffe8 ),	/* Offset= -24 (684) */			0x8,	/* FC_LONG
/* 710 */ 0x5c,	/* FC_PAD */		*/		
/* 712 */	0x5b,	/* FC_END */	/* 774 */ 0x8,	/* FC_LONG */	
/* 714 */ NdrFcShort( 0xfffff0c ),	/* Offset= -244 (470) */		/* 776 */	0x5b,	/* FC_END */
/* 716 */	0x1b,	/*		0x1b,	/*
FC_CARRAY */	0x0,	/* 0 */	FC_CARRAY */	0x3,	/* 3 */
/* 718 */ NdrFcShort( 0x1 ),	/* 1 */		/* 778 */ NdrFcShort( 0x4 ),	/* 4 */	
/* 720 */ 0x19,	/* Corr desc: field pointer,		/* 780 */ 0x19,	/* Corr desc: field pointer,	
FC_ULONG */			FC_ULONG */		

/* 782 */ NdrFcShort( 0x0 ),	0x0,	/* */	/*	0x8,	/* FC_LONG
/* 784 */ 0x8,	/* FC_LONG */		/* 842 */ 0x5c,	/* FC_PAD */	
/* 786 */	0x5b,	/* FC_END */	/* 844 */	0x5b,	/* FC_END */
FC_PSTRUCT */	0x16,	/*	FC_CARRAY */	0x1b,	/*
/* 788 */ NdrFcShort( 0x8 ),	0x3,	/* 3 */	/* 846 */ NdrFcShort( 0x8 ),	0x3,	/* 3 */
/* 790 */	/* 8 */		/* 848 */ 0x7,	/* 8 */	/* Corr desc: FC_USHORT */
/* 792 */	0x4b,	/* FC_PP */	/* 850 */ NdrFcShort( 0xffd8 ),	0x0,	/* */
	0x5c,	/* FC_PAD */	/* 852 */ 0x4c,	/* FC_EMBEDDED_COMPLEX */	/* -40 */
FC_NO_REPEAT */	0x46,	/*	/* 854 */ NdrFcShort( 0xfffffee ),	0x0,	/* 0 */
/* 794 */ NdrFcShort( 0x4 ),	0x5c,	/* FC_PAD */	/* 856 */ 0x5c,	/* Offset= -18 (836) */	/* FC_PAD */
/* 796 */ NdrFcShort( 0x4 ),	/* 4 */		/* 858 */	0x5b,	/* FC_END */
/* 798 */ 0x12, 0x0, /* FC_UP */	/* 4 */		FC_BOGUS_STRUCT */	0x1a,	/*
/* 800 */ NdrFcShort( 0xfffffe8 ),	/* Offset= -24 (776) */		/* 860 */ NdrFcShort( 0x28 ),	0x3,	/* 3 */
/* 802 */	0x5b,	/* FC_END */	/* 862 */ NdrFcShort( 0xfffffee ),	/* 40 */	/* Offset= -18 (844) */
	0x8,	/* FC_LONG	/* 864 */ NdrFcShort( 0x0 ),	/* Offset= 0 (864) */	
/*	/* FC_LONG */		/* 866 */ 0x6,	/* FC_SHORT */	
/* 804 */ 0x8,	0x5b,	/* FC_END */	/*	0x6,	/* FC_SHORT
/* 806 */	0x1b,	/*	/* 868 */ 0x38,	/* FC_ALIGNM4 */	
FC_CARRAY */	0x7,	/* 7 */	/* 870 */ 0x8,	0x8,	/* FC_LONG
/* 808 */ NdrFcShort( 0x8 ),	/* 8 */		/* FC_LONG */	0x4c,	/*
/* 810 */ 0x19,	/* Corr desc: field pointer,		FC_EMBEDDED_COMPLEX */	/* 0 */	
FC_ULONG */	0x0,	/* */	/* 872 */ 0x0,	NdrFcShort( 0xfffffd7 ),	/*
/* 812 */ NdrFcShort( 0x0 ),	/* 0 */		Offset= -521 (352) */	0x5b,	/* FC_END */
/* 814 */ 0xb,	/* FC_HYPER */		/* 876 */	0x12, 0x0, /* FC_UP */	
/* 816 */	0x5b,	/* FC_END */	/* 878 */ NdrFcShort( 0xfffffef6 ),	/* Offset= -266 (612) */	
FC_PSTRUCT */	0x16,	/*	/* 880 */	0x12, 0x8, /* FC_UP	
/* 818 */ NdrFcShort( 0x8 ),	0x3,	/* 3 */	[simple_pointer] */	/* FC_BYTE */	
/* 820 */	/* 8 */		/* 882 */ 0x1,	0x5c,	/* FC_PAD */
/* 822 */	0x4b,	/* FC_PP */	/* 884 */	0x12, 0x8, /* FC_UP	
	0x5c,	/* FC_PAD */	[simple_pointer] */	/* FC_SHORT */	
FC_NO_REPEAT */	0x46,	/*	/* 886 */ 0x6,	0x5c,	/* FC_PAD */
/* 824 */ NdrFcShort( 0x4 ),	0x5c,	/* FC_PAD */	/* 888 */	0x12, 0x8, /* FC_UP	
/* 826 */ NdrFcShort( 0x4 ),	/* 4 */		[simple_pointer] */	/* FC_LONG */	
/* 828 */ 0x12, 0x0, /* FC_UP */	/* 4 */		/* 890 */ 0x8,	0x5c,	/* FC_PAD */
/* 830 */ NdrFcShort( 0xfffffe8 ),	/* Offset= -24 (806) */		/* 892 */	0x12, 0x8, /* FC_UP	
/* 832 */	0x5b,	/* FC_END */	[simple_pointer] */	/* FC_FLOAT */	
	0x8,	/* FC_LONG	/* 894 */ 0xa,	0x5c,	/* FC_PAD */
/*	/* FC_LONG */		/* 896 */	0x12, 0x8, /* FC_UP	
/* 834 */ 0x8,	0x5b,	/* FC_END */	[simple_pointer] */	/* FC_DOUBLE */	
/* 836 */	0x15,	/*	/* 898 */ 0xc,	0x5c,	/* FC_PAD */
FC_STRUCT */	0x3,	/* 3 */	/* 900 */	0x12, 0x0, /* FC_UP */	
/* 838 */ NdrFcShort( 0x8 ),	/* 8 */		/* 902 */ NdrFcShort( 0xfffffd90 ),	/* Offset= -624 (278) */	
/* 840 */ 0x8,	/* FC_LONG */				

```

/* 904 */
                                0x12, 0x10, /* FC_UP */
/* 906 */ NdrFcShort( 0xffffd92 ), /* Offset= -622 (284) */
/* 908 */
                                0x12, 0x10, /* FC_UP */
/* 910 */ NdrFcShort( 0xffffda6 ), /* Offset= -602 (308) */
/* 912 */
                                0x12, 0x10, /* FC_UP */
/* 914 */ NdrFcShort( 0xffffdb4 ), /* Offset= -588 (326) */
/* 916 */
                                0x12, 0x10, /* FC_UP */
/* 918 */ NdrFcShort( 0xffffdc2 ), /* Offset= -574 (344) */
/* 920 */
                                0x12, 0x10, /* FC_UP */
/* 922 */ NdrFcShort( 0x2 ), /* Offset= 2 (924) */
/* 924 */
                                0x12, 0x0, /* FC_UP */
/* 926 */ NdrFcShort( 0xffffc62 ), /* Offset= -926 (0) */
/* 928 */
                                0x15, /*
FC_STRUCT */
                                0x7, /* 7 */
/* 930 */ NdrFcShort( 0x10 ), /* 16 */
/* 932 */ 0x6, /* FC_SHORT */
                                0x1, /* FC_BYTE */
/*
/* 934 */ 0x1, /* FC_BYTE */
                                0x38, /*
FC_ALIGNM4 */
/* 936 */ 0x8, /* FC_LONG */
                                0x39, /*
FC_ALIGNM8 */
/* 938 */ 0xb, /* FC_HYPER */
                                0x5b, /* FC_END */
/* 940 */
                                0x12, 0x0, /* FC_UP */
/* 942 */ NdrFcShort( 0xfffff2 ), /* Offset= -14 (928) */
/* 944 */
                                0x12, 0x8, /* FC_UP
[simple_pointer] */
/* 946 */ 0x2, /* FC_CHAR */
                                0x5c, /* FC_PAD */
/* 948 */
                                0x1a, /*
FC_BOGUS_STRUCT */
                                0x7, /* 7 */
/* 950 */ NdrFcShort( 0x20 ), /* 32 */
/* 952 */ NdrFcShort( 0x0 ), /* 0 */
/* 954 */ NdrFcShort( 0x0 ), /* Offset= 0 (954) */
/* 956 */ 0x8, /* FC_LONG */
                                0x8, /* FC_LONG */
/*
/* 958 */ 0x6, /* FC_SHORT */
                                0x6, /* FC_SHORT */
/*
/* 960 */ 0x6, /* FC_SHORT */
                                0x6, /* FC_SHORT */
/*
/* 962 */ 0x4c, /* FC_EMBEDDED_COMPLEX */
                                0x0, /* 0 */
/* 964 */ NdrFcShort( 0xffffc42 ), /* Offset= -958 (6) */
/* 966 */ 0x5c, /* FC_PAD */
                                0x5b, /* FC_END */
/* 968 */ 0xb4, /* FC_USER_MARSHAL */
                                0x83, /* 131 */
/* 970 */ NdrFcShort( 0x0 ), /* 0 */
/* 972 */ NdrFcShort( 0x10 ), /* 16 */
/* 974 */ NdrFcShort( 0x0 ), /* 0 */
/* 976 */ NdrFcShort( 0xffffc32 ), /* Offset= -974 (2) */
/* 978 */
                                0x11, 0x4, /* FC_RP
[allocated_on_stack] */
/* 980 */ NdrFcShort( 0x6 ), /* Offset= 6 (986) */
/* 982 */
                                0x13, 0x0, /* FC_OP */
/* 984 */ NdrFcShort( 0xfffff4c ), /* Offset= -36 (948) */
/* 986 */ 0xb4, /* FC_USER_MARSHAL */
                                0x83, /* 131 */
/* 988 */ NdrFcShort( 0x0 ), /* 0 */
/* 990 */ NdrFcShort( 0x10 ), /* 16 */
/* 992 */ NdrFcShort( 0x0 ), /* 0 */
/* 994 */ NdrFcShort( 0xfffff4 ), /* Offset= -12 (982) */
                                0x0
}
};
const CInterfaceProxyVtbl* _tpcc_com_remote_ps_ProxyVtblList[] =
{
( CInterfaceProxyVtbl* ) &_ITPCCremProxyVtbl,
0
};
const CInterfaceStubVtbl* _tpcc_com_remote_ps_StubVtblList[] =
{
( CInterfaceStubVtbl* ) &_ITPCCremStubVtbl,
0
};
PCInterfaceName const _tpcc_com_remote_ps_InterfaceNamesList[] =
{
"ITPCCrem",
0
};
#define _tpcc_com_remote_ps_CHECK_IID(n)
IID_GENERIC_CHECK_IID(_tpcc_com_remote_ps, pIID, n)
int __stdcall _tpcc_com_remote_ps_IID_Lookup( const IID* pIID, int
*pIndex )
{
if(!_tpcc_com_remote_ps_CHECK_IID(0))
{
*pIndex = 0;
return 1;
}
return 0;
}
const ExtendedProxyFileInfo tpcc_com_remote_ps_ProxyFileInfo =
{
(PCInterfaceProxyVtblList*) &
_tpcc_com_remote_ps_ProxyVtblList,
(PCInterfaceStubVtblList*) & _tpcc_com_remote_ps_StubVtblList,
(const PCInterfaceName*) &
_tpcc_com_remote_ps_InterfaceNamesList,
0, // no delegation
& _tpcc_com_remote_ps_IID_Lookup,
1,
2,
0, /* table of [async_uuid] interfaces */
}

```

```

0, /* Filler1 */
0, /* Filler2 */
0 /* Filler3 */
};

```

## ***tpcc\_com\_remote\_resource.h***

```

///<{NO_DEPENDENCIES}
// Microsoft Developer Studio generated include file.
// Used by tpcc_com_remote.rc
//
#define IDS_PROJNAME            100
#define IDR_TPCCrem             101

// Next default values for new objects
//
#ifdef APSTUDIO_INVOKED
#ifndef APSTUDIO_READONLY_SYMBOLS
#define _APS_NEXT_RESOURCE_VALUE    202
#define _APS_NEXT_COMMAND_VALUE    32768
#define _APS_NEXT_CONTROL_VALUE    201
#define _APS_NEXT_SYMED_VALUE      102
#endif
#endif

```

## ***tpcc\_com\_sl.rgs***

```

HKCR
{
    TPCC.StockLevel.1 = s 'StockLevel Class'
    {
        CLSID = s
        '{2668369E-A50D-11D2-BA4E-00C04FBFE08B}'
    }
    TPCC.StockLevel = s 'StockLevel Class'
    {
        CurVer = s 'TPCC.StockLevel.1'
    }
    NoRemove CLSID
    {
        ForceRemove
        {2668369E-A50D-11D2-BA4E-00C04FBFE08B} = s 'StockLevel
        Class'
    }
    ProgID = s 'TPCC.StockLevel.1'
    VersionIndependentProgID = s
    'TPCC.StockLevel'
    InprocServer32 = s '%MODULE%'
    {
        val ThreadingModel = s
    }
}
}

```

## ***tpcc\_odbc.cpp***

```

/* FILE: TPCC_ODBC.CPP
 * 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: Implements ODBC calls for TPC-C
 * txns.
 * Contact: Charles Levine (clevine@microsoft.com)
 *
 * Change history:
 * 4.30.000 - changes for w_id > 32767 and
 * multinode options.
 * 4.20.000 - updated rev number to match kit
 * 4.10.001 - not deleting error class in catch
 * handler on deadlock retry;
 * not a functional bug, but
 * a memory leak
 */
#include <windows.h>
#include <stdio.h>
#include <assert.h>

#define DBNTWIN32
#include <sqltypes.h>
#include <sql.h>
#include <sqllex.h>
#include <odbcss.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_odbc.h"

// 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

const int iErrOleDbProvider = 7312;
const char sErrTimeoutExpired[] = "Timeout expired";

#define ErrTrace 1

#ifdef ErrTrace
FILE *errFile;
#endif

static SQLHENV henv = SQL_NULL_HENV;
// ODBC environment handle

BOOL APIENTRY DllMain(HMODULE hModule, DWORD
ul_reason_for_call, LPVOID lpReserved)
{
    switch( ul_reason_for_call )
    {
        case DLL_PROCESS_ATTACH:
            DisableThreadLibraryCalls(hModule);
    }
}

```

```

        if (
SQLAllocHandleStd(SQL_HANDLE_ENV,SQL_NULL_HANDLE,
&henv) != SQL_SUCCESS )
        return FALSE;
#ifdef ErrTrace
        errFile =
fopen("c:\\odbcerrfile.txt","a");
#endif
        break;
        case DLL_PROCESS_DETACH:
        if (henv != NULL)
            SQLFreeEnv(henv);
        break;
        default:
            /* nothing */;
    }
    return TRUE;
}

/* FUNCTION: CTPCC_ODBC_ERR::ErrorText
 *
 */

char* CTPCC_ODBC_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,
"orders found for customer." },
        { ERR_RETRIED_TRANS,
"Retries before transaction succeeded." },
        { 0,
"" }
    };

};

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_ODBC* CTPCC_ODBC_new(
    LPCSTR szServer, // name of SQL server
    LPCSTR szUser, // user name
    LPCSTR szPassword, // password for login
    LPCSTR szHost, // not used
    LPCSTR szDatabase ) // name of database to use
{
    return new CTPCC_ODBC( szServer, szUser, szPassword,
szHost, szDatabase );
}

CTPCC_ODBC::CTPCC_ODBC (
    LPCSTR szServer, //
name of SQL server
    LPCSTR szUser, //
user name for login
    LPCSTR szPassword, //
password for login
    LPCSTR szHost, //
not used
    LPCSTR szDatabase //
name of database to use
)
{
    RETCODE rc;

    // initialization
    m_hdbc = SQL_NULL_HDBC;
    m_hstmt = SQL_NULL_HSTMT;

    m_hstmtNewOrder = SQL_NULL_HSTMT;
    m_hstmtPayment = SQL_NULL_HSTMT;
    m_hstmtDelivery = SQL_NULL_HSTMT;
    m_hstmtOrderStatus = SQL_NULL_HSTMT;
    m_hstmtStockLevel = SQL_NULL_HSTMT;

    m_descNewOrderCols1 = SQL_NULL_HDESC;
    m_descNewOrderCols2 = SQL_NULL_HDESC;
    m_descOrderStatusCols1 = SQL_NULL_HDESC;
    m_descOrderStatusCols2 = SQL_NULL_HDESC;

    if ( SQLAllocHandle(SQL_HANDLE_DBC,henv,
&m_hdbc) != SQL_SUCCESS )
        ThrowError(CODBCERR::eAllocHandle);

    if ( SQLSetConnectOption(m_hdbc, SQL_PACKET_SIZE,
4096) != SQL_SUCCESS )
        ThrowError(CODBCERR::eConnOption);

    {
        char
szConnectStr[256];
        char
szOutStr[1024];
        SQLSMALLINT iOutStrLen;

        sprintf( szConnectStr, "DRIVER=SQL
Server;SERVER=%s;UID=%s;PWD=%s;DATABASE=%s",
szServer, szUser, szPassword,
szDatabase );

        rc = SQLDriverConnect(m_hdbc, NULL,
(SQLCHAR*)szConnectStr, sizeof(szConnectStr),
(SQLCHAR*)szOutStr,
sizeof(szOutStr), &iOutStrLen, SQL_DRIVER_NOPROMPT);

        if (rc != SQL_SUCCESS && rc !=
SQL_SUCCESS_WITH_INFO)
            ThrowError(CODBCERR::eConnect);
    }
}

```

```

        if (SQLAllocHandle(SQL_HANDLE_STMT,m_hdbc,
&m_hstmt) != SQL_SUCCESS)
            ThrowError(CODBCERR::eAllocHandle);

        {

            // set some options affecting connection behavior
            char            buffer[128];

            // set some options affecting connection behavior
            strcpy(buffer, "set nocount on ");
            strcat(buffer, "set XACT_ABORT ON ");

            // for coyote
            strcat(buffer, "set ansi_warnings on ");
            strcat(buffer, "set ansi_nulls on ");
            rc = SQLExecDirect(m_hstmt, (unsigned char
*)buffer, SQL_NTS);
            if (rc != SQL_SUCCESS && rc !=
SQL_SUCCESS_WITH_INFO)
                ThrowError(CODBCERR::eExecDirect);

            // verify that version of stored procs on server is
correct
            char db_sp_version[10];
            strcpy(buffer, "{call tpcc_version}");
            rc = SQLExecDirect(m_hstmt, (unsigned char
*)buffer, SQL_NTS);
            if (rc != SQL_SUCCESS && rc !=
SQL_SUCCESS_WITH_INFO)
                ThrowError(CODBCERR::eExecDirect);
            if ( SQLBindCol(m_hstmt, 1, SQL_C_CHAR,
&db_sp_version, sizeof(db_sp_version), NULL) != SQL_SUCCESS )
                ThrowError(CODBCERR::eBindCol);
            if ( SQLFetch(m_hstmt) == SQL_ERROR )
                ThrowError(CODBCERR::eFetch);
            if (strcmp(db_sp_version,sVersion)
                throw new CTPCC_ODBC_ERR(
CTPCC_ODBC_ERR::ERR_WRONG_SP_VERSION);

            SQLFreeHandle(SQL_HANDLE_STMT,
m_hstmt);
        }

        // Bind parameters for each of the transactions
        InitNewOrderParams();
        InitPaymentParams();
        InitOrderStatusParams();
        InitDeliveryParams();
        InitStockLevelParams();
    }

CTPCC_ODBC::~CTPCC_ODBC( void )
{
    // note: descriptors are automatically released when the
connection is dropped
    SQLFreeHandle(SQL_HANDLE_STMT,
m_hstmtNewOrder);
    SQLFreeHandle(SQL_HANDLE_STMT,
m_hstmtPayment);
    SQLFreeHandle(SQL_HANDLE_STMT,
m_hstmtDelivery);
    SQLFreeHandle(SQL_HANDLE_STMT,
m_hstmtOrderStatus);
}

```

```

        SQLFreeHandle(SQL_HANDLE_STMT,
m_hstmtStockLevel);

        SQLDisconnect(m_hdbc);
        SQLFreeHandle(SQL_HANDLE_DBC,m_hdbc);
    }

void CTPCC_ODBC::ThrowError( CODBCERR::ACTION eAction )
{
    RETCODE            rc;
    SDWORD            INativeError;
    char            szState[6];
    char
    szMsg[SQL_MAX_MESSAGE_LENGTH];
    char
    szTmp[6*SQL_MAX_MESSAGE_LENGTH];
    CODBCERR            *pODBCErr;
    //
    not allocated until needed (maybe never)

    pODBCErr = new CODBCERR();

    pODBCErr->m_NativeError = 0;
    pODBCErr->m_eAction = eAction;
    pODBCErr->m_bDeadLock = FALSE;

    szTmp[0] = 0;
    while (TRUE)
    {
        rc = SQLError(henv, m_hdbc, m_hstmt, (BYTE
*)&szState, &INativeError,
                                (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 &&
            pODBCErr->m_NativeError =

        // quit if there isn't enough room to concatenate
error text
        if ( ( strlen(szMsg) + 2) > (sizeof(szTmp) -
strlen(szTmp)) )
        {
#ifdef ErrTrace
            fprintf(errFile, "%s\n",szMsg);
            fflush(errFile);

            break;
        }

        // include line break after first error msg
        if (szTmp[0] != 0)
            strcat( szTmp, "\n");
        strcat( 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 );
#ifdef ErrTrace
            fprintf(errFile,"%s\n",szTmp);
            fflush(errFile);
#endif
        }

        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;

    int i = 0;
    if ( SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT,SQL_C_SLONG,SQL_INTEGER,0,0,
&m_txn.StockLevel.w_id,0, NULL) != SQL_SUCCESS
        || SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT,SQL_C_UTINYINT,SQL_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          rc;
    int              iTryCount = 0;

    m_hstmt = m_hstmtStockLevel;

    while (TRUE)
    {
        try
        {
            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 period
                delete e;
                Sleep(10 * iTryCount);
            }
        }
    }

    // if (iTryCount)
    // throw new

    CTPCC_ODBC_ERR(CTPCC_ODBC_ERR::ERR_RETRIED_TRAN
S, 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_SLONG,SQL_INTEGER,0,0,
&m_txn.NewOrder.w_id,0, NULL) != SQL_SUCCESS
        || SQLBindParameter(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_SLONG, SQL_INTEGER, 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);
        }
#ifdef new_order_strstr
        // 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);
        i = 0;
        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);
#else
        // prototype to eliminate patindex in server; shift work to
client
        i = 0;
        if ( SQLBindCol(m_hstmt, ++i, SQL_C_CHAR,
&m_ol_i_name, sizeof(m_ol_i_name), NULL) != SQL_SUCCESS
            || SQLBindCol(m_hstmt, ++i,
SQL_C_SSHORT, &m_ol_stock, 0, NULL) != SQL_SUCCESS
            || SQLBindCol(m_hstmt, ++i, SQL_C_CHAR,
&m_i_data, sizeof(m_i_data), NULL) != SQL_SUCCESS
            || SQLBindCol(m_hstmt, ++i, SQL_C_CHAR,
&m_s_data, sizeof(m_s_data), NULL) != SQL_SUCCESS
            || SQLBindCol(m_hstmt, ++i,
SQL_C_DOUBLE, &m_ol_i_price, 0, NULL) != SQL_SUCCESS
            || SQLBindCol(m_hstmt, ++i,
SQL_C_DOUBLE, &m_ol_amount, 0, NULL) != SQL_SUCCESS
        )
            ThrowError(CODBCERR::eBindCol);
#endif

        // 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);

```

```

        i = 0;
        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(BOOL fLocalFlag)
{
    int i;
    RETCODE rc;
    int iTryCount = 0;

    // 0 1 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;
    wcsncpy( &szSqlTemplate[i], L" )" );

    // check whether any order lines are for a remote warehouse
    m_txn.NewOrder.o_all_local = 1;
    for ( i = 0; i < 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
    }
    }
    while (TRUE)
    {
        try
        {
            m_BindOffset = 0;
            rc = SQLExecDirectW(m_hstmt,
(SQLWCHAR*)szSqlTemplate,SQL_NTS);
            if (rc != SQL_SUCCESS && rc !=
SQL_SUCCESS_WITH_INFO)
            {
#ifdef ErrTrace
                fprintf(errFile,"NewOrder
Error\nw_id: %d\nd_id: %d\nc_id: %d\nError:\n\n",
m_txn.NewOrder.w_id,
m_txn.NewOrder.d_id,
m_txn.NewOrder.c_id);
                fflush(errFile);
#endif

                ThrowError(CODBCERR::eExecDirect);
            }
            // Get order line results
            m_txn.NewOrder.total_amount = 0;
            for (i = 0;
i < m_txn.NewOrder.o_ol_cnt; i++)
            {
#ifdef new_order_strstr
                // set the bind offset
                value...
                m_BindOffset = i *
sizeof(m_txn.NewOrder.OL[0]);
            }
            if ( SQLFetch(m_hstmt)
== SQL_ERROR)
                ThrowError(CODBCERR::eFetch);
            #else
                if ( SQLFetch(m_hstmt)
== SQL_ERROR)
                    ThrowError(CODBCERR::eFetch);
            }
            strcpy(
m_txn.NewOrder.OL[i].ol_i_name, m_ol_i_name );
            if ( strstr(m_i_data,
"ORIGINAL") != NULL && strstr(m_s_data, "ORIGINAL") != NULL
)
                m_txn.NewOrder.OL[i].ol_brand_generic[0] = 'B';
            else
                m_txn.NewOrder.OL[i].ol_brand_generic[0] = 'G';
            m_txn.NewOrder.OL[i].ol_brand_generic[1] = 0;
        }
    }
}

m_txn.NewOrder.OL[i].ol_stock = m_ol_stock;
m_txn.NewOrder.OL[i].ol_i_price =
m_ol_i_price;
m_txn.NewOrder.OL[i].ol_amount =
m_ol_amount;
#endif
// move to the next
resultset
if (
SQLMoreResults(m_hstmt) == SQL_ERROR )
    ThrowError(CODBCERR::eMoreResults);
m_txn.NewOrder.total_amount += m_txn.NewOrder.OL[i].ol_amount;
}
// 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 ( SQLFetch(m_hstmt) ==
SQL_ERROR)
    ThrowError(CODBCERR::eFetch);
SQLFreeStmt(m_hstmt,
SQL_CLOSE);
if (m_no_commit_flag == 1)
{
    m_txn.NewOrder.total_amount *= ((1 + m_txn.NewOrder.w_tax +
m_txn.NewOrder.d_tax) * (1 - m_txn.NewOrder.c_discount));
    m_txn.NewOrder.exec_status_code = eOK;
}
else
    m_txn.NewOrder.exec_status_code = eInvalidItem;
break;
}
catch (CODBCERR *e)
{
    if (!(e->m_bDeadLock) ||
(++iTryCount > iMaxRetries))
        throw;
    // hit deadlock; backoff for
increasingly longer period
    delete e;
    Sleep(10 * iTryCount);
}
}
// if (iTryCount)
// throw new

```

```

CTPCC_ODBC_ERR(CTPCC_ODBC_ERR::ERR_RETRIED_TRAN
S, iTryCount);
}

void CTPCC_ODBC::InitPaymentParams()
{
    if ( SQLAllocHandle(SQL_HANDLE_STMT,m_hdbc,
&m_hstmtPayment) != SQL_SUCCESS )
        ThrowError(CODBCERR::eAllocHandle);

    m_hstmt = m_hstmtPayment;

    int i = 0;
    if ( SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT,SQL_C_SLONG,SQL_INTEGER,0,0,
&m_txn.Payment.w_id,0, NULL) != SQL_SUCCESS
        || SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT,SQL_C_SLONG,SQL_INTEGER,0,0,
&m_txn.Payment.c_w_id,0, NULL) != SQL_SUCCESS
        || SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT,SQL_C_DOUBLE,SQL_NUMERIC,6,2,
&m_txn.Payment.h_amount,0, NULL) != SQL_SUCCESS
        || SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT,SQL_C_UTINYINT,SQL_TINYINT,0,0,
&m_txn.Payment.d_id,0, NULL) != SQL_SUCCESS
        || SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT,SQL_C_UTINYINT,SQL_TINYINT,0,0,
&m_txn.Payment.c_d_id,0, NULL) != SQL_SUCCESS
        || SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT,SQL_C_SLONG,SQL_INTEGER,0,0,
&m_txn.Payment.c_id,0, NULL) != SQL_SUCCESS
        || SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT,SQL_C_CHAR,SQL_CHAR,
sizeof(m_txn.Payment.c_last),0, &m_txn.Payment.c_last,
sizeof(m_txn.Payment.c_last), NULL) != SQL_SUCCESS
    )
        ThrowError(CODBCERR::eBindParam);

    i = 0;
    if ( SQLBindCol(m_hstmt, ++i,SQL_C_SLONG,
&m_txn.Payment.c_id,
0, NULL) !=
SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.c_last,
sizeof(m_txn.Payment.c_last), NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,
SQL_C_TYPE_TIMESTAMP,&m_txn.Payment.h_date,
0, NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.w_street_1, sizeof(m_txn.Payment.w_street_1),
NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.w_street_2, sizeof(m_txn.Payment.w_street_2),
NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.w_city,
sizeof(m_txn.Payment.w_city), NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.w_state,
sizeof(m_txn.Payment.w_state), NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.w_zip,
sizeof(m_txn.Payment.w_zip), NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&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) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.d_city,
sizeof(m_txn.Payment.d_city), NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.d_state,
sizeof(m_txn.Payment.d_state), NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.d_zip,
sizeof(m_txn.Payment.d_zip), NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.c_first,
sizeof(m_txn.Payment.c_first), NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.c_middle, sizeof(m_txn.Payment.c_middle),
NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.c_street_1, sizeof(m_txn.Payment.c_street_1),
NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.c_street_2, sizeof(m_txn.Payment.c_street_2),
NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.c_city,
sizeof(m_txn.Payment.c_city), NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.c_state,
sizeof(m_txn.Payment.c_state), NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.c_zip,
sizeof(m_txn.Payment.c_zip), NULL) != SQL_SUCCESS
        || 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,
SQL_C_TYPE_TIMESTAMP,&m_txn.Payment.c_since,
0, NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.c_credit, sizeof(m_txn.Payment.c_credit),
NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,
SQL_C_DOUBLE, &m_txn.Payment.c_credit_lim,0, NULL) !=
SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,
SQL_C_DOUBLE, &m_txn.Payment.c_discount, 0, NULL) !=
SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,
SQL_C_DOUBLE, &m_txn.Payment.c_balance, 0, NULL) !=
SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.c_data,
sizeof(m_txn.Payment.c_data), NULL) != SQL_SUCCESS
    )
        ThrowError(CODBCERR::eBindCol);
}

void CTPCC_ODBC::Payment(BOOL fLocalFlag)
{
    RETCODE rc;
    int iTryCount = 0;

    m_hstmt = m_hstmtPayment;

    if (m_txn.Payment.c_id != 0)
        m_txn.Payment.c_last[0] = 0;
}

```

```

        while (TRUE)
        {
            try
            {
                rc = SQLExecDirectW(m_hstmt,
(SQLWCHAR*)"L"{call tpcc_payment(?,?,?,?,?)", SQL_NTS);
                if (rc != SQL_SUCCESS && rc !=
SQL_SUCCESS_WITH_INFO)
                {
#ifdef ErrTrace
                    fprintf(errFile,"Payment
Error\n w_id: %d\nc_w_id: %d\nc_id: %d\nc_last: %s\nd_id:
%d\nc_d_id: %d\nError:\n",

m_txn.Payment.w_id,

m_txn.Payment.c_w_id,

m_txn.Payment.c_id,

m_txn.Payment.c_last,

m_txn.Payment.d_id,

m_txn.Payment.c_d_id);

                    fflush(errFile);

#endif

                ThrowError(CODBCERR::eExecDirect);
            }
            if ( SQLFetch(m_hstmt) ==
SQL_ERROR)

                ThrowError(CODBCERR::eFetch);

                SQLFreeStmt(m_hstmt,
SQL_CLOSE);

                if (m_txn.Payment.c_id == 0)
                    throw new
CTPCC_ODBC_ERR( CTPCC_ODBC_ERR::ERR_INVALID_CUST
);
                else

                    m_txn.Payment.exec_status_code = eOK;

                    break;
                }
            catch (CODBCERR *e)
            {
                if (!(e->m_bDeadLock) ||
(++iTryCount > iMaxRetries))

                    throw;

                // hit deadlock; backoff for
increasingly longer period

                delete e;
                Sleep(10 * iTryCount);
            }
        }

        // if (iTryCount)
        // throw new

CTPCC_ODBC_ERR(CTPCC_ODBC_ERR::ERR_RETRIED_TRAN
S, iTryCount);

```

```

    }

void CTPCC_ODBC::InitOrderStatusParams()
{
    if ( SQLAllocHandle(SQL_HANDLE_STMT,m_hdbc,
&m_hstmtOrderStatus) != SQL_SUCCESS
        || SQLAllocHandle(SQL_HANDLE_DESC,
m_hdbc, &m_descOrderStatusCols1) != SQL_SUCCESS
        || SQLAllocHandle(SQL_HANDLE_DESC,
m_hdbc, &m_descOrderStatusCols2) != SQL_SUCCESS
    )
        ThrowError(CODBCERR::eAllocHandle);

    m_hstmt = m_hstmtOrderStatus;

    if ( SQLSetStmtAttrW(m_hstmt,
SQL_ATTR_APP_ROW_DESC, m_descOrderStatusCols1,
SQL_IS_POINTER) != SQL_SUCCESS )
        ThrowError(CODBCERR::eSetStmtAttr);

    int i = 0;
    if ( SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT,SQL_C_SLONG,SQL_INTEGER,0,0,
&m_txn.OrderStatus.w_id, 0, NULL) != SQL_SUCCESS
        || SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT,SQL_C_UTINYINT,SQL_TINYINT,0,0,
&m_txn.OrderStatus.d_id, 0, NULL) != SQL_SUCCESS
        || SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT,SQL_C_SLONG,SQL_INTEGER,0,0,
&m_txn.OrderStatus.c_id, 0, NULL) != SQL_SUCCESS
        || SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT,SQL_C_CHAR,SQL_CHAR,
sizeof(m_txn.OrderStatus.c_last), 0, &m_txn.OrderStatus.c_last,
sizeof(m_txn.OrderStatus.c_last), NULL) != SQL_SUCCESS
    )
        ThrowError(CODBCERR::eBindParam);

    // configure block cursor
    if ( SQLSetStmtAttrW(m_hstmt,
SQL_ATTR_ROW_BIND_TYPE,
(SQLPOINTER)sizeof(m_txn.OrderStatus.OL[0]),0) !=
SQL_SUCCESS
        || SQLSetStmtAttrW(m_hstmt,
SQL_ATTR_ROWS_FETCHED_PTR, &m_RowsFetched, 0) !=
SQL_SUCCESS
    )
        ThrowError(CODBCERR::eSetStmtAttr);

    i = 0;
    if ( SQLBindCol(m_hstmt, ++i, SQL_C_SLONG,
&m_txn.OrderStatus.OL[0].ol_supply_w_id, 0, NULL) !=
SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i, SQL_C_SLONG,
&m_txn.OrderStatus.OL[0].ol_i_id, 0, NULL) != SQL_SUCCESS
        || 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_txn.OrderStatus.OL[0].ol_amount, 0, NULL)
!= SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,
SQL_C_TYPE_TIMESTAMP,
&m_txn.OrderStatus.OL[0].ol_delivery_d, 0, NULL) !=
SQL_SUCCESS
    )
        ThrowError(CODBCERR::eBindCol);

```

```

        if ( SQLSetStmtAttrW(m_hstmt,
SQL_ATTR_APP_ROW_DESC, m_descOrderStatusCols2,
SQL_IS_POINTER) != SQL_SUCCESS )
            ThrowError(CODBCERR::eSetStmtAttr);

        i = 0;
        if ( SQLBindCol(m_hstmt, ++i, SQL_C_SLONG,
&m_txn.OrderStatus.c_id, 0, NULL) != SQL_SUCCESS
            || SQLBindCol(m_hstmt, ++i, SQL_C_CHAR,
&m_txn.OrderStatus.c_last, sizeof(m_txn.OrderStatus.c_last), NULL)
!= SQL_SUCCESS
            || SQLBindCol(m_hstmt, ++i, SQL_C_CHAR,
&m_txn.OrderStatus.c_first, sizeof(m_txn.OrderStatus.c_first), NULL)
!= SQL_SUCCESS
            || SQLBindCol(m_hstmt, ++i, SQL_C_CHAR,
&m_txn.OrderStatus.c_middle, sizeof(m_txn.OrderStatus.c_middle),
NULL) != SQL_SUCCESS
            || SQLBindCol(m_hstmt, ++i,
SQL_C_TYPE_TIMESTAMP, &m_txn.OrderStatus.o_entry_d, 0,
NULL) != SQL_SUCCESS
            || SQLBindCol(m_hstmt, ++i,
SQL_C_SSHORT, &m_txn.OrderStatus.o_carrier_id, 0, NULL) !=
SQL_SUCCESS
            || SQLBindCol(m_hstmt, ++i,
SQL_C_DOUBLE, &m_txn.OrderStatus.c_balance, 0, NULL) !=
SQL_SUCCESS
            || SQLBindCol(m_hstmt, ++i, SQL_C_SLONG,
&m_txn.OrderStatus.o_id, 0, NULL) != SQL_SUCCESS
        )
            ThrowError(CODBCERR::eBindCol);
    }

void CTPCC_ODBC::OrderStatus()
{
    int
iTryCount = 0;
    RETCODE
rc;

    m_hstmt = m_hstmtOrderStatus;

    if ( SQLSetStmtAttrW(m_hstmt,
SQL_ATTR_APP_ROW_DESC, m_descOrderStatusCols1,
SQL_IS_POINTER) != SQL_SUCCESS )
        ThrowError(CODBCERR::eSetStmtAttr);

    if (m_txn.OrderStatus.c_id != 0)
        m_txn.OrderStatus.c_last[0] = 0;

    while (TRUE)
    {
        try
        {
            // configure block cursor
            if ( SQLSetStmtAttrW(m_hstmt,
SQL_ATTR_ROW_ARRAY_SIZE, (SQLPOINTER)1, 0) !=
SQL_SUCCESS )

                ThrowError(CODBCERR::eSetStmtAttr);

            rc = SQLExecDirectW(m_hstmt,
(SQLWCHAR*)" {call tpcc_orderstatus(?,?,?)}", SQL_NTS);
            if ( ((rc ==
SQL_SUCCESS_WITH_INFO) && (m_RowsFetched != 0)) || (rc ==
SQL_ERROR) )
                {
#ifdef ErrTrace

```

```

                fprintf(errFile, "Order
Status\n w_id: %d\nc_id: %d\nc_last: %s\nd_id: %d\nError:\n",
m_txn.OrderStatus.w_id,
m_txn.OrderStatus.c_id,
m_txn.OrderStatus.c_last,
m_txn.OrderStatus.d_id);

                fflush(errFile);
            #endif

            ThrowError(CODBCERR::eExecDirect);
        }

        // configure block cursor
        if ( SQLSetStmtAttrW(m_hstmt,
SQL_ATTR_ROW_ARRAY_SIZE,
(SQLPOINTER)MAX_OL_ORDER_STATUS_ITEMS, 0) !=
SQL_SUCCESS )

            ThrowError(CODBCERR::eSetStmtAttr);

            rc = SQLFetchScroll( m_hstmt,
SQL_FETCH_NEXT, 0 );
            if ( ((rc ==
SQL_SUCCESS_WITH_INFO) && (m_RowsFetched != 0)) || (rc ==
SQL_ERROR) )

                ThrowError(CODBCERR::eFetchScroll);

                m_txn.OrderStatus.o_ol_cnt =
(short)m_RowsFetched;

                if (m_txn.OrderStatus.o_ol_cnt != 0)
                {
                    if ( SQLSetStmtAttrW(
m_hstmt, SQL_ATTR_APP_ROW_DESC, m_descOrderStatusCols2,
SQL_IS_POINTER) != SQL_SUCCESS )

                        ThrowError(CODBCERR::eSetStmtAttr);

                    if (
SQLMoreResults(m_hstmt) == SQL_ERROR )

                        ThrowError(CODBCERR::eMoreResults);

                    if ( (rc =
SQLFetch(m_hstmt)) == SQL_ERROR )

                        ThrowError(CODBCERR::eFetch);
                }

                SQLFreeStmt(m_hstmt,
SQL_CLOSE);

                if (m_txn.OrderStatus.o_ol_cnt == 0)
                    throw new
CTPCC_ODBC_ERR(
CTPCC_ODBC_ERR::ERR_NO_SUCH_ORDER );
                else if (m_txn.OrderStatus.c_id == 0
&& m_txn.OrderStatus.c_last[0] == 0)
                    throw new
CTPCC_ODBC_ERR( CTPCC_ODBC_ERR::ERR_INVALID_CUST
);

```

```

                else
m_txn.OrderStatus.exec_status_code = eOK;

                break;
            }
            catch (CODBCERR *e)
            {
                if (!(e->m_bDeadLock) ||
(++iTryCount > iMaxRetries))
                    throw;

                // hit deadlock; backoff for
                // increasingly longer period
                delete e;
                Sleep(10 * iTryCount);
            }
        }

//        if (iTryCount)
//            throw new

CTPCC_ODBC_ERR(CTPCC_ODBC_ERR::ERR_RETRIED_TRAN
S, iTryCount);
}

void CTPCC_ODBC::InitDeliveryParams()
{
    if ( SQLAllocHandle(SQL_HANDLE_STMT,m_hdbc,
&m_hstmtDelivery) != SQL_SUCCESS )
        ThrowError(CODBCERR::eAllocHandle);

    m_hstmt = m_hstmtDelivery;

    int i = 0;
    if ( SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT,SQL_C_SLONG,SQL_INTEGER,0,0,
&m_txn.Delivery.w_id,0, NULL) != SQL_SUCCESS
        || SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT,SQL_C_SSHORT,SQL_SMALLINT,0,0,
&m_txn.Delivery.o_carrier_id,0, NULL) != SQL_SUCCESS
        )
        ThrowError(CODBCERR::eBindParam);

    for (i=0;i<10;i++)
    {
        if ( SQLBindCol(m_hstmt, (UWORD)(i+1),
SQL_C_SLONG, &m_txn.Delivery.o_id[i], 0, NULL) !=
SQL_SUCCESS )

        ThrowError(CODBCERR::eBindCol);
    }
}

void CTPCC_ODBC::Delivery()
{
    RETCODE                rc;
    int                    iTryCount = 0;

    m_hstmt = m_hstmtDelivery;

    while (TRUE)
    {
        try
        {
            rc = SQLExecDirectW(m_hstmt,
(SQLWCHAR*)L"{call tpcc_delivery(?,?)", 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.Delivery.exec_status_code =
eOK;
                break;
            }
            catch (CODBCERR *e)
            {
                if (!(e->m_bDeadLock) ||
(++iTryCount > iMaxRetries))
                    throw;

                // hit deadlock; backoff for
                // increasingly longer period
                delete e;
                Sleep(10 * iTryCount);
            }
        }

//        if (iTryCount)
//            throw new

```

```

CTPCC_ODBC_ERR(CTPCC_ODBC_ERR::ERR_RETRIED_TRAN
S, iTryCount);
}

```

### **tpcc\_odbc.h**

```

/*        FILE:                TPCC_ODBC.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

// need to declare functions for import, unless define has already been
// created
// by the DLL's .cpp module for export.
#ifdef DllDecl
#define DllDecl __declspec( dllimport )
#endif

class CODBCERR : public CBaseErr
{
public:

```

```

enum ACTION
{
    eNone,
    eUnknown,
    eAllocConn,
// error from SQLAllocConnect
    eAllocHandle,
error from SQLAllocHandle
    eConnOption,
error from SQLSetConnectOption
    eConnect,
error from SQLConnect
    eAllocStmt,
// error from SQLAllocStmt
    eExecDirect,
error from SQLExecDirect
    eBindParam,
// error from SQLBindParameter
    eBindCol,
error from SQLBindCol
    eFetch,
// error from SQLFetch
    eFetchScroll,
error from SQLFetchScroll
    eMoreResults,
error from SQLMoreResults
    ePrepare,
error from SQLPrepare
    eExecute,
error from SQLExecute
    eSetEnvAttr,
error from SQLSetEnvAttr
    eSetStmtAttr,
error from SQLSetStmtAttr
};

CODBCERR(void)
{
    m_eAction = eNone;
    m_NativeError = 0;
    m_bDeadLock = FALSE;
    m_odbcerrstr = NULL;
};

~CODBCERR()
{
    if (m_odbcerrstr != NULL)
        delete [] m_odbcerrstr;
};

ACTION m_eAction;
int m_NativeError;
BOOL m_bDeadLock;
char *m_odbcerrstr;

int ErrorType() {return ERR_TYPE_ODBC;};
int ErrorNum() {return m_NativeError;};
char *ErrorText() {return m_odbcerrstr;};
};

class CTPCC_ODBC_ERR : public CBaseErr
{
public:
    enum TPCC_ODBC_ERRS
    {
        ERR_WRONG_SP_VERSION = 1,
// "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."
    };

    CTPCC_ODBC_ERR( int iErr ) { m_erno =
iErr; m_iTryCount = 0; };

    CTPCC_ODBC_ERR( int iErr, int iTryCount ) {
m_erno = iErr; m_iTryCount = iTryCount; };

    int m_erno;
    int m_iTryCount;

    int ErrorType() {return
ERR_TYPE_TPCC_ODBC;};
    int ErrorNum() {return m_erno;};

    char *ErrorText();
};

class DllDecl CTPCC_ODBC : public CTPCC_BASE
{
private:
// declare variables and private functions here...
    BOOL m_bDeadlock;
// transaction was selected as deadlock victim
    int m_MaxRetries;
// retry count on deadlock

    SQLHENV m_henv;
// ODBC environment handle
    SQLHDBC m_hdbc;
    SQLHSTMT m_hstmt;
// the current hstmt

    SQLHSTMT m_hstmtNewOrder;
    SQLHSTMT m_hstmtPayment;
    SQLHSTMT m_hstmtDelivery;
    SQLHSTMT m_hstmtOrderStatus;
    SQLHSTMT m_hstmtStockLevel;

    SQLHDESC m_descNewOrderCols1;
    SQLHDESC m_descNewOrderCols2;
    SQLHDESC m_descOrderStatusCols1;
    SQLHDESC m_descOrderStatusCols2;

// new-order specific fields
    SQLINTEGER m_BindOffset;
    SQLINTEGER m_RowsFetched;
    int
m_no_commit_flag;
#ifdef new_order_strstr
// for new-order txn;
// output params
    char
m_ol_i_name[I_NAME_LEN+1];
    double m_ol_i_price;
    double m_ol_amount;
    short m_ol_stock;
// used locally, but not returned to caller
    char m_i_data[I_DATA_LEN];
    char
m_s_data[S_DATA_LEN];
};

```



```

#endif
void ThrowError( CODBCERR::ACTION
eAction );

void InitNewOrderParams();
void InitPaymentParams();
void InitDeliveryParams();
void InitStockLevelParams();
void InitOrderStatusParams();

union
{
    NEW_ORDER_DATA
NewOrder;
    PAYMENT_DATA
Payment;
    DELIVERY_DATA
Delivery;
    STOCK_LEVEL_DATA
StockLevel;
    ORDER_STATUS_DATA
OrderStatus;
} m_txn;

public:
    CTPCC_ODBC(LPCSTR szServer, LPCSTR
szUser, LPCSTR szPassword, LPCSTR szHost, LPCSTR szDatabase);
    ~CTPCC_ODBC(void);

    inline PNEW_ORDER_DATA
BuffAddr_NewOrder() { return
&m_txn.NewOrder; };
    inline PPAYMENT_DATA
BuffAddr_Payment() { return &m_txn.Payment; };
    inline PDELIVERY_DATA
BuffAddr_Delivery() { return &m_txn.Delivery; };
    inline PSTOCK_LEVEL_DATA
BuffAddr_StockLevel() { return &m_txn.StockLevel; };
    inline PORORDER_STATUS_DATA
BuffAddr_OrderStatus() { return &m_txn.OrderStatus; };

    void NewOrder (BOOL
fLocalFlag);
    void Payment (BOOL
fLocalFlag);
    void Delivery ();
    void StockLevel ();
    void OrderStatus ();

};

// wrapper routine for class constructor
extern "C" DllDecl CTPCC_ODBC* CTPCC_ODBC_new
(LPCSTR szServer, LPCSTR szUser, LPCSTR
szPassword, LPCSTR szHost, LPCSTR szDatabase );

typedef CTPCC_ODBC* (TYPE_CTPCC_ODBC)(LPCSTR,
LPCSTR, LPCSTR, LPCSTR, LPCSTR);

trans.h

/* FILE: TRANS.H
* Microsoft TPC-C Kit Ver.
4.30.002

```

```

* 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 structure
templates.
*
* Change history:
* 4.30.002 - changed w_id from short to long to
support warehouses above 32,767
* 4.20.000 - updated rev number to match kit
*/
#pragma once

// String length constants
#define SERVER_NAME_LEN 20
#define DATABASE_NAME_LEN 20
#define USER_NAME_LEN 20
#define PASSWORD_LEN 20
#define TABLE_NAME_LEN 20
#define I_DATA_LEN 50
#define I_NAME_LEN 24
#define BRAND_LEN 1
#define LAST_NAME_LEN 16
#define W_NAME_LEN 10
#define ADDRESS_LEN 20
#define STATE_LEN 2
#define ZIP_LEN 9
#define S_DIST_LEN 24
#define S_DATA_LEN 50
#define D_NAME_LEN 10
#define FIRST_NAME_LEN 16
#define MIDDLE_NAME_LEN 2
#define PHONE_LEN 16
#define DATETIME_LEN 30
#define CREDIT_LEN 2
#define C_DATA_LEN 250
#define H_DATA_LEN 24
#define DIST_INFO_LEN 24
#define MAX_OL_NEW_ORDER_ITEMS 15
#define MAX_OL_ORDER_STATUS_ITEMS 15
#define STATUS_LEN 25
#define OL_DIST_INFO_LEN 24

// TIMESTAMP_STRUCT is provided by the ODBC header file
sqltypes.h, but is not available
// when compiling with dblib, so redefined here. Note: we are using
the symbol "__SQLTYPES"
// (declared in sqltypes.h) as a way to determine if
TIMESTAMP_STRUCT has been declared.
#ifndef __SQLTYPES
typedef struct
{
    short /*
SQLSMALLINT*/ year;
    unsigned short /*
SQLUSMALLINT*/ month;
    unsigned short /*
SQLUSMALLINT*/ day;
    unsigned short /*
SQLUSMALLINT*/ hour;
    unsigned short /*
SQLUSMALLINT*/ minute;
    unsigned short /*
SQLUSMALLINT*/ second;

```

```

        unsigned long      /*
SQLINTEGER */ fraction;
    } TIMESTAMP_STRUCT;
#endif

// possible values for exec_status_code after transaction completes
enum EXEC_STATUS
{
    eOK,                // 0      "Transaction
committed."
    eInvalidItem,      // 1      "Item number is not
valid."
    eDeliveryFailed    // 2      "Delivery Post Failed."
};

// transaction structures
typedef struct
{
    // input params
    long
ol_supply_w_id;
    long                ol_i_id;
    short               ol_quantity;

    // output params
    char
ol_i_name[I_NAME_LEN+1];
    char
ol_brand_generic[BRAND_LEN+1];
    double              ol_i_price;
    double              ol_amount;
    short               ol_stock;
} OL_NEW_ORDER_DATA;

typedef struct
{
    // input params
    long                w_id;
    short               d_id;
    long                c_id;
    short               o_ol_cnt;

    // output params
    EXEC_STATUS
exec_status_code;
    char                c_last[LAST_NAME_LEN+1];
    char                c_credit[CREDIT_LEN+1];
    double              c_discount;
    double              w_tax;
    double              d_tax;
    long                o_id;
    short

o_commit_flag;
    TIMESTAMP_STRUCT   o_entry_d;
    short               o_all_local;
    double              total_amount;
    OL_NEW_ORDER_DATA
OL[MAX_OL_NEW_ORDER_ITEMS];
} NEW_ORDER_DATA, *PNEW_ORDER_DATA;

typedef struct
{
    // input params
    long                w_id;
    short               d_id;
    long                c_id;
    short               c_d_id;
    long                c_w_id;
    double              h_amount;

```

```

        char                c_last[LAST_NAME_LEN+1];

    // output params
    EXEC_STATUS
exec_status_code;
    TIMESTAMP_STRUCT   h_date;
    char
w_street_1[ADDRESS_LEN+1];
    char
w_street_2[ADDRESS_LEN+1];
    char
w_city[ADDRESS_LEN+1];
    char
w_state[STATE_LEN+1];
    char                w_zip[ZIP_LEN+1];
    char
d_street_1[ADDRESS_LEN+1];
    char
d_street_2[ADDRESS_LEN+1];
    char
d_city[ADDRESS_LEN+1];
    char                d_state[STATE_LEN+1];
    char                d_zip[ZIP_LEN+1];
    char
c_first[FIRST_NAME_LEN+1];
    char
c_middle[MIDDLE_NAME_LEN + 1];
    char
c_street_1[ADDRESS_LEN+1];
    char
c_street_2[ADDRESS_LEN+1];
    char
c_city[ADDRESS_LEN+1];
    char                c_state[STATE_LEN+1];
    char                c_zip[ZIP_LEN+1];
    char
c_phone[PHONE_LEN+1];
    TIMESTAMP_STRUCT   c_since;
    char
c_credit[CREDIT_LEN+1];
    double              c_credit_lim;
    double              c_discount;
    double              c_balance;
    char                c_data[200+1];
} PAYMENT_DATA, *PPAYMENT_DATA;

typedef struct
{
    long                ol_i_id;
    long                ol_supply_w_id;
    short               ol_quantity;
    double              ol_amount;
    TIMESTAMP_STRUCT   ol_delivery_d;
} OL_ORDER_STATUS_DATA;

typedef struct
{
    // input params
    long                w_id;
    short               d_id;
    long                c_id;
    char                c_last[LAST_NAME_LEN+1];

    // output params
    EXEC_STATUS
exec_status_code;
    char                c_first[FIRST_NAME_LEN+1];
    char                c_middle[MIDDLE_NAME_LEN+1];
    double              c_balance;

```

```

        long        o_id;
        TIMESTAMP_STRUCT    o_entry_d;
        short       o_carrier_id;
        OL_ORDER_STATUS_DATA
OL[MAX_OL_ORDER_STATUS_ITEMS];
        short       o_ol_cnt;
    } ORDER_STATUS_DATA, *PORDER_STATUS_DATA;

typedef struct
{
    // input params
    long        w_id;
    short       o_carrier_id;

    // output params
    EXEC_STATUS
exec_status_code;
    SYSTEMTIME    queue_time;
    long        o_id[10];
// id's of delivered orders for districts 1 to 10
} DELIVERY_DATA, *PDELIVERY_DATA;

//This structure is used for posting delivery transactions and for writing
them to the delivery server.
typedef struct _DELIVERY_TRANSACTION
{
    SYSTEMTIME    queue;
//time delivery transaction queued
    long        w_id;
//delivery warehouse
    short       o_carrier_id;    //carrier id
} DELIVERY_TRANSACTION;

typedef struct
{
    // input params
    long        w_id;
    short       d_id;
    short       threshold;

    // output params
    EXEC_STATUS
exec_status_code;
    long        low_stock;
} STOCK_LEVEL_DATA, *PSTOCK_LEVEL_DATA;

```

### **txn\_base.h**

```

/*      FILE:          TXN_BASE.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

```

```

// need to declare functions for import, unless define has already been
created
// by the DLL's .cpp module for export.
#ifdef DllDecl
#define DllDecl __declspec( dllimport )
#endif

class DllDecl CTPCC_BASE
{
public:
    CTPCC_BASE(void) {} ;
    virtual ~CTPCC_BASE(void) {} ;

    virtual PNEW_ORDER_DATA
BuffAddr_NewOrder() = 0;
    virtual PPAYMENT_DATA
BuffAddr_Payment() = 0;
    virtual PDELIVERY_DATA
BuffAddr_Delivery() = 0;
    virtual PSTOCK_LEVEL_DATA
BuffAddr_StockLevel() = 0;
    virtual PORDER_STATUS_DATA
BuffAddr_OrderStatus() = 0;
// virtual void NewOrder          () =
0;
    virtual void NewOrder
(BOOL fLocal) = 0;
// virtual void Payment          () = 0;
    virtual void Payment          (BOOL fLocal)
= 0;
    virtual void Delivery          () = 0;
    virtual void StockLevel        () =
0;
    virtual void OrderStatus       () = 0;
};

```

### **txnlog.h**

```

/*      FILE:          TXNLOG.H
*
*      Microsoft TPC-C Kit Ver.
4.10.000
*
*      not yet audited
*
*      PURPOSE:      Header file for txn log class
*
*      Copyright Microsoft,
1999
*      All Rights Reserved
*
*/

#pragma once

typedef struct _TXN_NEWORDER
{
    BYTE        OL_Count;    //range 0 to 31
    BYTE        OL_Remote_Count; //range 0 to 31
    WORD        c_id;
    int         o_id;
} TXN_NEWORDER;

typedef struct _TXN_PAYMENT
{
    BYTE        CustByName;
    BYTE        IsRemote;
} TXN_PAYMENT;

```

```

typedef struct _TXN_ORDERSTATUS
{
    BYTE    CustByName;
} TXN_ORDERSTATUS;

typedef union _TXN_DETAILS
{
    TXN_NEWORDER  NewOrder;
    TXN_PAYMENT   Payment;
    TXN_ORDERSTATUS OrderStatus;
} TXN_DETAILS;

// Common header for all records in txn log. The TxnType
field is // a switch which identifies the particular variant.
// #define TXN_REC_TYPE_CONTROL 1
// #define TXN_REC_TYPE_TPCC 2
// replaces TRANSACTION_TYPE_TPCC
// #define TXN_REC_TYPE_TPCC_DELIV_DEF 3

typedef struct _TXN_RECORD_HEADER
{
    JULIAN_TIME    TxnStartT0;
// start of txn
    BYTE    TxnType;
// one of TXN_REC_TYPE_*
    BYTE    TxnSubType;
// depends on TxnType
} TXN_RECORD_HEADER,
*PTXN_RECORD_HEADER;

typedef struct _TXN_RECORD_CONTROL
{
    // common header; must exactly match
    TXN_RECORD_HEADER
    JULIAN_TIME    TxnStartT0;
// start of txn
    BYTE    TxnType;
// TXN_REC_TYPE_CONTROL
    BYTE    TxnSubType;
// depends on TxnType
// end of common header

    DWORD    Len;
// number of bytes after this field
} TXN_RECORD_CONTROL,
*PTXN_RECORD_CONTROL;

// TPC-C Txn Record Layout:
//
//TxnStartT0' is a Julian timestamp corresponding to the moment
the
//txn is sent to the SUT, i.e., beginning of response time. Deltas
//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.
//
//Graphically:
//
// time -->
//
// |--- Menu ---|--- Keying ---|--- Response ---|--- Think ---|
// <- DeltaT1 -> <- DeltaT2 -> <- DeltaT4 -> <- DeltaT3 ->

```

```

//
//      ^
//      ^ 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
//reflect this.

typedef struct _TXN_RECORD_TPCC
{
    // common header; must exactly match
    TXN_RECORD_HEADER
    JULIAN_TIME    TxnStartT0;
// start of txn
    BYTE    TxnType;
// =
    TXN_REC_TYPE_TPCC
    BYTE    TxnSubType;
// depends on TxnType
// end of common header

    int    DeltaT1;
// menu time
    int    DeltaT2;
// keying time
    int    DeltaT3;
// think time
    int    DeltaT4;
// response
    int    RTDelay;
// response
    int    TxnError;
// error code providing more detail for TxnStatus
    int    w_id;
// warehouse ID
    BYTE    d_id;
// assigned district ID for this thread
    BYTE    d_id_ThisTxn;
// district ID
    BYTE    TxnStatus;
// completion
    BYTE    reserved;
// status for txn to indicate errors
// for word
    TXN_DETAILS    TxnDetails;
// alignment
} 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 bytes.
//Hence, we store delivery transaction details in a separate structure.
//
typedef struct _TXN_RECORD_TPCC_DELIV_DEF
{
    // common header; must exactly match
    TXN_RECORD_HEADER
    JULIAN_TIME    TxnStartT0;
// start of txn
    BYTE    TxnType;
// =
    TXN_REC_TYPE_TPCC_DELIV_DEF
    BYTE    TxnSubType;
// = 0
// end of common header

```

```

int    DeltaT4;           // response
time (ms)

int    DeltaTxnExec;     //
execution time (ms)

int    w_id;
// warehouse ID

BYTE   TxnStatus;       // completion
status for txn to indicate errors

BYTE   reserved;        // for word
alignment

short  o_carrier_id;    // carrier id
long   o_id[10];        // returned
delivery transaction ids
        } TXN_RECORD_TPCC_DELIV_DEF,
*PTXN_RECORD_TPCC_DELIV_DEF;

#define TXN_LOG_VERSION          2
#define TXN_DATA_START          4096 //
offset in log file where log records start
#define TXN_LOG_EYE_CATCHER     "BC" //
signature bytes at the start of log file

//
// The transaction log has a header as the first 4K block.
//
typedef struct _TXN_LOG_HEADER
{
    char    EyeCatcher[2];
// signature bytes; should always be "BC"
    int
LogVersion;           // set to TXN_LOG_VERSION
                    JULIAN_TIME    BeginTxnTS;
// timestamp of first (lowest) txn start
                    JULIAN_TIME    EndTxnTS;
// timestamp of last (highest) txn completion time
    int
iRecCount;           // number of records in log file
    BOOL           bLogSorted;
    int
iFileSize;           // file size in bytes

// the record map provides a fast way to get close
to a particular timestamp in a sorted log file.
//
//
//           JULIAN_TIME           TS;
// timestamp of record
//           int
iPos;           // byte position in file
//
//
RecMap[RecMapSize];
// #define RecMapSize           200
        } TXN_LOG_HEADER, *PTXN_LOG_HEADER;

#define READ_BUFFER_SIZE          64*1024
#define WRITE_BUFFER_SIZE         8*1024

#define NUM_READ_BUFFERS          1
#define NUM_WRITE_BUFFERS         2
#define MAX_NUM_BUFFERS           2

// flags passed in to the constructor
#define TXN_LOG_WRITE             0x01
#define TXN_LOG_READ              0x02
#define TXN_LOG_SORTED           0x04
#define TXN_LOG_CRASHOPEN        0x08 // if
set, invalid headers will be tolerated; used for recovery

#define TXN_LOG_OS_ERROR          1
#define TXN_LOG_NOT_SORTED       2

#define SKIP_CTRL_RECS           1

class CTxnLog
{
    private:
        DWORD           iBufferSize;
//buffer allocated size
        DWORD           iBytesFreeInBuffer;
//total bytes available for use in buffer
        int             iNumBuffers;
//buffers in use
        int             iActiveBuffer;
//indicates which buffer is active: 0 or 1
        int             iIoBuffer;
//buffer for any pending IO operation
        int             iFilePointer;
//position in file.
        LARGE_INTEGER   IFilePointer;
//position in file.
        int             iNextRec;
//when reading, ordinal value of next record

// A "save point" is remembered each time
getNextRecord is called with a start time specified.
// The next time it is called, if start time is after
the save point, we start scanning from the
// save point. This is particularly useful in
FindBestInterval, where the log is scanned repeatedly.
        JULIAN_TIME     SavePtTime;
        int
iSavePtFilePointer;
        LARGE_INTEGER   ISavePtFilePointer;
        int
iSavePtNextRec;

        JULIAN_TIME     lastTS;
//when writing sorted output, used to verify
records are sorted
        BOOL           bWrite;
//writing log file
        BOOL           bCrashOpen;
//tolerate bad headers and consistency checks

        BOOL           bLogSorted;
// is log file sorted? applies to both input and
output
        JULIAN_TIME     BeginTxnTS;
// timestamp of first (lowest) txn start
        JULIAN_TIME     EndTxnTS;
// timestamp of last (highest) txn completion
time
        int
iRecCount;           // number of records in log
file
        BYTE           *pCurrent;
//ptr to current buffer

```

```

        BYTE
*pBuffer[MAX_NUM_BUFFERS];

        PTXN_RECORD_HEADER *TxnArray;
//transaction record pointer array for sort

        DWORD dwError;
HANDLE hTxnFile;
//handle to log file
HANDLE hMapFile;
//map file used when sorting the log
HANDLE hIoComplete;
//event to signify that there are no pending IOs
HANDLE hLogFileIo;
//event to signal the IO thread to write the
inactive buffer

        Spinlock Spin;
//spin lock to protect the txn log file buffers

        int Write(BYTE *ptr, DWORD Size);
static void LogFileIO(CTXNLog*);

public:
        CTXNLog::CTXNLog(LPCTSTRszFileName,
DWORD dwOpts);
        ~CTXNLog(void);

        int WriteToLog(PTXN_RECORD_TPCC
pTxnRcrd);
        int
WriteToLog(PTXN_RECORD_TPCC_DELIV_DEFpTxnRcrd);
        int WriteToLog(PTXN_RECORD_CONTROL
pCtrlRec);
        int WriteToLog(PTXN_RECORD_HEADER
pCtrlRec);

        int WriteCtrlRecToLog(BYTE SubType,
LPTSTR lpStr, DWORD dwLen);

        void CloseTransactionLogFile(void);

        PTXN_RECORD_HEADER
GetNextRecord(BOOL bSkipCtrlRecs = FALSE);
        PTXN_RECORD_HEADER
GetNextRecord(JULIAN_TIMESeekTimeT0, BOOL bSkipCtrlRecs =
FALSE);

        int Sort(void);
        PTXN_RECORD_HEADER
GetSortedRecord(int index);

        inline BOOL IsSorted(void) { return
bLogSorted; };
        inline JULIAN_TIME BeginTS(void) { return
BeginTxnTS; };
        inline JULIAN_TIME EndTS(void) { return
EndTxnTS; };
        inline int RecordCount(void) { return
iRecCount; };
};

class CTXNLOG_ERR : public CBaseErr
{
public:
        enum CTXNLOG_ERRS
{

```

```

        ERR_BAD_FILE_FORMAT,
// "File format is invalid."
        ERR_UNKNOWN_LOG_VERSION,
// "Log file version is unknown."
        ERR_BROKEN_LOG_FILE,
// "Log file is broken."
        ERR_LOG_NOT_SORTED,
// "Log file is not sorted"
        ERR_INVALID_TIME_SEQ,
// "Internal Error: Record Time Sequence invalid."
};

        CTXNLOG_ERR(int iErr) : CBaseErr(iErr) {};

        int ErrorType() {return
ERR_TYPE_TXNLOG;};

        char *ErrorText()
{
        static char *szMsgs[] = {
                "File format is invalid.",
                "Log file version is
unknown.",
                "Log file is broken.",
                "Log file is not sorted",
                "Internal Error: Record
Time Sequence invalid.",
                ""
        };

        for(int i = 0; szMsgs[i][0]; i++)
        {
                if ( m_idMsg == i )
                        break;
        }

        return(szMsgs[i][0] ? szMsgs[i] :
ERR_UNKNOWN);
};
};

```

## Stored Procedures

### *delivery.sql*

```

-- File: DELIVERY.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.10
-- Copyright Microsoft, 1999
-- Purpose: Creates delivery transaction stored procedure

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONS off
go

use tpcc
go

if exists (select name from sysobjects where name = 'tpcc_delivery' )
drop procedure tpcc_delivery
go

create proc tpcc_delivery @w_id int,
                        @o_carrier_id smallint

```

```

as

declare @d_id tinyint,
        @o_id int,
        @c_id int,
        @total numeric(12,2),
        @oid1 int,
        @oid2 int,
        @oid3 int,
        @oid4 int,
        @oid5 int,
        @oid6 int,
        @oid7 int,
        @oid8 int,
        @oid9 int,
        @oid10 int,
        @time datetime

begin tran d

        select @d_id = 0,
               @time = getdate()

        while (@d_id < 10)
        begin
                select @d_id = @d_id + 1,
                       @total = 0,
                       @o_id = 0

                select top 1
                       @o_id = no_o_id
                from new_order (serializable uplock)
                where no_w_id = @w_id and
                       no_d_id = @d_id
                order by no_o_id asc

                if (@@rowcount <> 0)
                begin
-- claim the order for this district

                        delete new_order
                        where no_w_id = @w_id and
                               no_d_id = @d_id and
                               no_o_id = @o_id

-- set carrier_id on this order (and get customer id)

                        update orders
                        set o_carrier_id =
@o_carrier_id,
                                @c_id =
o_c_id
                                where o_w_id =
@w_id and
                                        o_d_id =
@d_id and
                                                o_id =
@o_id

-- set date in all lineitems for this order (and sum amounts)

                        update order_line
                        set ol_delivery_d =
@time,
                                @total =
@total + ol_amount

```

```

                                where ol_w_id =
@w_id and
                                        ol_d_id =
@d_id and
                                                ol_o_id =
@o_id

-- accumulate lineitem amounts for this order into customer

                                update customer
                                set c_balance = c_balance +
@total,
                                        c_delivery_cnt =
c_delivery_cnt + 1
                                where c_w_id =
@w_id and
                                        c_d_id =
@d_id and
                                                c_id =
@c_id

                                end

                                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,
                                        @oid3 = case @d_id when 3 then @o_id else @oid3 end,
                                        @oid4 = case @d_id when 4 then @o_id else @oid4 end,
                                        @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,
                                        @oid8 = case @d_id when 8 then @o_id else @oid8 end,
                                        @oid9 = case @d_id when 9 then @o_id else @oid9 end,
                                        @oid10 = case @d_id when 10 then @o_id else @oid10 end

                                end

commit tran d

-- return delivery data to client

select @oid1,
        @oid2,
        @oid3,
        @oid4,
        @oid5,
        @oid6,
        @oid7,
        @oid8,
        @oid9,
        @oid10
go

neword.sql

-- File: NEWORD.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.30.000
-- Copyright Microsoft, 1999
-- Purpose: Creates new order transaction stored procedure
--
-- Interface Level: 4.10.000

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONS off
go

```

```

use tpcc
go

if exists ( select name from sysobjects where name = 'tpcc_neworder' )
    drop procedure tpcc_neworder
go

create proc tpcc_neworder
    @w_id int,
    @d_id tinyint,
    @c_id int,
    @o_ol_cnt tinyint,
    @o_all_local tinyint,
    @i_id1 int = 0, @ol_qty1 smallint = 0,
    @i_id2 int = 0, @ol_qty2 smallint = 0,
    @i_id3 int = 0, @ol_qty3 smallint = 0,
    @i_id4 int = 0, @ol_qty4 smallint = 0,
    @i_id5 int = 0, @ol_qty5 smallint = 0,
    @i_id6 int = 0, @ol_qty6 smallint = 0,
    @i_id7 int = 0, @ol_qty7 smallint = 0,
    @i_id8 int = 0, @ol_qty8 smallint = 0,
    @i_id9 int = 0, @ol_qty9 smallint = 0,
    @i_id10 int = 0, @ol_qty10 smallint = 0,
    @i_id11 int = 0, @ol_qty11 smallint = 0,
    @i_id12 int = 0, @ol_qty12 smallint = 0,
    @i_id13 int = 0, @ol_qty13 smallint = 0,
    @i_id14 int = 0, @ol_qty14 smallint = 0,
    @i_id15 int = 0, @ol_qty15 smallint = 0

as
declare @w_tax numeric(4,4),
    @d_tax numeric(4,4),
    @c_last char(16),
    @c_credit char(2),
    @c_discount numeric(4,4),
    @i_price numeric(5,2),
    @i_name char(24),
    @i_data char(50),
    @o_entry_d datetime,
    @remote_flag int,
    @s_quantity smallint,
    @s_data char(50),
    @s_dist char(24),
    @li_no int,
    @o_id int,
    @commit_flag tinyint,

```

```

    @li_id int,
    @li_s_w_id int,
    @li_qty smallint,
    @ol_number int,
    @c_id_local int

begin
begin transaction n
-- get district tax and next available order id and update
-- plus initialize local variables
    update district
    set @d_tax = d_tax,
        @o_id = d_next_o_id,
        d_next_o_id = d_next_o_id + 1,
        @li_no = 0,
        @commit_flag = 1
    where d_w_id = @w_id and
        d_id = @d_id

-- process orderlines
    while (@li_no < @o_ol_cnt)
    begin
        select @li_no = @li_no + 1

-- set i_id, s_w_id, and qty for this lineitem
        select @li_id = case @li_no
            when 1 then @i_id1
            when 2 then @i_id2
            when 3 then @i_id3
            when 4 then @i_id4
            when 5 then @i_id5
            when 6 then @i_id6
            when 7 then @i_id7
            when 8 then @i_id8
            when 9 then @i_id9
            when 10 then @i_id10
            when 11 then @i_id11
            when 12 then @i_id12
            when 13 then @i_id13
            when 14 then @i_id14
            when 15 then @i_id15
            end,

        @li_s_w_id = case @li_no
            when 1 then @s_w_id1
            when 2 then @s_w_id2
            when 3 then @s_w_id3
            when 4 then @s_w_id4
            when 5 then @s_w_id5
            when 6 then @s_w_id6
            when 7 then @s_w_id7
            when 8 then @s_w_id8
            when 9 then @s_w_id9

```



```

when 10 then
@s_w_id10
when 11 then
@s_w_id11
when 12 then
@s_w_id12
when 13 then
@s_w_id13
when 14 then
@s_w_id14
when 15 then
@s_w_id15
end,

-- get item data (no one updates item)
select
    @i_price = i_price,
    @i_name = i_name,
    @i_data = i_data
from
    item (tablock repeatableread)
where
    i_id = @li_id

-- update stock values
update
    stock
set
    s_ytd = s_ytd +
@s_li_qty,
@s_quantity = s_quantity - @li_qty +
case when (s_quantity - @li_qty < 10) then 91 else 0 end,
    s_order_cnt = s_order_cnt + 1,
    s_remote_cnt = s_remote_cnt +
+ case when (@li_s_w_id = @w_id) then 0 else 1 end,
@s_data = s_data,
@s_dist = case @d_id
when 1 then
s_dist_01
when 2 then
s_dist_02
when 3 then
s_dist_03
when 4 then
s_dist_04
when 5 then
s_dist_05
when 6 then
s_dist_06
when 7 then
s_dist_07
when 8 then
s_dist_08
when 9 then
s_dist_09
when 10 then
s_dist_10
end
where
    s_i_id = @li_id and
    s_w_id = @li_s_w_id

-- if there actually is a stock (and item) with these ids, go to work
if (@@rowcount > 0)
begin
-- insert order_line data (using data from item and stock)
insert into order_line values(@o_id,
@d_id,
@w_id,
@li_no,
@li_id,
@li_s_w_id,
'dec 31, 1899',
@li_qty,
@i_price * @li_qty,
@s_dist)

-- send line-item data to client
select
    @i_name,
    @s_quantity,
    case when (
(patindex('%ORIGINAL%',@i_data) > 0) and
(patindex('%ORIGINAL%',@s_data) > 0) )
then 'B' else 'G' end,
    @i_price,
    @i_price * @li_qty
end
else
begin
-- no item (or stock) found - triggers rollback condition
select ",0",0,0
select @commit_flag = 0
end
end

-- get customer last name, discount, and credit rating
select
    @c_last = c_last,

```

```

        @c_discount = c_discount,
        @c_credit = c_credit,
        @c_id_local = c_id,
        @o_entry_d = getdate()
from customer (repeatableread)
where c_id = @c_id and
       c_w_id = @w_id and
       c_d_id = @d_id

-- insert fresh row into orders table

insert into orders values ( @o_id,
                           @d_id,
                           @w_id,
                           @c_id_local,
                           @o_entry_d,
                           0,
                           @o_ol_cnt,
                           @o_all_local)

-- insert corresponding row into new-order table

insert into new_order values ( @o_id,
                               @d_id,
                               @w_id)

-- select warehouse tax

select @w_tax = w_tax
from warehouse (repeatableread)
where w_id = @w_id

if (@commit_flag = 1)
    commit transaction n
else

-- all that work for nuthin!!!

rollback transaction n

-- return order data to client

select @w_tax,
       @d_tax,
       @o_id,
       @c_last,
       @c_discount,
       @c_credit,
       @o_entry_d,
       @commit_flag

end

go

```

## ***ordstat.sql***

```

-- File: ORDSTAT.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.10
-- Copyright Microsoft, 1999
-- Purpose: Creates order status transaction stored procedure

```

```

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONS off
go

```

```

use tpcc
go

if exists ( select name from sysobjects where name = 'tpcc_orderstatus' )
    drop procedure tpcc_orderstatus
go

create proc tpcc_orderstatus @w_id int,
                             @d_id tinyint,
                             @c_id int,
                             @c_last char(16) = ''
as

declare @c_balance numeric(12,2),
        @c_first char(16),
        @c_middle char(2),
        @o_id int,
        @o_entry_d datetime,
        @o_carrier_id smallint,
        @cnt smallint

begin tran o

if (@c_id = 0)
    begin

-- get customer id and info using last name

select @cnt = (count(*)+1)/2
from customer (repeatableread)
where c_last = @c_last and
       c_w_id = @w_id and
       c_d_id = @d_id

set rowcount @cnt

select @c_id = c_id,
       @c_balance = c_balance,
       @c_first = c_first,
       @c_last = c_last,
       @c_middle = c_middle
from customer (repeatableread)
where c_last = @c_last and
       c_w_id = @w_id and
       c_d_id = @d_id

order by c_w_id, c_d_id, c_last, c_first

set rowcount 0

end

else

begin

-- get customer info if by id

select @c_balance = c_balance,
       @c_first = c_first,
       @c_middle = c_middle,
       @c_last = c_last
from customer (repeatableread)
where c_id = @c_id and
       c_d_id = @d_id and
       c_w_id = @w_id

```

```

                select      @cnt      = @@rowcount
            end
-- if no such customer
        if (@cnt = 0)
            begin
                raiserror('Customer not found',18,1)
                goto custnotfound
            end
-- get order info
        select      @o_id      = o_id,
                   @o_entry_d  = o_entry_d,
                   @o_carrier_id = o_carrier_id
        from        orders (serializable)
        where       o_c_id      = @c_id and
                   o_d_id      = @d_id and
                   o_w_id      = @w_id
        order       by o_id asc
-- select order lines for the current order
        select      ol_supply_w_id,
                   ol_i_id,
                   ol_quantity,
                   ol_amount,
                   ol_delivery_d
        from        order_line (repeatableread)
        where       ol_o_id = @o_id and
                   ol_d_id = @d_id and
                   ol_w_id = @w_id
custnotfound:
commit tran o
-- return data to client
select      @c_id,
           @c_last,
           @c_first,
           @c_middle,
           @o_entry_d,
           @o_carrier_id,
           @c_balance,
           @o_id
go

```

## **payment.sql**

```

-- File:  PAYMENT.SQL
--      Microsoft TPC-C Benchmark Kit Ver. 4.10
--      Copyright Microsoft, 1999
-- Purpose:  Creates payment transaction stored procedure
set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONS off
go

```

```

use tpcc
go
if exists (select name from sysobjects where name = 'tpcc_payment')
    drop procedure tpcc_payment
go
create proc tpcc_payment      @w_id      int,
                             @c_w_id    int,
                             @h_amount  numeric(6,2),
                             @d_id      tinyint,
                             @c_d_id    tinyint,
                             @c_id      int,
                             @c_last   char(16) = "
as
declare @w_street_1 char(20),
        @w_street_2 char(20),
        @w_city     char(20),
        @w_state    char(2),
        @w_zip      char(9),
        @w_name     char(10),
        @d_street_1 char(20),
        @d_street_2 char(20),
        @d_city     char(20),
        @d_state    char(2),
        @d_zip      char(9),
        @d_name     char(10),
        @c_first   char(16),
        @c_middle  char(2),
        @c_street_1 char(20),
        @c_street_2 char(20),
        @c_city    char(20),
        @c_state   char(2),
        @c_zip     char(9),
        @c_phone   char(16),
        @c_since   datetime,
        @c_credit  char(2),
        @c_credit_lim numeric(12,2),
        @c_balance numeric(12,2),
        @c_discount numeric(4,4),
        @data     char(500),
        @c_data   char(500),
        @datetime datetime,
        @w_ytd   numeric(12,2),
        @d_ytd   numeric(12,2),
        @cnt     smallint,
        @val     smallint,
        @screen_data char(200),
        @d_id_local tinyint,
        @w_id_local int,
        @c_id_local int
select @screen_data = "
begin tran p
-- get payment date
        select      @datetime = getdate()
        if (@c_id = 0)
            begin
-- get customer id and info using last name

```

```

select top 1 @c_id = c_id
from (
    select top 50 percent c_id, c_first
    from customer (repeatable read)
    where c_last = @c_last and
          c_w_id = @c_w_id
    and
          c_d_id = @c_d_id
    order by c_first
) top_fifty
order by c_first desc
end

-- get customer info and update balances

update customer
set @c_balance = c_balance -
@c_h_amount,
    c_payment_cnt = c_payment_cnt + 1,
    c_ytd_payment = c_ytd_payment +
@c_h_amount,
    @c_first = c_first,
    @c_middle = c_middle,
    @c_last = c_last,
    @c_street_1 = c_street_1,
    @c_street_2 = c_street_2,
    @c_city = c_city,
    @c_state = c_state,
    @c_zip = c_zip,
    @c_phone = c_phone,
    @c_credit = c_credit,
    @c_credit_lim = c_credit_lim,
    @c_discount = c_discount,
    @c_since = c_since,
    @data = c_data,
    @c_id_local = c_id
where c_id = @c_id and
      c_w_id = @c_w_id and
      c_d_id = @c_d_id

-- if customer has bad credit get some more info

if (@c_credit = 'BC')
begin
-- compute new info

select @c_data = convert(char(5),@c_id)
+
convert(char(4),@c_d_id) +
convert(char(5),@c_w_id) +
+
convert(char(4),@d_id)
+
convert(char(5),@w_id)
+
convert(char(19),@h_amount) +
substring(@data, 1, 458)

-- update customer info

update customer
set c_data = @c_data

```

```

where c_id = @c_id and
      c_w_id = @c_w_id and
      c_d_id = @c_d_id

select @screen_data = substring
(@c_data,1,200)
end

-- get district data and update year-to-date

update district
set d_ytd = d_ytd + @h_amount,
    @d_street_1 = d_street_1,
    @d_street_2 = d_street_2,
    @d_city = d_city,
    @d_state = d_state,
    @d_zip = d_zip,
    @d_name = d_name,
    @d_id_local = d_id
where d_w_id = @w_id and
      d_id = @d_id

-- get warehouse data and update year-to-date

update warehouse
set w_ytd = w_ytd + @h_amount,
    @w_street_1 = w_street_1,
    @w_street_2 = w_street_2,
    @w_city = w_city,
    @w_state = w_state,
    @w_zip = w_zip,
    @w_name = w_name,
    @w_id_local = w_id
where w_id = @w_id

-- create history record

insert into history values (
    @c_id_local,
    @c_d_id,
    @c_w_id,
    @d_id_local,
    @w_id_local,
    @datetime,
    @h_amount,
    @w_name +
' + @d_name)
commit tran p

-- return data to client

select @c_id,
       @c_last,
       @datetime,
       @w_street_1,
       @w_street_2,
       @w_city,
       @w_state,
       @w_zip,
       @d_street_1,
       @d_street_2,
       @d_city,
       @d_state,
       @d_zip,
       @c_first,
       @c_middle,
       @c_street_1,
       @c_street_2,

```

```
@c_city,  
@c_state,  
@c_zip,  
@c_phone,  
@c_since,  
@c_credit,  
@c_credit_lim,  
@c_discount,  
@c_balance,  
@screen_data
```

```
go
```

## **stocklev.sql**

```
-- File: STOCKLEV.SQL  
-- Microsoft TPC-C Benchmark Kit Ver. 4.10  
-- Copyright Microsoft, 1999  
-- Purpose: Creates stock level transaction stored procedure
```

```
set ANSI_DEFAULTS on  
set IMPLICIT_TRANSACTIONS off  
go
```

```
use tpcc  
go
```

```
if exists (select name from sysobjects where name = 'tpcc_stocklevel' )  
    drop procedure tpcc_stocklevel  
go
```

```
create proc tpcc_stocklevel @w_id int,  
                            @d_id tinyint,  
                            @threshold smallint  
as
```

```
declare @o_id_low int,  
        @o_id_high int
```

```
select @o_id_low = (d_next_o_id - 20),  
       @o_id_high = (d_next_o_id - 1)  
from district  
where d_w_id = @w_id and  
       d_id = @d_id
```

```
select count(distinct(s_i_id))  
from stock, order_line  
where ol_w_id = @w_id and  
       ol_d_id = @d_id and  
       ol_o_id between @o_id_low and  
                @o_id_high and  
       s_w_id = ol_w_id and  
       s_i_id = ol_i_id and  
       s_quantity < @threshold
```

```
option (order group)
```

```
go
```

# Appendix B: Database Design

## Database Build

### 4x2492

1	sgnode01	1	2492
2	sgnode02	2493	4984
3	sgnode03	4985	7476
4	sgnode04	7477	9968

### add\_constraints\_t1.sql

```
--file 1_to_2492\add_constraints_t.sql

set ansi_warnings on
set ansi_nulls on
go

use tpcc
go

-- Drop any existing partitioning constraints
--

alter table t_warehouse drop constraint cnst_w_id
go

alter table t_district drop constraint cnst_d_w_id
go

alter table t_customer drop constraint cnst_c_w_id
go

alter table t_history drop constraint cnst_h_w_id
go

alter table t_stock drop constraint cnst_s_w_id
go

alter table t_orders drop constraint cnst_o_w_id
go

alter table t_order_line drop constraint cnst_ol_w_id
go

alter table t_new_order drop constraint cnst_no_w_id
go

-- Add partitioning constraints between 1 and 2492
--

alter table t_warehouse add constraint cnst_w_id check (w_id <=
convert(int,2492))
go

alter table t_district add constraint cnst_d_w_id check (d_w_id <=
convert(int,2492))
go

alter table t_customer add constraint cnst_c_w_id check (c_w_id <=
convert(int,2492))
go
```

```
alter table t_history add constraint cnst_h_w_id check (h_w_id <=
convert(int,2492))
go
```

```
alter table t_stock add constraint cnst_s_w_id check (s_w_id <=
convert(int,2492))
go
```

```
alter table t_orders add constraint cnst_o_w_id check (o_w_id <=
convert(int,2492))
go
```

```
alter table t_order_line add constraint cnst_ol_w_id check (ol_w_id <=
convert(int,2492))
go
```

```
alter table t_new_order add constraint cnst_no_w_id check (no_w_id
<= convert(int,2492))
go
```

### add\_constraints\_t2.sql

```
--file 2493_to_4984\add_constraints_t.sql

set ansi_warnings on
set ansi_nulls on
go

use tpcc
go

-- Drop any existing partitioning constraints
--

alter table t_warehouse drop constraint cnst_w_id
go

alter table t_district drop constraint cnst_d_w_id
go

alter table t_customer drop constraint cnst_c_w_id
go

alter table t_history drop constraint cnst_h_w_id
go

alter table t_stock drop constraint cnst_s_w_id
go

alter table t_orders drop constraint cnst_o_w_id
go

alter table t_order_line drop constraint cnst_ol_w_id
go

alter table t_new_order drop constraint cnst_no_w_id
go

-- Add partitioning constraints between 2493 and 4984
--

alter table t_warehouse add constraint cnst_w_id check (w_id between
convert(int,2493) and convert(int,4984))
go
```

```

alter table t_district add constraint cnst_d_w_id check (d_w_id
between convert(int,2493) and convert(int,4984))
go

alter table t_customer add constraint cnst_c_w_id check (c_w_id
between convert(int,2493) and convert(int,4984))
go

alter table t_history add constraint cnst_h_w_id check (h_w_id
between convert(int,2493) and convert(int,4984))
go

alter table t_stock add constraint cnst_s_w_id check (s_w_id between
convert(int,2493) and convert(int,4984))
go

alter table t_orders add constraint cnst_o_w_id check (o_w_id between
convert(int,2493) and convert(int,4984))
go

alter table t_order_line add constraint cnst_ol_w_id check (ol_w_id
between convert(int,2493) and convert(int,4984))
go

alter table t_new_order add constraint cnst_no_w_id check (no_w_id
between convert(int,2493) and convert(int,4984))
go

```

### ***add\_constraints\_t3.sql***

```

--file 2493_to_4984\add_constraints_t.sql

set ansi_warnings on
set ansi_nulls on
go

use tpcc
go

-- Drop any existing partitioning constraints
--

alter table t_warehouse drop constraint cnst_w_id
go

alter table t_district drop constraint cnst_d_w_id
go

alter table t_customer drop constraint cnst_c_w_id
go

alter table t_history drop constraint cnst_h_w_id
go

alter table t_stock drop constraint cnst_s_w_id
go

alter table t_orders drop constraint cnst_o_w_id
go

alter table t_order_line drop constraint cnst_ol_w_id
go

alter table t_new_order drop constraint cnst_no_w_id
go

```

```

-- Add partitioning constraints between 2493 and 4984
--

alter table t_warehouse add constraint cnst_w_id check (w_id between
convert(int,2493) and convert(int,4984))
go

alter table t_district add constraint cnst_d_w_id check (d_w_id
between convert(int,2493) and convert(int,4984))
go

alter table t_customer add constraint cnst_c_w_id check (c_w_id
between convert(int,2493) and convert(int,4984))
go

alter table t_history add constraint cnst_h_w_id check (h_w_id
between convert(int,2493) and convert(int,4984))
go

alter table t_stock add constraint cnst_s_w_id check (s_w_id between
convert(int,2493) and convert(int,4984))
go

alter table t_orders add constraint cnst_o_w_id check (o_w_id between
convert(int,2493) and convert(int,4984))
go

alter table t_order_line add constraint cnst_ol_w_id check (ol_w_id
between convert(int,2493) and convert(int,4984))
go

alter table t_new_order add constraint cnst_no_w_id check (no_w_id
between convert(int,2493) and convert(int,4984))
go

```

### ***add\_constraints\_t4.sql***

```

--file 7477_to_9968\add_constraints_t.sql

set ansi_warnings on
set ansi_nulls on
go

use tpcc
go

-- Drop any existing partitioning constraints
--

alter table t_warehouse drop constraint cnst_w_id
go

alter table t_district drop constraint cnst_d_w_id
go

alter table t_customer drop constraint cnst_c_w_id
go

alter table t_history drop constraint cnst_h_w_id
go

alter table t_stock drop constraint cnst_s_w_id
go

alter table t_orders drop constraint cnst_o_w_id
go

```

```

alter table t_order_line drop constraint cnst_ol_w_id
go

alter table t_new_order drop constraint cnst_no_w_id
go

-- Add partitioning constraints between 7477 and 9968
--

alter table t_warehouse add constraint cnst_w_id check (w_id >=
convert(int,7477))
go

alter table t_district add constraint cnst_d_w_id check (d_w_id >=
convert(int,7477))
go

alter table t_customer add constraint cnst_c_w_id check (c_w_id >=
convert(int,7477))
go

alter table t_history add constraint cnst_h_w_id check (h_w_id >=
convert(int,7477))
go

alter table t_stock add constraint cnst_s_w_id check (s_w_id >=
convert(int,7477))
go

alter table t_orders add constraint cnst_o_w_id check (o_w_id >=
convert(int,7477))
go

alter table t_order_line add constraint cnst_ol_w_id check (ol_w_id >=
convert(int,7477))
go

alter table t_new_order add constraint cnst_no_w_id check (no_w_id
>= convert(int,7477))
go

```

## ***add\_views1.sql***

```

-- file 1_to_2492\add_views.sql

set ansi_warnings on
set ansi_nulls on
go

use tpcc
go

-- drop any existing views
if object_id('warehouse') is not null drop view warehouse
if object_id('district') is not null drop view district
if object_id('customer') is not null drop view customer
if object_id('history') is not null drop view history
if object_id('stock') is not null drop view stock
if object_id('orders') is not null drop view orders
if object_id('order_line') is not null drop view order_line
if object_id('new_order') is not null drop view new_order
if object_id('item') is not null drop view item
go

--add views for partition 1
create view warehouse as

```

```

select * from /*rmt1.tpcc.dbo.*/t_warehouse
union all
select * from rmt2.tpcc.dbo.t_warehouse
union all
select * from rmt3.tpcc.dbo.t_warehouse
union all
select * from rmt4.tpcc.dbo.t_warehouse
go

create view district as
select * from /*rmt1.tpcc.dbo.*/t_district
union all
select * from rmt2.tpcc.dbo.t_district
union all
select * from rmt3.tpcc.dbo.t_district
union all
select * from rmt4.tpcc.dbo.t_district
go

create view customer as
select * from /*rmt1.tpcc.dbo.*/t_customer
union all
select * from rmt2.tpcc.dbo.t_customer
union all
select * from rmt3.tpcc.dbo.t_customer
union all
select * from rmt4.tpcc.dbo.t_customer
go

create view history as
select * from /*rmt1.tpcc.dbo.*/t_history
union all
select * from rmt2.tpcc.dbo.t_history
union all
select * from rmt3.tpcc.dbo.t_history
union all
select * from rmt4.tpcc.dbo.t_history
go

create view stock as
select * from /*rmt1.tpcc.dbo.*/t_stock
union all
select * from rmt2.tpcc.dbo.t_stock
union all
select * from rmt3.tpcc.dbo.t_stock
union all
select * from rmt4.tpcc.dbo.t_stock
go

create view orders as
select * from /*rmt1.tpcc.dbo.*/t_orders
union all
select * from rmt2.tpcc.dbo.t_orders
union all
select * from rmt3.tpcc.dbo.t_orders
union all
select * from rmt4.tpcc.dbo.t_orders
go

create view order_line as
select * from /*rmt1.tpcc.dbo.*/t_order_line
union all
select * from rmt2.tpcc.dbo.t_order_line
union all
select * from rmt3.tpcc.dbo.t_order_line
union all
select * from rmt4.tpcc.dbo.t_order_line
go

```



```

create view new_order as
select * from /*rmt1.tpc.c.dbo.*/t_new_order
union all
select * from rmt2.tpc.c.dbo.t_new_order
union all
select * from rmt3.tpc.c.dbo.t_new_order
union all
select * from rmt4.tpc.c.dbo.t_new_order
go

create view item as
select * from t_item
go

```

## **add\_views2.sql**

```

-- file 2493_to_4984\add_views.sql

set ansi_warnings on
set ansi_nulls on
go

use tpcc
go

-- drop any existing views
if object_id('warehouse') is not null drop view warehouse
if object_id('district') is not null drop view district
if object_id('customer') is not null drop view customer
if object_id('history') is not null drop view history
if object_id('stock') is not null drop view stock
if object_id('orders') is not null drop view orders
if object_id('order_line') is not null drop view order_line
if object_id('new_order') is not null drop view new_order
if object_id('item') is not null drop view item
go

--add views for partition 2
create view warehouse as
select * from /*rmt2.tpc.c.dbo.*/t_warehouse
union all
select * from rmt1.tpc.c.dbo.t_warehouse
union all
select * from rmt3.tpc.c.dbo.t_warehouse
union all
select * from rmt4.tpc.c.dbo.t_warehouse
go

create view district as
select * from /*rmt2.tpc.c.dbo.*/t_district
union all
select * from rmt1.tpc.c.dbo.t_district
union all
select * from rmt3.tpc.c.dbo.t_district
union all
select * from rmt4.tpc.c.dbo.t_district
go

create view customer as
select * from /*rmt2.tpc.c.dbo.*/t_customer
union all
select * from rmt1.tpc.c.dbo.t_customer
union all
select * from rmt3.tpc.c.dbo.t_customer
union all

```

```

select * from rmt4.tpc.c.dbo.t_customer
go

create view history as
select * from /*rmt2.tpc.c.dbo.*/t_history
union all
select * from rmt1.tpc.c.dbo.t_history
union all
select * from rmt3.tpc.c.dbo.t_history
union all
select * from rmt4.tpc.c.dbo.t_history
go

create view stock as
select * from /*rmt2.tpc.c.dbo.*/t_stock
union all
select * from rmt1.tpc.c.dbo.t_stock
union all
select * from rmt3.tpc.c.dbo.t_stock
union all
select * from rmt4.tpc.c.dbo.t_stock
go

create view orders as
select * from /*rmt2.tpc.c.dbo.*/t_orders
union all
select * from rmt1.tpc.c.dbo.t_orders
union all
select * from rmt3.tpc.c.dbo.t_orders
union all
select * from rmt4.tpc.c.dbo.t_orders
go

create view order_line as
select * from /*rmt2.tpc.c.dbo.*/t_order_line
union all
select * from rmt1.tpc.c.dbo.t_order_line
union all
select * from rmt3.tpc.c.dbo.t_order_line
union all
select * from rmt4.tpc.c.dbo.t_order_line
go

create view new_order as
select * from /*rmt2.tpc.c.dbo.*/t_new_order
union all
select * from rmt1.tpc.c.dbo.t_new_order
union all
select * from rmt3.tpc.c.dbo.t_new_order
union all
select * from rmt4.tpc.c.dbo.t_new_order
go

create view item as
select * from t_item
go

```

## **add\_views3.sql**

```

-- file 2493_to_4984\add_views.sql

set ansi_warnings on
set ansi_nulls on
go

```

```

use tpcc
go

-- drop any existing views
if object_id('warehouse') is not null drop view warehouse
if object_id('district') is not null drop view district
if object_id('customer') is not null drop view customer
if object_id('history') is not null drop view history
if object_id('stock') is not null drop view stock
if object_id('orders') is not null drop view orders
if object_id('order_line') is not null drop view order_line
if object_id('new_order') is not null drop view new_order
if object_id('item') is not null drop view item
go

--add views for partition 2
create view warehouse as
select * from /*rmt2.tpcc.dbo.*/t_warehouse
union all
select * from rmt1.tpcc.dbo.t_warehouse
union all
select * from rmt3.tpcc.dbo.t_warehouse
union all
select * from rmt4.tpcc.dbo.t_warehouse
go

create view district as
select * from /*rmt2.tpcc.dbo.*/t_district
union all
select * from rmt1.tpcc.dbo.t_district
union all
select * from rmt3.tpcc.dbo.t_district
union all
select * from rmt4.tpcc.dbo.t_district
go

create view customer as
select * from /*rmt2.tpcc.dbo.*/t_customer
union all
select * from rmt1.tpcc.dbo.t_customer
union all
select * from rmt3.tpcc.dbo.t_customer
union all
select * from rmt4.tpcc.dbo.t_customer
go

create view history as
select * from /*rmt2.tpcc.dbo.*/t_history
union all

```

```

select * from rmt1.tpcc.dbo.t_history
union all
select * from rmt3.tpcc.dbo.t_history
union all
select * from rmt4.tpcc.dbo.t_history
go

create view stock as
select * from /*rmt2.tpcc.dbo.*/t_stock
union all
select * from rmt1.tpcc.dbo.t_stock
union all
select * from rmt3.tpcc.dbo.t_stock
union all
select * from rmt4.tpcc.dbo.t_stock
go

create view orders as
select * from /*rmt2.tpcc.dbo.*/t_orders
union all
select * from rmt1.tpcc.dbo.t_orders
union all
select * from rmt3.tpcc.dbo.t_orders
union all
select * from rmt4.tpcc.dbo.t_orders
go

create view order_line as
select * from /*rmt2.tpcc.dbo.*/t_order_line
union all
select * from rmt1.tpcc.dbo.t_order_line
union all
select * from rmt3.tpcc.dbo.t_order_line
union all
select * from rmt4.tpcc.dbo.t_order_line
go

create view new_order as
select * from /*rmt2.tpcc.dbo.*/t_new_order
union all
select * from rmt1.tpcc.dbo.t_new_order
union all
select * from rmt3.tpcc.dbo.t_new_order
union all
select * from rmt4.tpcc.dbo.t_new_order
go

create view item as
select * from t_item

```

```
go
```

## ***add\_views4.sql***

```
-- file 7477_to_9968\add_views.sql

set ansi_warnings on
set ansi_nulls on
go

use tpcc
go

-- drop any existing views
if object_id('warehouse') is not null drop view warehouse
if object_id('district') is not null drop view district
if object_id('customer') is not null drop view customer
if object_id('history') is not null drop view history
if object_id('stock') is not null drop view stock
if object_id('orders') is not null drop view orders
if object_id('order_line') is not null drop view order_line
if object_id('new_order') is not null drop view new_order
if object_id('item') is not null drop view item
go

--add views for partition 4
create view warehouse as
select * from /*rmt4.tpc.dbo.*/t_warehouse
union all
select * from rmt1.tpc.dbo.t_warehouse
union all
select * from rmt2.tpc.dbo.t_warehouse
union all
select * from rmt3.tpc.dbo.t_warehouse
go

create view district as
select * from /*rmt4.tpc.dbo.*/t_district
union all
select * from rmt1.tpc.dbo.t_district
union all
select * from rmt2.tpc.dbo.t_district
union all
select * from rmt3.tpc.dbo.t_district
go

create view customer as
select * from /*rmt4.tpc.dbo.*/t_customer
union all
select * from rmt1.tpc.dbo.t_customer
union all
select * from rmt2.tpc.dbo.t_customer
union all
select * from rmt3.tpc.dbo.t_customer
go

create view history as
select * from /*rmt4.tpc.dbo.*/t_history
union all
select * from rmt1.tpc.dbo.t_history
union all
select * from rmt2.tpc.dbo.t_history
union all
select * from rmt3.tpc.dbo.t_history
go
```

```
create view stock as
select * from /*rmt4.tpc.dbo.*/t_stock
union all
select * from rmt1.tpc.dbo.t_stock
union all
select * from rmt2.tpc.dbo.t_stock
union all
select * from rmt3.tpc.dbo.t_stock
go
```

```
create view orders as
select * from /*rmt4.tpc.dbo.*/t_orders
union all
select * from rmt1.tpc.dbo.t_orders
union all
select * from rmt2.tpc.dbo.t_orders
union all
select * from rmt3.tpc.dbo.t_orders
go
```

```
create view order_line as
select * from /*rmt4.tpc.dbo.*/t_order_line
union all
select * from rmt1.tpc.dbo.t_order_line
union all
select * from rmt2.tpc.dbo.t_order_line
union all
select * from rmt3.tpc.dbo.t_order_line
go
```

```
create view new_order as
select * from /*rmt4.tpc.dbo.*/t_new_order
union all
select * from rmt1.tpc.dbo.t_new_order
union all
select * from rmt2.tpc.dbo.t_new_order
union all
select * from rmt3.tpc.dbo.t_new_order
go
```

```
create view item as
select * from t_item
go
```

## ***item\_iot1.sql***

```
-- file 1_to_2492\item_iot.sql

--add item instead-of-trigger for partition 1

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONS off
go

use tpcc
go

drop trigger iot_item
go

create trigger iot_item on item instead of update, insert, delete as
begin

--
-- local node (partition 1)
DELETE A FROM t_item A, DELETED D WHERE A.i_id = D.i_id
```

```

INSERT t_item SELECT * FROM INSERTED

--
-- remote node, partition 2
DELETE A FROM rmt2.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt2.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 3
DELETE A FROM rmt3.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt3.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 4
DELETE A FROM rmt4.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt4.tpcc.dbo.t_item SELECT * FROM INSERTED

end

```

### **item\_iot2.sql**

```

-- file 2493_to_4984\item_iot.sql

--add item instead-of-trigger for partition 2

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONSOFF
go

use tpcc
go

drop trigger iot_item
go

create trigger iot_item on item instead of update, insert, delete as
begin

--
-- remote node, partition 1
DELETE A FROM rmt1.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt1.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- local node (partition 2)
DELETE A FROM t_item A, DELETED D WHERE A.i_id = D.i_id
INSERT t_item SELECT * FROM INSERTED

--
-- remote node, partition 3
DELETE A FROM rmt3.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt3.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 4
DELETE A FROM rmt4.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt4.tpcc.dbo.t_item SELECT * FROM INSERTED

end

```

### **item\_iot3.sql**

```

-- file 2493_to_4984\item_iot.sql

--add item instead-of-trigger for partition 2

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONSOFF
go

use tpcc
go

drop trigger iot_item
go

create trigger iot_item on item instead of update, insert, delete as
begin

--
-- remote node, partition 1
DELETE A FROM rmt1.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt1.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- local node (partition 2)
DELETE A FROM t_item A, DELETED D WHERE A.i_id = D.i_id
INSERT t_item SELECT * FROM INSERTED

--
-- remote node, partition 3
DELETE A FROM rmt3.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt3.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 4
DELETE A FROM rmt4.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt4.tpcc.dbo.t_item SELECT * FROM INSERTED

end

```

### **item\_iot4.sql**

```

-- file 7477_to_9968\item_iot.sql

--add item instead-of-trigger for partition 4

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONSOFF
go

use tpcc
go

drop trigger iot_item
go

create trigger iot_item on item instead of update, insert, delete as
begin

--
-- remote node, partition 1

```

```

DELETE A FROM rmt1.tpc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt1.tpc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 2
DELETE A FROM rmt2.tpc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt2.tpc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 3
DELETE A FROM rmt3.tpc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt3.tpc.dbo.t_item SELECT * FROM INSERTED

--
-- local node (partition 4)
DELETE A FROM t_item A, DELETED D WHERE A.i_id = D.i_id
INSERT t_item SELECT * FROM INSERTED

end

```

## backup.sql

```

-- File: BACKUP.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.30
-- Copyright Microsoft, 2000
-- 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, blocksize = 4096

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. 4.30
-- Copyright Microsoft, 2000
-- Purpose: Creates tpcc database Backup Devices

```

```

use master
go

-- create backup devices

exec sp_addumpdevice 'disk','tpccback1','s:\tpccback1.dmp'
exec sp_addumpdevice 'disk','tpccback2','s:\tpccback2.dmp'
exec sp_addumpdevice 'disk','tpccback3','t:\tpccback3.dmp'
exec sp_addumpdevice 'disk','tpccback4','t:\tpccback4.dmp'

go

```

## createdb.sql

```

-- File: CREATEDB.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.30
-- Copyright Microsoft, 2000
-- Purpose: Creates tpcc database and backup files for 10 warehouses

```

```

use master
go

-- Create temporary table for timing

if exists ( select name from sysobjects where name = 'tpcc_timer' )
drop table tpcc_timer
go

create table tpcc_timer
(
    start_date char(30),
    end_date char(30)
)

insert into tpcc_timer values (0,0)
go

-- Store starting time

update tpcc_timer
set start_date = (select convert(char(30), getdate(),9))
go

-- create main database files

CREATE DATABASE tpcc
ON PRIMARY
(
    NAME = MSSQL80_tpcc_root,
    FILENAME = "C:\MSSQL80_tpcc_root.mdf",
    SIZE = 8MB,
    FILEGROWTH = 0),
FILEGROUP misc_fg
(
    NAME = MSSQL80_tpcc_misc_1,
    FILENAME = "H:",
    SIZE = 12000MB,
    FILEGROWTH = 0),
(
    NAME = MSSQL80_tpcc_misc_2,
    FILENAME = "J:",
    SIZE = 12000MB,
    FILEGROWTH = 0),
(
    NAME = MSSQL80_tpcc_misc_3,
    FILENAME = "L:",
    SIZE = 12000MB,
    FILEGROWTH = 0),
(
    NAME = MSSQL80_tpcc_misc_4,
    FILENAME = "N:",
    SIZE = 12000MB,
    FILEGROWTH = 0),
(
    NAME = MSSQL80_tpcc_misc_5,
    FILENAME = "P:",
    SIZE = 12000MB,
    FILEGROWTH = 0),
(
    NAME = MSSQL80_tpcc_misc_6,
    FILENAME = "R:",
    SIZE = 12000MB,
    FILEGROWTH = 0),
FILEGROUP big_fg
(
    NAME = MSSQL80_tpcc_big_1,

```

```

        FILENAME      = "G:",
        SIZE          = 23800MB,
        FILEGROWTH   = 0),
(
    NAME            = MSSQL80_tpcc_big_2,
    FILENAME       = "I:",
    SIZE           = 23800MB,
    FILEGROWTH    = 0),
(
    NAME            = MSSQL80_tpcc_big_3,
    FILENAME       = "K:",
    SIZE           = 23800MB,
    FILEGROWTH    = 0),
(
    NAME            = MSSQL80_tpcc_big_4,
    FILENAME       = "M:",
    SIZE           = 23800MB,
    FILEGROWTH    = 0),
(
    NAME            = MSSQL80_tpcc_big_5,
    FILENAME       = "O:",
    SIZE           = 23800MB,
    FILEGROWTH    = 0),
(
    NAME            = MSSQL80_tpcc_big_6,
    FILENAME       = "Q:",
    SIZE           = 23800MB,
    FILEGROWTH    = 0)
LOG ON
(
    NAME            = MSSQL80_tpcc_log,
    FILENAME       = "F:",
    SIZE           = 80000MB,
    FILEGROWTH    = 0)
go

-- Store ending time
update tpcc_timer
set end_date = (select convert(char(30), getdate(),9))
go

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' )
drop table tpcc_timer
go

```

## dbopt1.sql

```

-- File: DBOPT1.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.30
-- Copyright Microsoft, 2000
-- Purpose: Sets database options for data load

```

```

use master
go

```

```

exec sp_dboption tpcc,'select into/bulkcopy',true
exec sp_dboption tpcc,'trunc. log on chkpt.',true
exec sp_dboption tpcc,'torn page detection',false
go

```

```

use tpcc
go

```

```

checkpoint
go

```

## dbopt2.sql

```

-- File: DBOPT2_SHILOH.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.30
-- Copyright Microsoft, 2000
-- Purpose: Resets database options after data load

```

```

use master
go

```

```

sp_dboption tpcc,'select',false
go

```

```

sp_dboption tpcc,'trunc. ',false
go

```

```

use tpcc
go

```

```

checkpoint
go

```

```

sp_configure allow,1
go

```

```

reconfigure with override
go

```

```

/*                               */
/* Set option values for user-defined indexes */
/*                               */

```

```

sp_indexoption 't_customer', 'DisallowPageLocks', TRUE
go

```

```

sp_indexoption 't_district', 'DisallowPageLocks',TRUE
go

```

```

sp_indexoption 't_warehouse', 'DisallowPageLocks',
TRUE
go

```

```

sp_indexoption 't_stock', 'DisallowPageLocks',TRUE
go

```

```

sp_indexoption 't_order_line', 'DisallowRowLocks',
TRUE
go

```

```

sp_indexoption 't_orders', 'DisallowRowLocks',TRUE
go

```

```

sp_indexoption 't_new_order', 'DisallowRowLocks',
TRUE
go

```

```

sp_indexoption 't_item', 'DisallowRowLocks',TRUE
go

```

```

sp_indexoption 't_item', 'DisallowPageLocks',TRUE
go

```

```

Print ''
Print '*****'
Print 'Pre-specified Locking Hierarchy:'
Print ' Lockflag = 0 ==> No pre-specified hierarchy'
Print ' Lockflag = 1 ==> Lock at Page-level then Table-level'
Print ' Lockflag = 2 ==> Lock at Row-level then Table-level'
Print ' Lockflag = 3 ==> Lock at Table-level'
Print ''

```

```

select name,lockflags
from sysindexes
where object_id("t_warehouse") = id or

```

```

        object_id("t_district")          = id or
        object_id("t_customer")          = id or
        object_id("t_stock")             = id or
        object_id("t_orders")            = id or
        object_id("t_order_line")        = id or
        object_id("t_history")            = id or
        object_id("t_new_order")         = id or
        object_id("t_item")               = id
order   by lockflags asc
go

sp_configure allow,0
go

reconfigure with override
go

exec sp_dboption tpcc,          'auto update statistics',
FALSE
exec sp_dboption tpcc,          'auto create statistics', FALSE
go

exec sp_tableoption  "t_district",      "pintable",true
exec sp_tableoption  "t_warehouse",    "pintable",true
exec sp_tableoption  "t_new_order",    "pintable",true
exec sp_tableoption  "t_item",         "pintable",true
go

```

### ***idxcuscl.sql***

```

-- File:  IDXCUSCL.SQL
--       Microsoft TPC-C Benchmark Kit Ver. 4.30
--       Copyright Microsoft, 2000
-- Purpose:  Creates clustered index on customer table

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONS off
set QUOTED_IDENTIFIER off
go

use tpcc
go

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

if exists (select o.name from sysconstraints c, sysobjects o
where c.id = object_id('t_customer') and c.constid = o.id and o.name =
'customer_c1PK' )
    alter table t_customer drop constraint customer_c1PK

alter table t_customer add constraint customer_c1PK primary key
clustered (c_w_id, c_d_id, c_id)
on big_fg

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

go

```

### ***idxcusnc.sql***

```

-- File:  IDXCUSNC.SQL
--       Microsoft TPC-C Benchmark Kit Ver. 4.30
--       Copyright Microsoft, 2000
-- Purpose:  Creates non-clustered index on customer table

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONS off
set QUOTED_IDENTIFIER off
go

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 = 'customer_nc1' )
    drop index t_customer.customer_nc1

create unique nonclustered index customer_nc1 on t_customer(c_w_id,
c_d_id, c_last, c_first, c_id)
on big_fg

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

go

```

### ***idxdiscl.sql***

```

-- File:  IDXDISCL.SQL
--       Microsoft TPC-C Benchmark Kit Ver. 4.20
--       Copyright Microsoft, 2000
-- Purpose:  Creates clustered index on district table

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONS off
set QUOTED_IDENTIFIER off
go

use tpcc
go

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

if exists (select o.name from sysconstraints c, sysobjects o
where c.id = object_id('t_district') and c.constid = o.id and o.name =
'district_c1PK' )
    alter table t_district drop constraint district_c1PK

alter table t_district add constraint district_c1PK primary key clustered
(d_w_id, d_id)
with fillfactor=100
on misc_fg

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

```

```
go
```

## ***idxitmcl.sql***

```
-- File:   IDXITMCL.SQL
--        Microsoft TPC-C Benchmark Kit Ver. 4.30
--        Copyright Microsoft, 2000
-- Purpose: Creates clustered index on item table

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONS off
set QUOTED_IDENTIFIER off
go

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 = 'item_c1' )
    drop index t_item.item_c1

create unique clustered index item_c1 on t_item(i_id)
    on misc_fg

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

go
```

## ***idxnodcl.sql***

```
-- File:   IDXNODCL.SQL
--        Microsoft TPC-C Benchmark Kit Ver. 4.30
--        Copyright Microsoft, 2000
-- Purpose: Creates clustered index on new_order table

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONS off
set QUOTED_IDENTIFIER off
go

use tpcc
go

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

if exists (select o.name from sysconstraints c, sysobjects o
where c.id = object_id('t_new_order') and c.constid = o.id and o.name
= 't_new_order_c1PK' )
    alter table t_new_order drop constraint new_order_c1PK

alter table t_new_order add constraint new_order_c1PK primary key
clustered (no_w_id, no_d_id, no_o_id)
    on misc_fg

select @enddate = getdate()
```

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

```
go
```

## ***idxodlcl.sql***

```
-- File:   IDXODLCL.SQL
--        Microsoft TPC-C Benchmark Kit Ver. 4.30
--        Copyright Microsoft, 2000
-- Purpose: Creates clustered index on order_line table

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONS off
set QUOTED_IDENTIFIER off
go

use tpcc
go

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

if exists (select o.name from sysconstraints c, sysobjects o
where c.id = object_id('t_order_line') and c.constid = o.id and o.name
= 'order_line_c1PK' )
    alter table t_order_line drop constraint order_line_c1PK

alter table t_order_line add constraint order_line_c1PK primary key
clustered (ol_w_id, ol_d_id, ol_o_id, ol_number)
    on misc_fg

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

```
go
```

## ***idxordcl.sql***

```
-- File:   IDXORDCL.SQL
--        Microsoft TPC-C Benchmark Kit Ver. 4.30
--        Copyright Microsoft, 2000
-- Purpose: Creates clustered index on orders table

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONS off
set QUOTED_IDENTIFIER off
go

use tpcc
go

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

if exists (select o.name from sysconstraints c, sysobjects o
where c.id = object_id('t_orders') and c.constid = o.id and o.name =
'orders_c1PK' )
    alter table t_orders drop constraint orders_c1PK
```



```
alter table t_orders add constraint orders_c1PK primary key clustered
(o_w_id, o_d_id, o_id)
on 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. 4.30
--       Copyright Microsoft, 2000
-- Purpose: Creates non-clustered index on orders table
```

```
set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONSoFF
set QUOTED_IDENTIFIER off
go
```

```
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 t_orders.orders_nc1
```

```
create index orders_nc1 on t_orders(o_w_id, o_d_id, o_c_id, o_id)
on 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. 4.30
--       Copyright Microsoft, 2000
-- Purpose: Creates clustered index on stock table
```

```
set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONSoFF
set QUOTED_IDENTIFIER off
go
```

```
use tpcc
go
```

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

```
if exists (select o.name from sysconstraints c, sysobjects o
where c.id = object_id('t_stock') and c.constid = o.id and o.name =
'stock_c1PK' )
```

```
alter table t_stock drop constraint stock_c1PK
```

```
alter table t_stock add constraint stock_c1PK primary key clustered
(s_i_id, s_w_id)
on big_fg
```

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

```
go
```

## ***idxwarcl.sql***

```
-- File:   IDXWARCL.SQL
--       Microsoft TPC-C Benchmark Kit Ver. 4.30
--       Copyright Microsoft, 2000
-- Purpose: Creates clustered index on warehouse table
```

```
set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONSoFF
set QUOTED_IDENTIFIER off
go
```

```
use tpcc
go
```

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

```
if exists (select o.name from sysconstraints c, sysobjects o
where c.id = object_id('t_warehouse') and c.constid = o.id and o.name =
'warehouse_c1PK' )
alter table t_warehouse drop constraint warehouse_c1PK
```

```
alter table t_warehouse add constraint warehouse_c1PK primary key
clustered (w_id)
with fillfactor=100
on misc_fg
```

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

```
go
```

## ***tables.sql***

```
-- File:   TABLES.SQL
--       Microsoft TPC-C Benchmark Kit Ver. 4.30
--       Copyright Microsoft, 2000
-- Purpose: Creates TPC-C tables
```

```
set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONSoFF
set QUOTED_IDENTIFIER off
```

```

SET ANSI_NULL_DFLT_ONoff

go

use tpcc
go

--
-- Remove all existing TPC-C tables, views, etc.
--

if objectproperty(object_id('warehouse'), 'IsView') is not null
    drop view warehouse
go
if objectproperty(object_id('district'), 'IsView') is not null
    drop view district
go
if objectproperty(object_id('customer'), 'IsView') is not null
    drop view customer
go
if objectproperty(object_id('history'), 'IsView') is not null
    drop view history
go
if objectproperty(object_id('stock'), 'IsView') is not null
    drop view stock
go
if objectproperty(object_id('orders'), 'IsView') is not null
    drop view orders
go
if objectproperty(object_id('order_line'), 'IsView') is not null
    drop view order_line
go
if objectproperty(object_id('new_order'), 'IsView') is not null
    drop view new_order
go
if objectproperty(object_id('item'), 'IsView') is not null
    drop view item
go

if objectproperty(object_id('t_warehouse'), 'IsTable') is not null
    drop table t_warehouse

```

```

go
if objectproperty(object_id('t_district'), 'IsTable') is not null
    drop table t_district
go
if objectproperty(object_id('t_customer'), 'IsTable') is not null
    drop table t_customer
go
if objectproperty(object_id('t_history'), 'IsTable') is not null
    drop table t_history
go
if objectproperty(object_id('t_new_order'), 'IsTable') is not null
    drop table t_new_order
go
if objectproperty(object_id('t_orders'), 'IsTable') is not null
    drop table t_orders
go
if objectproperty(object_id('t_order_line'), 'IsTable') is not null
    drop table t_order_line
go
if objectproperty(object_id('t_item'), 'IsTable') is not null
    drop table t_item
go
if objectproperty(object_id('t_stock'), 'IsTable') is not null
    drop table t_stock
go

--
-- Create new tables
--

create table t_warehouse
(
    w_id                int,
    w_name              char(10),
    w_street_1         char(20),
    w_street_2         char(20),
    w_city             char(20),
    w_state            char(2),
    w_zip             char(9),
    w_tax             numeric(4,4),
    w_ytd             numeric(12,2)
) on misc_fg
go

```

```

create table t_district
(
    d_id                tinyint,
    d_w_id              int,
    d_name              char(10),
    d_street_1         char(20),
    d_street_2         char(20),
    d_city              char(20),
    d_state             char(2),
    d_zip              char(9),
    d_tax               numeric(4,4),
    d_ytd               numeric(12,2),
    d_next_o_id        int
) on misc_fg
go

create table t_customer
(
    c_id                int,
    c_d_id              tinyint,
    c_w_id              int,
    c_first             char(16),
    c_middle            char(2),
    c_last              char(16),
    c_street_1         char(20),
    c_street_2         char(20),
    c_city              char(20),
    c_state             char(2),
    c_zip              char(9),
    c_phone            char(16),
    c_since             datetime,
    c_credit            char(2),
    c_credit_lim        numeric(12,2),
    c_discount          numeric(4,4),
    c_balance           numeric(12,2),
    c_ytd_payment       numeric(12,2),
    c_payment_cnt       smallint,
    c_delivery_cnt      smallint,
    c_data              char(500)
) on big_fg
go

create table t_history
(
    h_c_id              int,
    h_c_d_id            tinyint,
    h_c_w_id            int,
    h_d_id              tinyint,
    h_w_id              int,
    h_date              datetime,
    h_amount            numeric(6,2),
    h_data              char(24),
    constraint history_c1PK primary key clustered (h_w_id,
h_d_id, h_c_id, h_date)
) on misc_fg
go

create table t_new_order
(
    no_o_id             int,
    no_d_id             tinyint,
    no_w_id             int
) on misc_fg
go

create table t_orders
(
    o_id                int,
    o_d_id              tinyint,
    o_w_id              int,
    o_c_id              int,
    o_entry_d           datetime,
    o_carrier_id        tinyint,
    o_ol_cnt            tinyint,
    o_all_local         tinyint
) on misc_fg
go

create table t_order_line
(
    ol_o_id             int,
    ol_d_id             tinyint,
    ol_w_id             int,
    ol_number           tinyint,
    ol_i_id             int,
    ol_supply_w_id      int,
    ol_delivery_d        datetime,
    ol_quantity         smallint,
    ol_amount           numeric(6,2),
    ol_dist_info        char(24)
) on misc_fg

```

```

go

create table t_item
(
    i_id                int,
    i_im_id             int,
    i_name              char(24),
    i_price             numeric(5,2),
    i_data              char(50)
) on misc_fg
go

```

```

create table t_stock
(
    s_i_id              int,
    s_w_id              int,
    s_quantity          smallint,
    s_dist_01           char(24),
    s_dist_02           char(24),
    s_dist_03           char(24),
    s_dist_04           char(24),
    s_dist_05           char(24),
    s_dist_06           char(24),
    s_dist_07           char(24),
    s_dist_08           char(24),
    s_dist_09           char(24),
    s_dist_10           char(24),
    s_ytd               int,
    s_order_cnt         smallint,
    s_remote_cnt        smallint,
    s_data              char(50)
) on big_fg
go

```

### **restore.sql**

```

-- File:  RESTORE.SQL
--      Microsoft TPC-C Benchmark Kit Ver. 4.30
--      Copyright Microsoft, 2000
-- Purpose: Loads database backup from backup files

```

```

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

load database tpcc from tpccback1, tpccback2, tpccback3, tpccback4
with stats = 1, replace

```

```

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

```

### **removedb.sql**

```

-- File:  REMOVEDB.SQL
--      Microsoft TPC-C Benchmark Kit Ver. 4.30
--      Copyright Microsoft, 2000
-- Purpose: Removes tpcc database and backup files

```

```

use master
go

-- remove any existing database and backup files

exec sp_dbremove tpcc, dropdev
go

exec sp_dropdevice 'tpccback1'
exec sp_dropdevice 'tpccback2'
exec sp_dropdevice 'tpccback3'
exec sp_dropdevice 'tpccback4'
go

```

### **version.sql**

```

-- File:  VERSION.SQL
--      Microsoft TPC-C Benchmark Kit Ver. 4.10
--      Copyright Microsoft, 1999
-- 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.

```

```

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONS off
go

use tpcc
go

if exists ( select name from sysobjects where name = 'tpcc_version' )
    drop procedure tpcc_version
go

create proc tpcc_version
as
declare @version char(8)

```

```

begin
    select @version = '4.10.000'
    select @version as 'Version'
end

```

```
go
```

## Load Source Code

### getargs.c

```

//      File:          GETARGS.C
//                        Microsoft TPC-C Kit Ver.
4.30
//                        Copyright Microsoft,
1996, 1997, 1998, 1999, 2000
//      Purpose:    Source file for command line processing

// Includes
#include "tpcc.h"

//=====
//
// Function name: GetArgsLoader
//
//=====

void GetArgsLoader(int argc, char **argv, TPCCLDR_ARGS *pargs)
{
    int          i;
    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;
    pargs->password        = 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       = DEFPLDPACKSIZE;
    pargs->starting_warehouse =
DEF_STARTING_WAREHOUSE;
    pargs->build_index     =
BUILD_INDEX;
    pargs->index_order     =
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 through */
            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;

            case 'U':
                pargs->user = ptr+2;
                break;

            case 'b':
                pargs->batch =
                atol(ptr+2);
                break;

            case 'W':
                pargs->num_warehouses
                = atol(ptr+2);
                break;

            case 's':
                pargs->starting_warehouse = atol(ptr+2);
                break;

            case 't':
                {
                    if
                    (strcmp(ptr+2,"item") == 0)
                        pargs->table_item = TRUE;
                    else if
                    (strcmp(ptr+2,"warehouse") == 0)

```

```

pargs->table_warehouse = TRUE;
else if
//=====
//
// Function name: GetArgsLoaderUsage
//
//=====
pargs->table_customer = TRUE;
else if
//=====
//=====
pargs->table_orders = TRUE;
else
void GetArgsLoaderUsage()
{
printf("\nUnrecognized command");
#ifdef DEBUG
printf("[%d]DBG: Entering GetArgsLoaderUsage()\n", (int)
GetCurrentThreadId());
#endif
exit(1);
}
break;
printf("TPCCLDR:\n\n");
printf("Parameter Default\n");
case 'f':
pargs->loader_res_file = printf("-----\n");
ptr+2;
break;
printf("-W Number of Warehouses to Load Required
\n");
printf("-S Server %s\n", SERVER);
case 'p':
printf("-U Username %s\n", USER);
pargs->pack_size = printf("-P Password %s\n",
atol(ptr+2);
PASSWORD);
break;
printf("-D Database %s\n", DATABASE);
case 'i':
printf("-b Batch Size %d\n",
atol(ptr+2);
(long) BATCH);
printf("-p TDS packet size %d\n",
(long) DEFLDPACKSIZE);
break;
printf("-f Loader Results Output Filename
%s\n", LOADER_RES_FILE);
case 'o':
printf("-s Starting Warehouse %d\n",
atol(ptr+2);
(long) DEF_STARTING_WAREHOUSE);
printf("-i Build Option (data = 0, data and index = 1)
%ld\n", (long) BUILD_INDEX);
break;
printf("-o Cluster Index Build Order (before = 1, after = 0)
%ld\n", (long) INDEX_ORDER);
case 'c':
pargs->scale_down = printf("-c Build Scaled Database (normal = 0, tiny = 1)
atol(ptr+2);
%ld\n", (long) SCALE_DOWN);
break;
printf("-d Index Script Path %s\n",
INDEX_SCRIPT_PATH);
case 'd':
printf("-t Table to Load all tables
= ptr+2;
pargs->index_script_path
\n");
break;
printf(" [item|warehouse|customer|orders]\n");
printf(" Notes: \n");
printf(" - the '-t' parameter may be included multiple times to \n");
default:
printf(" specify multiple tables to be loaded \n");
GetArgsLoaderUsage();
exit(-1);
printf(" - 'item' loads ITEM table \n");
break;
printf(" - 'warehouse' loads WAREHOUSE, DISTRICT, and
STOCK tables \n");
}
printf(" - 'customer' loads CUSTOMER and HISTORY tables
\n");
printf(" - 'orders' load NEW-ORDER, ORDERS, ORDER-LINE
tables \n");
/* check for required args */
if (pargs->num_warehouses == UNDEF)
{
printf("Number of Warehouses is required\n");
printf("\nNote: Command line switches are case
exit(-2);
sensitive.\n");
}
return;
}
exit(0);
}

```

## random.c

```
// File: RANDOM.C
// Microsoft TPC-C Kit Ver.
4.30
// Copyright Microsoft,
1996, 1997, 1998, 1999, 2000
// Purpose: Random number generation routines for
database loader

// Includes
#include "tpcc.h"
#include "math.h"

// Defines
#define A 16807
#define M 2147483647
#define Q 127773 /* M div A */
#define R 2836 /* M mod A */
#define Thread __declspec(thread)

// Globals
long Thread Seed = 0; /* thread local seed */

/*****
*****
*
* random -
* Implements a GOOD pseudo random number generator. This
generator *
* will/should? run the complete period before repeating. *
*
* Copied from:
* Random Numbers Generators: Good Ones Are Hard to Find.
*
* Communications of the ACM - October 1988 Volume 31
Number 10 *
*
* Machine Dependencies:
* long must be 2 ^ 31 - 1 or greater.
*
*****
*****/

/*****
*****
* seed - load the Seed value used in irand and drand. Should be used
before *
* first call to irand or drand.
*****
*****/

void seed(long val)
{

#ifdef DEBUG
printf("[%ld]DBG: Entering seed(...\n", (int)
GetCurrentThreadId());
printf("Old Seed %ld New Seed %ld\n",Seed, val);
#endif

if ( val < 0 )
val = abs(val);
```

```
Seed = val;
}

/*****
*****
*
* irand - returns a 32 bit integer pseudo random number with a period
of *
* 1 to 2 ^ 32 - 1.
*
* parameters:
* none.
*
* returns:
* 32 bit integer - defined as long ( see above ).
*
* side effects:
* seed get recomputed.
*****
*****/

long irand()
{
register long s; /* copy of seed */
register long test; /* test flag */
register long hi; /* tmp value for speed */
register long lo; /* tmp value for speed */

#ifdef DEBUG
printf("[%ld]DBG: Entering irand(...\n", (int)
GetCurrentThreadId());
#endif

s = Seed;
hi = s / Q;
lo = s % Q;

test = A * lo - R * hi;
if ( test > 0 )
Seed = test;
else
Seed = test + M;

return( Seed );
}

/*****
*****
*
* drand - returns a double pseudo random number between 0.0 and 1.0.
*
* See irand.
*****
*****/

double drand()
{

#ifdef DEBUG
printf("[%ld]DBG: Entering drand(...\n", (int)
GetCurrentThreadId());
#endif

return( (double)irand() / 2147483647.0);
}
```

```

=====
// Function : RandomNumber
//
// Description:
=====
long RandomNumber(long lower, long upper)
{
    long rand_num;

#ifdef DEBUG
    printf("[%ld]DBG: Entering RandomNumber()...\n", (int)
GetCurrentThreadId());
#endif

        if ( upper == lower ) /* pgd 08-13-96 perf enhancement */
            return lower;

        upper++;

    if ( upper <= lower )
        rand_num = upper;
    else
        rand_num = lower + irand() % (upper - lower);
/* pgd 08-13-96 perf enhancement */

#ifdef DEBUG
    printf("[%ld]DBG: RandomNumber between %ld & %ld ==>
%ld\n",
                                (int)
GetCurrentThreadId(), lower, upper, rand_num);
#endif

    return rand_num;
}

#if 0
//Orginal code pgd 08/13/96

long RandomNumber(long lower,
                                long upper)
{
    long rand_num;

#ifdef DEBUG
    printf("[%ld]DBG: Entering RandomNumber()...\n", (int)
GetCurrentThreadId());
#endif

        upper++;

    if ((upper <= lower))
        rand_num = upper;
    else
        rand_num = lower + irand() % ((upper > lower)
? upper - lower : upper);

#ifdef DEBUG
    printf("[%ld]DBG: RandomNumber between %ld & %ld ==>
%ld\n",

```

```

                                (int)
GetCurrentThreadId(), lower, upper, rand_num);
#endif

        return rand_num;
    }
#endif

=====
// Function : NURand
//
// Description:
=====
long NURand(int iConst,
            long x,
            long y,
            long C)
{
    long rand_num;

#ifdef DEBUG
    printf("[%ld]DBG: Entering NURand()...\n", (int)
GetCurrentThreadId());
#endif

    rand_num = (((RandomNumber(0,iConst) | RandomNumber(x,y)) +
C) % (y-x+1))+x;

#ifdef DEBUG
    printf("[%ld]DBG: NURand: num = %d\n", (int)
GetCurrentThreadId(), rand_num);
#endif

    return rand_num;
}

```

### strings.c

```

//      File:          STRINGS.C
//                               Microsoft TPC-C Kit Ver.
4,30
//                               Copyright Microsoft,
1996, 1997, 1998, 1999, 2000
//      Purpose:     Source file for database loader string functions

// Includes
#include "tpcc.h"
#include <string.h>
#include <ctype.h>

=====
//
// Function name: MakeAddress
//
=====

void MakeAddress(char *street_1,
                char *street_2,
                char *city,
                char *state,

```



```

char *zip)
{
#ifdef DEBUG
printf("[%ld]DBG: Entering MakeAddress()\n", (int)
GetCurrentThreadId());
#endif

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 >= 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 y,
//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)
{
int len;
int i;
char cc = 'a';
static char chArray[] =
"0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz";
static int chArrayMax = 61;

#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[z] = 0;

return len;
}

```

```

}

//=====
//=====
//
// Function name: MakeOriginalAlphaString
//
//=====
//=====

int MakeOriginalAlphaString(int x,
int y,
int z,
char *str,
int
percent)
{
    int len;
    int val;
    int start;

#ifdef DEBUG
    printf("[%ld]DBG: Entering MakeOriginalAlphaString()\n", (int)
GetCurrentThreadId());
#endif

    // verify percentage is valid
    if ((percent < 0) || (percent > 100))
    {
        printf("MakeOriginalAlphaString: Invalid
percentage: %d\n", percent);
        exit(-1);
    }

    // verify string is at least 8 chars in length
    if ((x + y) <= 8)
    {
        printf("MakeOriginalAlphaString: string length
must be >= 8\n");
        exit(-1);
    }

    // Make Alpha String
    len = MakeAlphaString(x,y, z, str);

    val = RandomNumber(1,100);
    if (val <= percent)
    {
        start = RandomNumber(0, len - 8);
        strncpy(str + start, "ORIGINAL", 8);
    }

#ifdef DEBUG
    printf("[%ld]DBG: MakeOriginalAlphaString: : %s\n",
(int) GetCurrentThreadId(), str);
#endif

    return strlen(str);
}

//=====
//=====

```

```

//
// Function name: MakeNumberString
//
//=====
//=====
int MakeNumberString(int x, int y, int z, char *str)
{
    char tmp[16];

    //MakeNumberString is always called
    MakeZipNumberString(16, 16, 16, string)

    memset(str, '0', 16);
    itoa(RandomNumber(0, 99999999), tmp, 10);
    memcpy(str, tmp, strlen(tmp));

    itoa(RandomNumber(0, 99999999), tmp, 10);
    memcpy(str+8, tmp, strlen(tmp));

    str[16] = 0;

    return 16;
}

//=====
//=====
//
// Function name: MakeZipNumberString
//
//=====
//=====
int MakeZipNumberString(int x, int y, int z, char *str)
{
    char tmp[16];

    //MakeZipNumberString is always called
    MakeZipNumberString(9, 9, 9, string)

    strcpy(str, "000011111");

    itoa(RandomNumber(0, 9999), tmp, 10);
    memcpy(str, tmp, strlen(tmp));

    return 9;
}

//=====
//=====
//
// Function name: InitString
//
//=====
//=====
void InitString(char *str, int len)
{
#ifdef DEBUG
    printf("[%ld]DBG: Entering InitString()\n", (int)
GetCurrentThreadId());
#endif

    memset(str, ' ', len);
    str[len] = 0;
}

```

```

//=====
// Function name: InitAddress
//
// Description:
//
//=====
=====

void InitAddress(char *street_1, char *street_2, char *city, char *state,
char *zip)
{
    memset(street_1, '\0', ADDRESS_LEN+1);
    memset(street_2, '\0', ADDRESS_LEN+1);
    memset(city, '\0', ADDRESS_LEN+1);

    street_1[ADDRESS_LEN+1] = 0;
    street_2[ADDRESS_LEN+1] = 0;
    city[ADDRESS_LEN+1] = 0;

    memset(state, '\0', STATE_LEN+1);
    state[STATE_LEN+1] = 0;

    memset(zip, '\0', ZIP_LEN+1);
    zip[ZIP_LEN+1] = 0;
}

```

```

//=====
//
// Function name: PaddString
//
//=====
=====

void PaddString(int max, char *name)
{
    int len;

    len = strlen(name);
    if ( len < max )
        memset(name+len, '\0', max - len);
    name[max] = 0;

    return;
}

```

### **time.c**

```

// File: TIME.C
// Microsoft TPC-C Kit Ver.
4.30
// Copyright Microsoft,
1996, 1997, 1998, 1999,2000
// Purpose: Source file for time functions

// Includes
#include "tpcc.h"

// Globals
static long start_sec;

```

```

//=====
//
// Function name: TimeNow
//
//=====
=====

```

```

long TimeNow()
{
    long time_now;
    struct _timeb el_time;

#ifdef DEBUG
    printf("[%ld]DBG: Entering TimeNow()\n", (int)
GetCurrentThreadId());
#endif

    _ftime(&el_time);

    time_now = ((el_time.time - start_sec) * 1000) + el_time.millitm;

    return time_now;
}

```

### **tpcc.h**

```

// File: TPCC.H
// Microsoft TPC-C Kit Ver.
4.30
// Copyright Microsoft,
1996, 1997, 1998, 1999, 2000
// Purpose: Header file for TPC-C database loader

```

```

// Build number of TPC Benchmark Kit
#define TPCKIT_VER "4.30"

```

```

// General headers
#include <windows.h>
#include <winbase.h>
#include <stdlib.h>
#include <stdio.h>
#include <process.h>
#include <stddef.h>
#include <stdarg.h>
#include <string.h>
#include <time.h>
#include <sys\timeb.h>
#include <sys\types.h>

```

```

// ODBC headers
#include <sql.h>
#include <sqlext.h>
#include <odbcss.h>

```

```

// General constants
#define MILLI 1000
#define FALSE 0
#define TRUE 1
#define UNDEF -1
#define MINPRINTASCII 32
#define MAXPRINTASCII 126

```

```

// Default environment constants
#define SERVER ""

```

```

#define DATABASE "tpcc"
#define USER "sa"
#define PASSWORD ""

// Default loader arguments
#define BATCH 10000
#define DEF_LDPACKSIZE 32768
#define LOADER_RES_FILE "logs\\load.out"
#define LOADER_NURAND_C 123
#define DEF_STARTING_WAREHOUSE 1
#define BUILD_INDEX 1 // build both data and indexes
#define INDEX_ORDER 1 // build indexes before load
#define SCALE_DOWN 0 // build a normal scale database
#define INDEX_SCRIPT_PATH "scripts"

typedef struct
{
    char *server;
    char *database;
    char *user;
    char *password;
    BOOL tables_all; // set if loading all tables
    BOOL table_item; // set if loading ITEM table specifically
    BOOL // set if loading WAREHOUSE, DISTRICT, and STOCK
    table_warehouse;
    BOOL table_customer; // set if loading CUSTOMER and HISTORY
    BOOL table_orders; // set if loading NEW-ORDER, ORDERS, ORDER-LINE
    long num_warehouses;
    long batch;
    long verbose;
    long pack_size;
    char *loader_res_file;
    char *synch_servername;
    long case_sensitivity;
    long starting_warehouse;
    long build_index;
    long index_order;
    long scale_down;
    char *index_script_path;
} TPCC_LDR_ARGS;

// String length constants
#define SERVER_NAME_LEN 20
#define DATABASE_NAME_LEN 20
#define USER_NAME_LEN 20
#define PASSWORD_LEN 20
#define TABLE_NAME_LEN 20
#define I_DATA_LEN 50
#define I_NAME_LEN 24
#define BRAND_LEN 1
#define LAST_NAME_LEN 16
#define W_NAME_LEN 10
#define ADDRESS_LEN 20
#define STATE_LEN 2

#define ZIP_LEN 9
#define S_DIST_LEN 24
#define S_DATA_LEN 50
#define D_NAME_LEN 10
#define FIRST_NAME_LEN 16
#define MIDDLE_NAME_LEN 2
#define PHONE_LEN 16
#define CREDIT_LEN 2
#define C_DATA_LEN 500
#define H_DATA_LEN 24
#define DIST_INFO_LEN 24
#define MAX_OL_NEW_ORDER_ITEMS 15
#define MAX_OL_ORDER_STATUS_ITEMS 15
#define STATUS_LEN 25
#define OL_DIST_INFO_LEN 24
#define C_SINCE_LEN 23
#define H_DATE_LEN 23
#define OL_DELIVERY_D_LEN 23
#define O_ENTRY_D_LEN 23

// Functions in random.c
void seed();
long irand();
double drand();
void WUCreate();
short WURand();
long RandomNumber(long lower, long upper);

// Functions in getargs.c
void GetArgsLoader();
void GetArgsLoaderUsage();

// Functions in time.c
long TimeNow();

// Functions in strings.c
void MakeAddress();
void LastName();
int MakeAlphaString();
int MakeOriginalAlphaString();
int MakeNumberString();
int MakeZipNumberString();
void InitString();
void InitAddress();
void PaddString();

tpccldr.c
// File: TPCC_LDR.C
// Microsoft TPC-C Kit Ver.
4.30
// Copyright Microsoft,
1996, 1997, 1998, 1999, 2000
// Purpose: Source file for TPC-C database loader

// Includes
#include "tpcc.h"
#include "search.h"

// Defines
#define MAXITEMS 100000
#define MAXITEMS_SCALE_DOWN 100
#define CUSTOMERS_PER_DISTRICT 3000

```

```

#define CUSTOMERS_SCALE_DOWN 30
#define DISTRICT_PER_WAREHOUSE 10
#define ORDERS_PER_DISTRICT 3000
#define ORDERS_SCALE_DOWN 30
#define MAX_CUSTOMER_THREADS 2
#define MAX_ORDER_THREADS 3
#define MAX_MAIN_THREADS 4

// Functions declarations

void HandleErrorDBC (SQLHDBC hdbc1);

void CheckSQL();
void CheckDataBase();

long NURand();
void LoadItem();
void LoadWarehouse();

void Stock();
void District();

void LoadCustomer();
void CustomerBufInit();
void CustomerBufLoad();
void LoadCustomerTable();
void LoadHistoryTable();

void LoadOrders();
void OrdersBufInit();
void OrdersBufLoad();
void LoadOrdersTable();
void LoadNewOrderTable();
void LoadOrderLineTable();
void GetPermutation();
void CheckForCommit();
void OpenConnections();
void BuildIndex();
void FormatDate ();

// Shared memory structures

typedef struct
{
    long ol;
    long ol_i_id;
    long ol_supply_w_id;
    short ol_quantity;
    double ol_amount;
    char ol_dist_info[DIST_INFO_LEN+1];
    char ol_delivery_d[OL_DELIVERY_D_LEN+1];
} ORDER_LINE_STRUCT;

typedef struct
{
    long o_id;
    short o_d_id;
    long o_w_id;
    long o_c_id;
    short o_carrier_id;
    short o_ol_cnt;
    short o_all_local;
    ORDER_LINE_STRUCT o_ol[15];
} ORDERS_STRUCT;

typedef struct
{
    long c_id;
    short c_d_id;
    long c_w_id;
    char c_first[FIRST_NAME_LEN+1];
    char c_middle[MIDDLE_NAME_LEN+1];
    char c_last[LAST_NAME_LEN+1];
    char c_street_1[ADDRESS_LEN+1];
    char c_street_2[ADDRESS_LEN+1];
    char c_city[ADDRESS_LEN+1];
    char c_state[STATE_LEN+1];
    char c_zip[ZIP_LEN+1];
    char c_phone[PHONE_LEN+1];
    double c_credit[CREDIT_LEN+1];
    double c_credit_lim;
    double c_discount;
    // fix to avoid ODBC float to numeric conversion problem.
    // double c_balance;
    char c_balance[6];
    double c_ytd_payment;
    short c_payment_cnt;
    short c_delivery_cnt;
    char c_data[C_DATA_LEN+1];
    double h_amount;
    char h_data[H_DATA_LEN+1];
} CUSTOMER_STRUCT;

typedef struct
{
    char c_last[LAST_NAME_LEN+1];
    char c_first[FIRST_NAME_LEN+1];
    long c_id;
} CUSTOMER_SORT_STRUCT;

typedef struct
{
    long time_start;
} LOADER_TIME_STRUCT;

// Global variables

char szLastError[300];

HENV henv;

HDBC v_hdbc;
// for SQL Server version verification
HDBC i_hdbc1; //
for ITEM table
HDBC w_hdbc1; //
for WAREHOUSE, DISTRICT, STOCK
HDBC c_hdbc1; //
for CUSTOMER

```

```

HDBC    c_hdbc2; //
for HISTORY
HDBC    o_hdbc1; //
for ORDERS
HDBC    o_hdbc2; //
for NEW-ORDER
HDBC    o_hdbc3; //
for ORDER-LINE

HSTMT   v_hstmt; //
for SQL Server version verification
HSTMT   i_hstmt1;
HSTMT   w_hstmt1;
HSTMT   c_hstmt1, c_hstmt2;
HSTMT   o_hstmt1, o_hstmt2, o_hstmt3;

ORDERS_STRUCT  orders_buf[ORDERS_PER_DISTRICT];
CUSTOMER_STRUCT
customer_buf[CUSTOMERS_PER_DISTRICT];
long           orders_rows_loaded;
long           new_order_rows_loaded;
long           order_line_rows_loaded;
long           history_rows_loaded;
long           customer_rows_loaded;
long           stock_rows_loaded;
long           district_rows_loaded;
long           item_rows_loaded;
long           warehouse_rows_loaded;
long           main_time_start;
long           main_time_end;
long           max_items;
long           customers_per_district;
long           orders_per_district;
long           first_new_order;
long           last_new_order;

TPCCLDR_ARGS  *aptr, args;

// support for log directory determined by environment
#define LOGBASE_MAXLEN
128
int           UseAltLogDir = 0;
char          LogBase[LOGBASE_MAXLEN];
char          LogFile[2*LOGBASE_MAXLEN];
#define LOGFILE_CREATE(name)
\
\           strcpy(LogFile, LogBase);
\
\           strcat(LogFile, name);

//=====
//
// Function name: main
//
//=====

int main(int argc, char **argv)
{
    DWORD      dwThreadID[MAX_MAIN_THREADS];
    HANDLE     hThread[MAX_MAIN_THREADS];
    FILE       *fLoader;
    char       buffer[255];
    int        i;

    for (i=0; i<MAX_MAIN_THREADS; i++)
        hThread[i] = NULL;

    printf("\n*****\n");
    printf("\n*                               *");
    printf("\n* Microsoft SQL Server          *");
    printf("\n*                               *");
    printf("\n* TPC-C BENCHMARK KIT: Database loader\n");
    printf("\n*                               *");
    printf("\n* Version %s                     *",
    TPCKIT_VER);
    printf("\n*                               *");

    printf("\n*****\n\n");

    // process command line arguments

    aptr = &args;
    GetArgsLoader(argc, argv, aptr);

    // process environment variables
    if ( GetEnvironmentVariable("LOGBASE", LogBase,
    LOGBASE_MAXLEN))
    {
        UseAltLogDir = 1;
        strcat(LogBase, "\\logs");
        printf("Will use %s for log files\n", LogBase);
    }
    else
        strcpy(LogBase, "logs");

    // verify database and tables exist before attempting to load
    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.\n");
            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.\n");
            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,

0,

&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;
    long          i_im_id;
    char          i_name[I_NAME_LEN+1];
    double        i_price;
    char          i_data[I_DATA_LEN+1];
    char          name[20];
    long          time_start;
    RETCODE       rc;
    DBINT         rcint;
    char          bcphint[128];

    // 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("idxitm1");

    InitString(i_name, I_NAME_LEN+1);
    InitString(i_data, I_DATA_LEN+1);

    sprintf(name, "%s.%s", aptr->database, "t_item");

    LOGFILE_CREATE("\\item.err")
    rc = bcp_init(i_hdbc1, name, NULL, LogFile, DB_IN);

```

```

        if (rc != SUCCEED)
            HandleErrorDBC(i_hdbc1);

        if ((aptr->build_index == 1) && (aptr->index_order == 1))
        {
            sprintf(bcphint, "tablock, order (i_id,
ROWS_PER_BATCH = 100000, CHECK_CONSTRAINTS");
            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);
        if (rc != SUCCEED)
            HandleErrorDBC(i_hdbc1);

        rc = bcp_bind(i_hdbc1, (BYTE *) &i_im_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 2);
        if (rc != SUCCEED)
            HandleErrorDBC(i_hdbc1);

        rc = bcp_bind(i_hdbc1, (BYTE *) i_name, 0,
I_NAME_LEN, NULL, 0, 0, 3);
        if (rc != SUCCEED)
            HandleErrorDBC(i_hdbc1);

        rc = bcp_bind(i_hdbc1, (BYTE *) &i_price, 0,
SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 4);
        if (rc != SUCCEED)
            HandleErrorDBC(i_hdbc1);

        rc = bcp_bind(i_hdbc1, (BYTE *) i_data, 0, I_DATA_LEN,
NULL, 0, 0, 5);
        if (rc != SUCCEED)
            HandleErrorDBC(i_hdbc1);

        time_start = (TimeNow() / MILLI);

        item_rows_loaded = 0;

        for (i_id = 1; i_id <= max_items; i_id++)
        {
            i_im_id = RandomNumber(1L, 10000L);

            MakeAlphaString(14, 24, I_NAME_LEN,
i_name);

            i_price = ((float) RandomNumber(100L,
10000L))/100.0;

            MakeOriginalAlphaString(26, 50,
I_DATA_LEN, i_data, 10);

            rc = bcp_sendrow(i_hdbc1);
            if (rc != SUCCEED)
                HandleErrorDBC(i_hdbc1);

            item_rows_loaded++;
            CheckForCommit(i_hdbc1, i_hstmt1,
item_rows_loaded, "item", &time_start);
        }

        rcint = bcp_done(i_hdbc1);
        if (rcint < 0)
            HandleErrorDBC(i_hdbc1);

```



```

printf("Finished loading item table.\n");

SQLFreeStmt(i_hstmt1, SQL_DROP);
SQLDisconnect(i_hdbc1);
SQLFreeConnect(i_hdbc1);

// if build index after load
if ((aptr->build_index == 1) && (aptr->index_order == 0))
    BuildIndex("idxitmc1");
}

//=====
//
// Function : LoadWarehouse
//
// Loads WAREHOUSE table and loads Stock and District as
// Warehouses are created
//
//=====

void LoadWarehouse()
{
    long    w_id;
    char    w_name[W_NAME_LEN+1];
    char    w_street_1[ADDRESS_LEN+1];
    char    w_street_2[ADDRESS_LEN+1];
    char    w_city[ADDRESS_LEN+1];
    char    w_state[STATE_LEN+1];
    char    w_zip[ZIP_LEN+1];
    double  w_tax;
    double  w_ytd;
    char    name[20];
    long    time_start;
    RETCODE rc;
    DBINT   rcint;
    char    bcp hint[128];

    // Seed with unique number
    seed(aptr->starting_warehouse + 1);

    printf("Loading warehouse table...\n");

    // if build index before load...
    if ((aptr->build_index == 1) && (aptr->index_order == 1))
        BuildIndex("idxwarcl");

    InitString(w_name, W_NAME_LEN+1);
    InitAddress(w_street_1, w_street_2, w_city, w_state,
w_zip);

    sprintf(name, "%s.%s", aptr->database, "t_warehouse");

    LOGFILE_CREATE("\\whouse.err")
    rc = bcp_init(w_hdbc1, name, NULL, LogFile, DB_IN);

    if (rc != SUCCEED)
        HandleErrorDBC(w_hdbc1);

    if ((aptr->build_index == 1) && (aptr->index_order == 1))
    {

```

```

        sprintf(bcp hint, "tablock, order (w_id),
ROWS_PER_BATCH = %d, CHECK_CONSTRAINTS",
aptr->num_warehouses);
        rc = bcp_control(w_hdbc1, BCPHINTS, (void*)
bcp hint);
        if (rc != SUCCEED)
            HandleErrorDBC(w_hdbc1);
    }

    rc = bcp_bind(w_hdbc1, (BYTE *) &w_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 1);
    if (rc != SUCCEED)
        HandleErrorDBC(w_hdbc1);

    rc = bcp_bind(w_hdbc1, (BYTE *) w_name, 0,
W_NAME_LEN, NULL, 0, 0, 2);
    if (rc != SUCCEED)
        HandleErrorDBC(w_hdbc1);

    rc = bcp_bind(w_hdbc1, (BYTE *) w_street_1, 0,
ADDRESS_LEN, NULL, 0, 0, 3);
    if (rc != SUCCEED)
        HandleErrorDBC(w_hdbc1);

    rc = bcp_bind(w_hdbc1, (BYTE *) w_street_2, 0,
ADDRESS_LEN, NULL, 0, 0, 4);
    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);

    rc = bcp_bind(w_hdbc1, (BYTE *) w_state, 0,
STATE_LEN, NULL, 0, 0, 6);
    if (rc != SUCCEED)
        HandleErrorDBC(w_hdbc1);

    rc = bcp_bind(w_hdbc1, (BYTE *) w_zip, 0, ZIP_LEN,
NULL, 0, 0, 7);
    if (rc != SUCCEED)
        HandleErrorDBC(w_hdbc1);

    rc = bcp_bind(w_hdbc1, (BYTE *) &w_tax, 0,
SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 8);
    if (rc != SUCCEED)
        HandleErrorDBC(w_hdbc1);

    rc = bcp_bind(w_hdbc1, (BYTE *) &w_ytd, 0,
SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 9);
    if (rc != SUCCEED)
        HandleErrorDBC(w_hdbc1);

    time_start = (TimeNow() / MILLI);

    warehouse_rows_loaded = 0;

    for (w_id = (long)aptr->starting_warehouse; w_id <=
aptr->num_warehouses; w_id++)
    {
        MakeAlphaString(6,10, W_NAME_LEN,
w_name);

        MakeAddress(w_street_1, w_street_2, w_city,
w_state, w_zip);

```

```

        w_tax = ((float)
RandomNumber(0L,2000L))/10000.00;

        w_ytd = 300000.00;

        rc = bcp_sendrow(w_hdbc1);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        warehouse_rows_loaded++;
        CheckForCommit(w_hdbc1, i_hstmt1,
warehouse_rows_loaded, "warehouse", &time_start);
    }

    rcint = bcp_done(w_hdbc1);
    if (rcint < 0)
        HandleErrorDBC(w_hdbc1);

    printf("Finished loading warehouse table.\n");

    // if build index after load...
    if ((aptr->build_index == 1) && (aptr->index_order == 0))
        BuildIndex("idxwarc1");

    stock_rows_loaded = 0;
    district_rows_loaded = 0;

    District();
    Stock();
}

//=====
//
// Function : District
//
//=====

void District()
{
    short  d_id;
    long   d_w_id;
    char   d_name[D_NAME_LEN+1];
    char   d_street_1[ADDRESS_LEN+1];
    char   d_street_2[ADDRESS_LEN+1];
    char   d_city[ADDRESS_LEN+1];
    char   d_state[STATE_LEN+1];
    char   d_zip[ZIP_LEN+1];
    double d_tax;
    double d_ytd;
    char   name[20];
    long   d_next_o_id;
    long   time_start;
    int     w_id;
    RETCODE rc;
    DBINT  rcint;
    char   bcphint[128];

    // Seed with unique number
    seed(aptr->starting_warehouse + 2);

    printf("Loading district table...\n");

    // build index before load
    if ((aptr->build_index == 1) && (aptr->index_order == 1))

```

```

        BuildIndex("idxdisc1");

        InitString(d_name, D_NAME_LEN+1);
        InitAddress(d_street_1, d_street_2, d_city, d_state, d_zip);

        sprintf(name, "%s..%s", aptr->database, "t_district");

        LOGFILE_CREATE("\\district.err")
        rc = bcp_init(w_hdbc1, name, NULL, LogFile, DB_IN);

        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        if ((aptr->build_index == 1) && (aptr->index_order == 1))
        {
            sprintf(bcphint, "tablock, order (d_w_id, d_id),
ROWS_PER_BATCH = %u, CHECK_CONSTRAINTS",
(aptr->num_warehouses * 10));
            rc = bcp_control(w_hdbc1, BCPHINTS, (void*)
bcphint);
            if (rc != SUCCEEDED)
                HandleErrorDBC(w_hdbc1);
        }

        rc = bcp_bind(w_hdbc1, (BYTE *) &d_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 1);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) &d_w_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 2);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) d_name, 0,
D_NAME_LEN, NULL, 0, 0, 3);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) d_street_1, 0,
ADDRESS_LEN, NULL, 0, 0, 4);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) d_street_2, 0,
ADDRESS_LEN, NULL, 0, 0, 5);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) d_city, 0,
ADDRESS_LEN, NULL, 0, 0, 6);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) d_state, 0,
STATE_LEN, NULL, 0, 0, 7);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) d_zip, 0, ZIP_LEN,
NULL, 0, 0, 8);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) &d_tax, 0,
SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 9);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

```

```

        rc = bcp_bind(w_hdbc1, (BYTE *) &d_ytd, 0,
SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 10);
        if (rc != SUCCEED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) &d_next_o_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 11);
        if (rc != SUCCEED)
            HandleErrorDBC(w_hdbc1);

        d_ytd = 30000.0;

        d_next_o_id = orders_per_district+1;

        time_start = (TimeNow() / MILLI);

        for (w_id = apr->starting_warehouse; w_id <=
apr->num_warehouses; w_id++)
        {
            d_w_id = w_id;

            for (d_id = 1; d_id <=
DISTRICT_PER_WAREHOUSE; d_id++)
            {
                MakeAlphaString(6,10,D_NAME_LEN, d_name);

                MakeAddress(d_street_1, d_street_2,
d_city, d_state, d_zip);

                d_tax = ((float)
RandomNumber(0L,2000L))/10000.0;

                rc = bcp_sendrow(w_hdbc1);
                if (rc != SUCCEED)

HandleErrorDBC(w_hdbc1);

                district_rows_loaded++;
                CheckForCommit(w_hdbc1,
w_hstmt1, district_rows_loaded, "district", &time_start);
            }
        }

        rcint = bcp_done(w_hdbc1);
        if (rcint < 0)
            HandleErrorDBC(w_hdbc1);

        printf("Finished loading district table.\n");

        // if build index after load...
        if ((apr->build_index == 1) && (apr->index_order == 0))
            BuildIndex("idxdiscl");

    return;
}

=====
//
// Function : Stock
//
=====
void Stock()

```

```

{
    long    s_i_id;
    long    s_w_id;
    short   s_quantity;
    char    s_dist_01[S_DIST_LEN+1];
    char    s_dist_02[S_DIST_LEN+1];
    char    s_dist_03[S_DIST_LEN+1];
    char    s_dist_04[S_DIST_LEN+1];
    char    s_dist_05[S_DIST_LEN+1];
    char    s_dist_06[S_DIST_LEN+1];
    char    s_dist_07[S_DIST_LEN+1];
    char    s_dist_08[S_DIST_LEN+1];
    char    s_dist_09[S_DIST_LEN+1];
    char    s_dist_10[S_DIST_LEN+1];
    long    s_ytd;
    short   s_order_cnt;
    short   s_remote_cnt;
    char    s_data[S_DATA_LEN+1];
    short   len;
    char    name[20];
    long    time_start;
    RETCODE rc;
    DBINT   rcint;
    char    bcphint[128];

    // Seed with unique number
    seed(apr->starting_warehouse + 3);

    // if build index before load...
    if ((apr->build_index == 1) && (apr->index_order == 1))
        BuildIndex("idxstkcl");

    sprintf(name, "%s..%s", apr->database, "t_stock");

    LOGFILE_CREATE("\\stock.err")
    rc = bcp_init(w_hdbc1, name, NULL, LogFile, DB_IN);
    if (rc != SUCCEED)
        HandleErrorDBC(w_hdbc1);

    if ((apr->build_index == 1) && (apr->index_order == 1))
    {
        sprintf(bcphint, "tablock, order (s_i_id, s_w_id),
ROWS_PER_BATCH = %u, CHECK_CONSTRAINTS",
(apr->num_warehouses * 100000));
        rc = bcp_control(w_hdbc1, BCPHINTS, (void*)
bcphint);
        if (rc != SUCCEED)
            HandleErrorDBC(w_hdbc1);
    }

    rc = bcp_bind(w_hdbc1, (BYTE *) &s_i_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 1);
    if (rc != SUCCEED)
        HandleErrorDBC(w_hdbc1);

    bcp_bind(w_hdbc1, (BYTE *) &s_w_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 2);
    if (rc != SUCCEED)
        HandleErrorDBC(w_hdbc1);

    rc = bcp_bind(w_hdbc1, (BYTE *) &s_quantity, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 3);
    if (rc != SUCCEED)
        HandleErrorDBC(w_hdbc1);

    rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_01, 0,
S_DIST_LEN, NULL, 0, 0, 4);
    if (rc != SUCCEED)

```

```

        HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_02, 0,
S_DIST_LEN, NULL, 0, 0, 5);
        if (rc != SUCCEED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_03, 0,
S_DIST_LEN, NULL, 0, 0, 6);
        if (rc != SUCCEED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_04, 0,
S_DIST_LEN, NULL, 0, 0, 7);
        if (rc != SUCCEED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_05, 0,
S_DIST_LEN, NULL, 0, 0, 8);
        if (rc != SUCCEED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_06, 0,
S_DIST_LEN, NULL, 0, 0, 9);
        if (rc != SUCCEED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_07, 0,
S_DIST_LEN, NULL, 0, 0, 10);
        if (rc != SUCCEED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_08, 0,
S_DIST_LEN, NULL, 0, 0, 11);
        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)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_10, 0,
S_DIST_LEN, NULL, 0, 0, 13);
        if (rc != SUCCEED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) &s_ytd, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 14);
        if (rc != SUCCEED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) &s_order_cnt, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 15);
        if (rc != SUCCEED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) &s_remote_cnt, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 16);
        if (rc != SUCCEED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) s_data, 0,
S_DATA_LEN, NULL, 0, 0, 17);
        if (rc != SUCCEED)
            HandleErrorDBC(w_hdbc1);

        s_ytd = s_order_cnt = s_remote_cnt = 0;

```

```

        time_start = (TimeNow() / MILLI);

        printf("...Loading stock table\n");

        for (s_i_id=1; s_i_id <= max_items; s_i_id++)
        {
                for (s_w_id = (long)aptr->starting_warehouse;
s_w_id <= aptr->num_warehouses; s_w_id++)
                {
                        s_quantity =
(short)RandomNumber(10L,100L);
                        len =
MakeAlphaString(24,24,S_DIST_LEN, s_dist_01);
                        len =
MakeAlphaString(24,24,S_DIST_LEN, s_dist_02);
                        len =
MakeAlphaString(24,24,S_DIST_LEN, 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;
    long w_id;

    short d_id;
    DWORD
dwThreadID[MAX_CUSTOMER_THREADS];
    HANDLE
hThread[MAX_CUSTOMER_THREADS];
    char name[20];
    RETCODE
rc;
    DBINT
rcint;
    char
bcphint[128];
    char
cmd[256];
    // SQLRETURN
rc_1;
    // SQLSMALLINT
recnum, MsgLen;
    // SQLCHAR
SqlState[6], Msg[SQL_MAX_MESSAGE_LENGTH];
    // SQLINTEGER
NativeError;

    // Seed with unique number
seed(aptr->starting_warehouse + 4);

    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, "t_customer");

    LOGFILE_CREATE("\\customer.err")
rc = bcp_init(c_hdbc1, name, NULL, LogFile, DB_IN);
if (rc != SUCCEEDED)
    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, CHECK_CONSTRAINTS",
(aptr->num_warehouses * 30000));
rc = bcp_control(c_hdbc1, BCPHINTS, (void*)
bcphint);
        if (rc != SUCCEEDED)
            HandleErrorDBC(c_hdbc1);
    }

    sprintf(name, "%s.%s", aptr->database, "t_history");

    LOGFILE_CREATE("\\history.err")
rc = bcp_init(c_hdbc2, name, NULL, LogFile, DB_IN);

```

```

if (rc != SUCCEEDED)
    HandleErrorDBC(c_hdbc2);

    sprintf(bcphint, "tablock, order (h_w_id, h_d_id, h_c_id,
h_date), ROWS_PER_BATCH = %u, CHECK_CONSTRAINTS",
(aptr->num_warehouses * 30000));
rc = bcp_control(c_hdbc2, BCPHINTS, (void*) bcphint);
if (rc != SUCCEEDED)
    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 = (long)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 = %d, w_id = %d\n", d_id, w_id);

            hThread[0] = CreateThread(NULL,
0,
(LPTHREAD_START_ROUTINE)
LoadCustomerTable,
&customer_time_start,
0,
&dwThreadID[0]);

            if (hThread[0] == NULL)
            {
                printf("Error, failed in
creating creating thread = 0.\n");
                exit(-1);
            }

            // Start History table thread

            printf("...Loading history table for:
d_id = %d, w_id = %d\n", d_id, w_id);

            hThread[1] = CreateThread(NULL,
0,
(LPTHREAD_START_ROUTINE)
LoadHistoryTable,
&history_time_start,

```

```

0,
&dwThreadID[1]);

    if (hThread[1] == NULL)
    {
        printf("Error, failed in
creating creating thread = 1.\n");
        exit(-1);
    }

    WaitForSingleObject( hThread[0],
INFINITE );
    WaitForSingleObject( hThread[1],
INFINITE );

    if (CloseHandle(hThread[0]) ==
FALSE)
    {
        printf("Error, failed in
closing customer thread handle with errno: %d\n", GetLastError());
    }

    if (CloseHandle(hThread[1]) ==
FALSE)
    {
        printf("Error, failed in
closing history thread handle with errno: %d\n", GetLastError());
    }
}

// flush the bulk connection
rcint = bcp_done(c_hdbc1);
if (rcint < 0)
    HandleErrorDBC(c_hdbc1);

rcint = bcp_done(c_hdbc2);
if (rcint < 0)
    HandleErrorDBC(c_hdbc2);

printf("Finished loading customer table.\n");

// if build index after load...
if ((aptr->build_index == 1) && (aptr->index_order == 0))
    BuildIndex("idxcuscl");

// build non-clustered index
if (aptr->build_index == 1)
    BuildIndex("idxcusnc");

// Output the NURAND used for the loader into C_FIRST
for C_ID = 1,
    // C_W_ID = 1, and C_D_ID = 1
    sprintf(cmd, "isql -S%s -U%s -P%s -d%s -e -Q\"update
t_customer set c_first = 'C_LOAD = %d' where c_id = 1 and c_w_id =
1 and c_d_id = 1\" > %s\\nurand_load.log",
        aptr->server,
        aptr->user,
        aptr->password,
        aptr->database,
        LOADER_NURAND_C,
        LogBase);

```

```

system(cmd);

SQLFreeStmt(c_hstmt1, SQL_DROP);
SQLDisconnect(c_hdbc1);
SQLFreeConnect(c_hdbc1);

SQLFreeStmt(c_hstmt2, SQL_DROP);
SQLDisconnect(c_hdbc2);
SQLFreeConnect(c_hdbc2);

return;
}

//=====
//
// Function : CustomerBufInit
//
//=====
void CustomerBufInit()
{
    int i;

    for (i=0;i<customers_per_district;i++)
    {
        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,"");
        strcpy(customer_buf[i].c_street_2,"");
        strcpy(customer_buf[i].c_city,"");
        strcpy(customer_buf[i].c_state,"");
        strcpy(customer_buf[i].c_zip,"");
        strcpy(customer_buf[i].c_phone,"");
        strcpy(customer_buf[i].c_credit,"");

        customer_buf[i].c_credit_lim = 0;
        customer_buf[i].c_discount = (float) 0;

        // fix to avoid ODBC float to numeric
conversion problem.
        // customer_buf[i].c_balance = 0;
        strcpy(customer_buf[i].c_balance,"");

        customer_buf[i].c_ytd_payment = 0;
        customer_buf[i].c_payment_cnt = 0;
        customer_buf[i].c_delivery_cnt = 0;

        strcpy(customer_buf[i].c_data,"");

        customer_buf[i].h_amount = 0;

        strcpy(customer_buf[i].h_data,"");
    }
}

```

```

=====
//
// Function : CustomerBufLoad
//
// Fills shared buffer for HISTORY and CUSTOMER
=====
void CustomerBufLoad(int d_id, int w_id)
{
    long i;
    CUSTOMER_SORT_STRUCT
c[CUSTOMERS_PER_DISTRICT];

    for (i=0;i<customers_per_district;i++)
    {
        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);

        c[i].c_id = i+1;
    }

    printf("...Loading customer buffer for: d_id = %d, w_id =
%d\n",
        d_id, w_id);

    for (i=0;i<customers_per_district;i++)
    {
        customer_buf[i].c_d_id = d_id;
        customer_buf[i].c_w_id = w_id;
        customer_buf[i].h_amount = 10.0;

        customer_buf[i].c_ytd_payment = 10.0;

        customer_buf[i].c_payment_cnt = 1;
        customer_buf[i].c_delivery_cnt = 0;

        // Generate CUSTOMER and HISTORY data

        customer_buf[i].c_id = c[i].c_id;

        strcpy(customer_buf[i].c_first, c[i].c_first);
        strcpy(customer_buf[i].c_last, c[i].c_last);

        customer_buf[i].c_middle[0] = 'O';
        customer_buf[i].c_middle[1] = 'E';

        MakeAddress(customer_buf[i].c_street_1,
customer_buf[i].c_street_2,
customer_buf[i].c_city,

customer_buf[i].c_state,
customer_buf[i].c_zip);

        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)
RandomNumber(0L, 5000L)) / 10000.0;

            // fix to avoid ODBC float to numeric
conversion problem.
            // customer_buf[i].c_balance = -10.0;
            strcpy(customer_buf[i].c_balance, "-10.0");

            MakeAlphaString(300, 500, C_DATA_LEN,
customer_buf[i].c_data);

            // Generate HISTORY data
            MakeAlphaString(12, 24, H_DATA_LEN,
customer_buf[i].h_data);
    }
}

=====
//
// Function : LoadCustomerTable
//
=====
void LoadCustomerTable(LOADER_TIME_STRUCT
*customer_time_start)
{
    int i;
    long c_id;
    short c_d_id;
    long c_w_id;
    char c_first[FIRST_NAME_LEN+1];
    char c_middle[MIDDLE_NAME_LEN+1];
    char c_last[LAST_NAME_LEN+1];
    char c_street_1[ADDRESS_LEN+1];
    char c_street_2[ADDRESS_LEN+1];
    char c_city[ADDRESS_LEN+1];
    char c_state[STATE_LEN+1];
    char c_zip[ZIP_LEN+1];
    char c_phone[PHONE_LEN+1];
    char c_credit[CREDIT_LEN+1];
    double c_credit_lim;
    double c_discount;

    // fix to avoid ODBC float to numeric conversion problem.

    // double c_balance;
    char c_balance[6];

    double c_ytd_payment;
    short c_payment_cnt;
    short c_delivery_cnt;
    char c_data[C_DATA_LEN+1];
    char c_since[C_SINCE_LEN+1];
    RETCODE rc;

    rc = bcp_bind(c_hdbc1, (BYTE *) &c_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 1);

```

```

        if (rc != SUCCEEDED)
            HandleErrorDBC(c_hdbc1);

        rc = bcp_bind(c_hdbc1, (BYTE *) &c_d_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 2);
        if (rc != SUCCEEDED)
            HandleErrorDBC(c_hdbc1);

        rc = bcp_bind(c_hdbc1, (BYTE *) &c_w_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 3);
        if (rc != SUCCEEDED)
            HandleErrorDBC(c_hdbc1);

        rc = bcp_bind(c_hdbc1, (BYTE *) c_first, 0, FIRST_NAME_LEN,
NULL, 0, 0, 4);
        if (rc != SUCCEEDED)
            HandleErrorDBC(c_hdbc1);

        rc = bcp_bind(c_hdbc1, (BYTE *) c_middle, 0,
MIDDLE_NAME_LEN, NULL, 0, 0, 5);
        if (rc != SUCCEEDED)
            HandleErrorDBC(c_hdbc1);

        rc = bcp_bind(c_hdbc1, (BYTE *) c_last, 0, LAST_NAME_LEN,
NULL, 0, 0, 6);
        if (rc != SUCCEEDED)
            HandleErrorDBC(c_hdbc1);

        rc = bcp_bind(c_hdbc1, (BYTE *) c_street_1, 0, ADDRESS_LEN,
NULL, 0, 0, 7);
        if (rc != SUCCEEDED)
            HandleErrorDBC(c_hdbc1);

        rc = bcp_bind(c_hdbc1, (BYTE *) c_street_2, 0, ADDRESS_LEN,
NULL, 0, 0, 8);
        if (rc != SUCCEEDED)
            HandleErrorDBC(c_hdbc1);

        rc = bcp_bind(c_hdbc1, (BYTE *) c_city, 0, ADDRESS_LEN,
NULL, 0, 0, 9);
        if (rc != SUCCEEDED)
            HandleErrorDBC(c_hdbc1);

        rc = bcp_bind(c_hdbc1, (BYTE *) c_state, 0, STATE_LEN, NULL,
0, 0, 10);
        if (rc != SUCCEEDED)
            HandleErrorDBC(c_hdbc1);

        rc = bcp_bind(c_hdbc1, (BYTE *) c_zip, 0, ZIP_LEN, NULL, 0, 0,
11);
        if (rc != SUCCEEDED)
            HandleErrorDBC(c_hdbc1);

        rc = bcp_bind(c_hdbc1, (BYTE *) c_phone, 0, PHONE_LEN,
NULL, 0, 0, 12);
        if (rc != SUCCEEDED)
            HandleErrorDBC(c_hdbc1);

        rc = bcp_bind(c_hdbc1, (BYTE *) &c_since, 0,
C_SINCE_LEN, NULL, 0, SQLCHARACTER, 13);
        if (rc != SUCCEEDED)
            HandleErrorDBC(c_hdbc1);

        rc = bcp_bind(c_hdbc1, (BYTE *) c_credit, 0, CREDIT_LEN,
NULL, 0, 0, 14);
        if (rc != SUCCEEDED)
            HandleErrorDBC(c_hdbc1);

        rc = bcp_bind(c_hdbc1, (BYTE *) &c_credit_lim, 0,
SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 15);
        if (rc != SUCCEEDED)
            HandleErrorDBC(c_hdbc1);

        rc = bcp_bind(c_hdbc1, (BYTE *) &c_discount, 0,
SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 16);
        if (rc != SUCCEEDED)
            HandleErrorDBC(c_hdbc1);

        // fix to avoid ODBC float to numeric conversion problem.

        // rc = bcp_bind(c_hdbc1, (BYTE *) &c_balance, 0,
SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 17);
        // if (rc != SUCCEEDED)
        //     HandleErrorDBC(c_hdbc1);
        rc = bcp_bind(c_hdbc1, (BYTE *) c_balance, 0, 5, NULL, 0,
SQLCHARACTER, 17);
        if (rc != SUCCEEDED)
            HandleErrorDBC(c_hdbc1);

        rc = bcp_bind(c_hdbc1, (BYTE *) &c_ytd_payment, 0,
SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 18);
        if (rc != SUCCEEDED)
            HandleErrorDBC(c_hdbc1);

        rc = bcp_bind(c_hdbc1, (BYTE *) &c_payment_cnt, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 19);
        if (rc != SUCCEEDED)
            HandleErrorDBC(c_hdbc1);

        rc = bcp_bind(c_hdbc1, (BYTE *) &c_delivery_cnt, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 20);
        if (rc != SUCCEEDED)
            HandleErrorDBC(c_hdbc1);

        rc = bcp_bind(c_hdbc1, (BYTE *) c_data, 0, 500, NULL, 0, 0, 21);
        if (rc != SUCCEEDED)
            HandleErrorDBC(c_hdbc1);

        for (i = 0; i < customers_per_district; i++)
        {
            c_id = customer_buf[i].c_id;
            c_d_id = customer_buf[i].c_d_id;
            c_w_id = customer_buf[i].c_w_id;

            strcpy(c_first, customer_buf[i].c_first);
            strcpy(c_middle, customer_buf[i].c_middle);
            strcpy(c_last, customer_buf[i].c_last);
            strcpy(c_street_1, customer_buf[i].c_street_1);
            strcpy(c_street_2, customer_buf[i].c_street_2);
            strcpy(c_city, customer_buf[i].c_city);
            strcpy(c_state, customer_buf[i].c_state);
            strcpy(c_zip, customer_buf[i].c_zip);
            strcpy(c_phone, customer_buf[i].c_phone);
            strcpy(c_credit, customer_buf[i].c_credit);

            FormatDate(&c_since);

            c_credit_lim = customer_buf[i].c_credit_lim;
            c_discount = customer_buf[i].c_discount;

            // fix to avoid ODBC float to numeric
conversion problem.
            // c_balance = customer_buf[i].c_balance;

```



```

strcpy(c_balance, customer_buf[i].c_balance);

c_ytd_payment =
customer_buf[i].c_ytd_payment;
c_payment_cnt =
customer_buf[i].c_payment_cnt;
c_delivery_cnt =
customer_buf[i].c_delivery_cnt;

strcpy(c_data, customer_buf[i].c_data);

// Send data to server
rc = bcp_sendrow(c_hdbc1);
if (rc != SUCCEEDED)
    HandleErrorDBC(c_hdbc1);

customer_rows_loaded++;
CheckForCommit(c_hdbc1, c_hstmt1,
customer_rows_loaded, "customer",
&customer_time_start->time_start);
}

}

//=====
//
// Function : LoadHistoryTable
//
//=====

void LoadHistoryTable(LOADER_TIME_STRUCT
*history_time_start)
{
    int i;
    long c_id;
    short c_d_id;
    long c_w_id;
    double h_amount;
    char h_data[H_DATA_LEN+1];
    char h_date[H_DATE_LEN+1];
    RETCODE rc;

    rc = bcp_bind(c_hdbc2, (BYTE *) &c_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 1);
    if (rc != SUCCEEDED)
        HandleErrorDBC(c_hdbc2);

    rc = bcp_bind(c_hdbc2, (BYTE *) &c_d_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 2);
    if (rc != SUCCEEDED)
        HandleErrorDBC(c_hdbc2);

    rc = bcp_bind(c_hdbc2, (BYTE *) &c_w_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 3);
    if (rc != SUCCEEDED)
        HandleErrorDBC(c_hdbc2);

    rc = bcp_bind(c_hdbc2, (BYTE *) &c_d_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 4);
    if (rc != SUCCEEDED)
        HandleErrorDBC(c_hdbc2);

    rc = bcp_bind(c_hdbc2, (BYTE *) &c_w_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 5);
    if (rc != SUCCEEDED)

```

```

HandleErrorDBC(c_hdbc2);

rc = bcp_bind(c_hdbc2, (BYTE *) &h_date, 0,
H_DATE_LEN, NULL, 0, SQLCHARACTER, 6);
if (rc != SUCCEEDED)
    HandleErrorDBC(c_hdbc2);

rc = bcp_bind(c_hdbc2, (BYTE *) &h_amount, 0,
SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 7);
if (rc != SUCCEEDED)
    HandleErrorDBC(c_hdbc2);

rc = bcp_bind(c_hdbc2, (BYTE *) h_data, 0, H_DATA_LEN,
NULL, 0, 0, 8);
if (rc != SUCCEEDED)
    HandleErrorDBC(c_hdbc2);

for (i = 0; i < customers_per_district; i++)
{
    c_id = customer_buf[i].c_id;
    c_d_id = customer_buf[i].c_d_id;
    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 != SUCCEEDED)
        HandleErrorDBC(c_hdbc2);

    history_rows_loaded++;
    CheckForCommit(c_hdbc2, c_hstmt2,
history_rows_loaded, "history", &history_time_start->time_start);
}

}

//=====
//
// Function : LoadOrders
//
//=====

void LoadOrders()
{
    LOADER_TIME_STRUCT orders_time_start;
    LOADER_TIME_STRUCT new_order_time_start;
    LOADER_TIME_STRUCT order_line_time_start;
    long w_id;
    short d_id;
    DWORD
dwThreadID[MAX_ORDER_THREADS];
    HANDLE
hThread[MAX_ORDER_THREADS];
    char name[20];
    RETCODE
rc;
    char
bcphint[128];

    // seed with unique number
    seed(aptr->starting_warehouse + 5);

```

```

printf("Loading orders...\n");

// if build index before load...
if ((aptr->build_index == 1) && (aptr->index_order == 1))
{
    BuildIndex("idxordcl");
    BuildIndex("idxnodcl");
    BuildIndex("idxodcl");
}

// initialize bulk copy
sprintf(name, "%s.%s", aptr->database, "t_orders");

LOGFILE_CREATE("\\orders.err")
rc = bcp_init(o_hdbc1, name, NULL, LogFile, 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, CHECK_CONSTRAINTS",
(aptr->num_warehouses * 30000));
    rc = bcp_control(o_hdbc1, BCPHINTS, (void*)
bcphint);
    if (rc != SUCCEED)
        HandleErrorDBC(o_hdbc1);
}

sprintf(name, "%s.%s", aptr->database, "t_new_order");

LOGFILE_CREATE("\\neword.err")
rc = bcp_init(o_hdbc2, name, NULL, LogFile, 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,
CHECK_CONSTRAINTS", (aptr->num_warehouses * 9000));
    rc = bcp_control(o_hdbc2, BCPHINTS, (void*)
bcphint);
    if (rc != SUCCEED)
        HandleErrorDBC(o_hdbc2);
}

sprintf(name, "%s.%s", aptr->database, "t_order_line");

LOGFILE_CREATE("\\ordline.err")
rc = bcp_init(o_hdbc3, name, NULL, LogFile, 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,
CHECK_CONSTRAINTS", (aptr->num_warehouses * 300000));
    rc = bcp_control(o_hdbc3, BCPHINTS, (void*)
bcphint);
    if (rc != SUCCEED)
        HandleErrorDBC(o_hdbc3);
}

orders_rows_loaded = 0;
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 = (long)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,

0,

&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,
0,
(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 i;
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)
{

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
00:00:00.000");

}

}

}

//=====
//
// Function : LoadOrdersTable
//

```

```

//=====
//=====
void LoadOrdersTable(LOADER_TIME_STRUCT*orders_time_start)
{
int i;
long o_id;
short o_d_id;
long o_w_id;
long o_c_id;
short o_carrier_id;
short o_ol_cnt;
short o_all_local;
char o_entry_d[O_ENTRY_D_LEN+1];
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, SQLINT4, 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++)
{
o_id = orders_buf[i].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;
o_carrier_id = orders_buf[i].o_carrier_id;
o_ol_cnt = orders_buf[i].o_ol_cnt;
o_all_local = orders_buf[i].o_all_local;

```

```

FormatDate(&o_entry_d);

// send data to server
rc = bcp_sendrow(o_hdbc1);
if (rc != SUCCEED)
    HandleErrorDBC(o_hdbc1);

orders_rows_loaded++;
CheckForCommit(o_hdbc1, o_hstmt1,
orders_rows_loaded, "orders", &orders_time_start->time_start);
}

// rcint = bcp_batch(o_hdbc1);
// if (rcint < 0)
//     HandleErrorDBC(o_hdbc1);

if ((o_w_id == apr->num_warehouses) && (o_d_id ==
10))
{
    rcint = bcp_done(o_hdbc1);
    if (rcint < 0)
        HandleErrorDBC(o_hdbc1);

    SQLFreeStmt(o_hstmt1, SQL_DROP);
    SQLDisconnect(o_hdbc1);
    SQLFreeConnect(o_hdbc1);

    // if build index after load...
    if ((apr->build_index == 1) &&
(aptr->index_order == 0))
        BuildIndex("idxordcl");

    // build non-clustered index
    if (apr->build_index == 1)
        BuildIndex("idxordnc");
}
}

=====
//
// Function : LoadNewOrderTable
//
=====

void LoadNewOrderTable(LOADER_TIME_STRUCT
*new_order_time_start)
{
    int    i;
    long   o_id;
    short  o_d_id;
    long   o_w_id;
    RETCODE          rc;
    DBINT           rcint;

    // Bind NEW-ORDER data

    rc = bcp_bind(o_hdbc2, (BYTE *) &o_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 1);
    if (rc != SUCCEED)
        HandleErrorDBC(o_hdbc2);

    rc = bcp_bind(o_hdbc2, (BYTE *) &o_d_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 2);
    if (rc != SUCCEED)

```

```

HandleErrorDBC(o_hdbc2);

    rc = bcp_bind(o_hdbc2, (BYTE *) &o_w_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 3);
    if (rc != SUCCEED)
        HandleErrorDBC(o_hdbc2);

    for (i = first_new_order; i < last_new_order; i++)
    {
        o_id = orders_buf[i].o_id;
        o_d_id = orders_buf[i].o_d_id;
        o_w_id = orders_buf[i].o_w_id;

        rc = bcp_sendrow(o_hdbc2);
        if (rc != SUCCEED)
            HandleErrorDBC(o_hdbc2);

        new_order_rows_loaded++;

        CheckForCommit(o_hdbc2, o_hstmt2,
new_order_rows_loaded, "new_order",
&new_order_time_start->time_start);
    }

    // rcint = bcp_batch(o_hdbc2);
    // if (rcint < 0)
    //     HandleErrorDBC(o_hdbc2);

    if ((o_w_id == apr->num_warehouses) && (o_d_id ==
10))
    {
        rcint = bcp_done(o_hdbc2);
        if (rcint < 0)
            HandleErrorDBC(o_hdbc2);

        SQLFreeStmt(o_hstmt2, SQL_DROP);
        SQLDisconnect(o_hdbc2);
        SQLFreeConnect(o_hdbc2);

        // if build index after load...
        if ((apr->build_index == 1) &&
(aptr->index_order == 0))
            BuildIndex("idxnodcl");
    }
}

=====
//
// Function : LoadOrderLineTable
//
=====

void LoadOrderLineTable(LOADER_TIME_STRUCT
*order_line_time_start)
{
    int    i,j;
    long   o_id;
    short  o_d_id;
    long   o_w_id;

    long   ol;
    long   ol_i_id;
    long   ol_supply_w_id;
    short  ol_quantity;

```

```

double   ol_amount;
char     ol_dist_info[DIST_INFO_LEN+1];
char
ol_delivery_d[OL_DELIVERY_D_LEN+1];
        RETCODE          rc;
        DBINT            rcint;

        // bind ORDER-LINE data
rc = bcp_bind(o_hdbc3, (BYTE *) &o_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 1);
if (rc != SUCCEED)
    HandleErrorDBC(o_hdbc3);

rc = bcp_bind(o_hdbc3, (BYTE *) &o_d_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 2);
if (rc != SUCCEED)
    HandleErrorDBC(o_hdbc3);

rc = bcp_bind(o_hdbc3, (BYTE *) &o_w_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 3);
if (rc != SUCCEED)
    HandleErrorDBC(o_hdbc3);

rc = bcp_bind(o_hdbc3, (BYTE *) &o_l, 0, SQL_VARLEN_DATA,
NULL, 0, SQLINT4, 4);
if (rc != SUCCEED)
    HandleErrorDBC(o_hdbc3);

rc = bcp_bind(o_hdbc3, (BYTE *) &o_l_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 5);
if (rc != SUCCEED)
    HandleErrorDBC(o_hdbc3);

rc = bcp_bind(o_hdbc3, (BYTE *) &o_l_supply_w_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 6);
if (rc != SUCCEED)
    HandleErrorDBC(o_hdbc3);

rc = bcp_bind(o_hdbc3, (BYTE *) &o_l_delivery_d, 0,
OL_DELIVERY_D_LEN, NULL, 0, SQLCHARACTER, 7);
if (rc != SUCCEED)
    HandleErrorDBC(o_hdbc3);

rc = bcp_bind(o_hdbc3, (BYTE *) &o_l_quantity, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 8);
if (rc != SUCCEED)
    HandleErrorDBC(o_hdbc3);

rc = bcp_bind(o_hdbc3, (BYTE *) &o_l_amount, 0,
SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 9);
if (rc != SUCCEED)
    HandleErrorDBC(o_hdbc3);

rc = bcp_bind(o_hdbc3, (BYTE *) o_l_dist_info, 0,
DIST_INFO_LEN, NULL, 0, 0, 10);
if (rc != SUCCEED)
    HandleErrorDBC(o_hdbc3);

for (i = 0; i < orders_per_district; i++)
{
    o_id = orders_buf[i].o_id;
    o_d_id = orders_buf[i].o_d_id;
    o_w_id = orders_buf[i].o_w_id;

    for (j=0; j < orders_buf[i].o_ol_cnt; j++)
    {
        ol = orders_buf[i].o_ol[j].ol;

```

```

        ol_i_id =
orders_buf[i].o_ol[j].ol_i_id;
        ol_supply_w_id =
orders_buf[i].o_ol[j].ol_supply_w_id;
        ol_quantity =
orders_buf[i].o_ol[j].ol_quantity;
        ol_amount =
orders_buf[i].o_ol[j].ol_amount;

strcpy(ol_delivery_d,orders_buf[i].o_ol[j].ol_delivery_d);

strcpy(ol_dist_info,orders_buf[i].o_ol[j].ol_dist_info);

rc = bcp_sendrow(o_hdbc3);
if (rc != SUCCEED)

HandleErrorDBC(o_hdbc3);

        order_line_rows_loaded++;
        CheckForCommit(o_hdbc3,
o_hstmt3, order_line_rows_loaded, "order_line",
&order_line_time_start->time_start);
    }
}

// rcint = bcp_batch(o_hdbc3);
// if (rcint < 0)
//     HandleErrorDBC(o_hdbc3);

if ((o_w_id == apr->num_warehouses) && (o_d_id ==
10))
{
    rcint = bcp_done(o_hdbc3);
    if (rcint < 0)
        HandleErrorDBC(o_hdbc3);

    SQLFreeStmt(o_hstmt3, SQL_DROP);
    SQLDisconnect(o_hdbc3);
    SQLFreeConnect(o_hdbc3);

    // if build index after load...
    if ((apr->build_index == 1) &&
(apr->index_order == 0))
        BuildIndex("idxodcl");
}
}

//=====
//
// Function : GetPermutation
//
//=====

void GetPermutation(int perm[], int n)
{
    int i, r, t;

    for (i=1;i<=n;i++)
        perm[i] = i;

    for (i=1;i<=n;i++)

```

```

    {
        r = RandomNumber(i,n);
        t = perm[i];
        perm[i] = perm[r];
        perm[r] = t;
    }
}

//=====
//
// Function : CheckForCommit
//
//=====
void CheckForCommit(HDBC hdbc,
                   HSTMT hstmt,
                   long rows_loaded,
                   char
*table_name,
                   long *time_start)
{
    long    time_end, time_diff;
           // DBINT rcint;

    if ( !(rows_loaded % aptr->batch) )
    {
        // rcint = bcp_batch(hdbc);
        // if (rcint < 0)
        //     HandleErrorDBC(hdbc);

        time_end = (TimeNow() / MILLI);
        time_diff = time_end - *time_start;

        printf("-> Loaded %ld rows into %s in %ld sec -
Total = %d (%.2f rps)\n",
              aptr->batch,
              table_name,
              time_diff,
              rows_loaded,
              (float) aptr->batch /
(time_diff ? time_diff : 1L));

        *time_start = time_end;
    }

    return;
}

//=====
//
// Function : OpenConnections
//
//=====
void OpenConnections()
{
    RETCODE    rc;

```

```

    char        szDriverString[300];
    char        szDriverStringOut[1024];
    SQLSMALLINT cbDriverStringOut;

    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);
    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);

    SQLSetConnectAttr(i_hdbc1, SQL_COPT_SS_BCP, (void
*)SQL_BCP_ON, SQL_IS_INTEGER);
    SQLSetConnectAttr(w_hdbc1, SQL_COPT_SS_BCP, (void
*)SQL_BCP_ON, SQL_IS_INTEGER);
    SQLSetConnectAttr(c_hdbc1, SQL_COPT_SS_BCP, (void
*)SQL_BCP_ON, SQL_IS_INTEGER);
    SQLSetConnectAttr(c_hdbc2, SQL_COPT_SS_BCP, (void
*)SQL_BCP_ON, SQL_IS_INTEGER);
    SQLSetConnectAttr(o_hdbc1, SQL_COPT_SS_BCP, (void
*)SQL_BCP_ON, SQL_IS_INTEGER);
    SQLSetConnectAttr(o_hdbc2, SQL_COPT_SS_BCP, (void
*)SQL_BCP_ON, SQL_IS_INTEGER);
    SQLSetConnectAttr(o_hdbc3, SQL_COPT_SS_BCP, (void
*)SQL_BCP_ON, SQL_IS_INTEGER);

    // Open connections to SQL Server

    // Connection 1

    sprintf( szDriverString , "DRIVER={SQL
Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s",
aptr->server,
aptr->user,
aptr->password,
aptr->database );

    rc = SQLSetConnectOption (i_hdbc1,
SQL_PACKET_SIZE, aptr->pack_size);
    if (rc != SUCCEED)
        HandleErrorDBC(i_hdbc1);

    rc = SQLDriverConnect ( i_hdbc1,
NULL,
(SQLCHAR*)&szDriverString[0],
SQL_NTS,
(SQLCHAR*)&szDriverStringOut[0],

```

```

sizeof(szDriverStringOut),
&cbDriverStringOut,
SQL_DRIVER_NOPROMPT);
    if (rc != SUCCEEDED)
        HandleErrorDBC(i_hdbc1);

        // Connection 2

        sprintf( szDriverString , "DRIVER={SQL
Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s",
aptr->server,
aptr->user,
aptr->password,
aptr->database );

        rc = SQLSetConnectOption (w_hdbc1,
SQL_PACKET_SIZE, aptr->pack_size);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = SQLDriverConnect ( w_hdbc1,
NULL,
(SQLCHAR*)&szDriverString[0],
SQL_NTS,
(SQLCHAR*)&szDriverStringOut[0],
sizeof(szDriverStringOut),
&cbDriverStringOut,
SQL_DRIVER_NOPROMPT);
    if (rc != SUCCEEDED)
        HandleErrorDBC(w_hdbc1);

        // Connection 3

        sprintf( szDriverString , "DRIVER={SQL
Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s",
aptr->server,
aptr->user,
aptr->password,
aptr->database );

        rc = SQLSetConnectOption (c_hdbc1,
SQL_PACKET_SIZE, aptr->pack_size);
        if (rc != SUCCEEDED)
            HandleErrorDBC(c_hdbc1);

        rc = SQLDriverConnect ( c_hdbc1,
NULL,
(SQLCHAR*)&szDriverString[0],

```

```

SQL_NTS,
(SQLCHAR*)&szDriverStringOut[0],
sizeof(szDriverStringOut),
&cbDriverStringOut,
SQL_DRIVER_NOPROMPT);
    if (rc != SUCCEEDED)
        HandleErrorDBC(c_hdbc1);

        // Connection 4

        sprintf( szDriverString , "DRIVER={SQL
Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s",
aptr->server,
aptr->user,
aptr->password,
aptr->database );

        rc = SQLSetConnectOption (c_hdbc2,
SQL_PACKET_SIZE, aptr->pack_size);
        if (rc != SUCCEEDED)
            HandleErrorDBC(c_hdbc2);

        rc = SQLDriverConnect ( c_hdbc2,
NULL,
(SQLCHAR*)&szDriverString[0],
SQL_NTS,
(SQLCHAR*)&szDriverStringOut[0],
sizeof(szDriverStringOut),
&cbDriverStringOut,
SQL_DRIVER_NOPROMPT);
    if (rc != SUCCEEDED)
        HandleErrorDBC(c_hdbc2);

        // Connection 5

        sprintf( szDriverString , "DRIVER={SQL
Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s",
aptr->server,
aptr->user,
aptr->password,
aptr->database );

        rc = SQLSetConnectOption (o_hdbc1,
SQL_PACKET_SIZE, aptr->pack_size);
        if (rc != SUCCEEDED)
            HandleErrorDBC(o_hdbc1);

        rc = SQLDriverConnect ( o_hdbc1,

```



```

NULL,
(SQLCHAR*)&szDriverString[0],
SQL_NTS,
(SQLCHAR*)&szDriverStringOut[0],
sizeof(szDriverStringOut),
&cbDriverStringOut,
SQL_DRIVER_NOPROMPT);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc1);

        // Connection 6
        sprintf( szDriverString , "DRIVER={SQL
Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s",
aptr->server,
aptr->user,
aptr->password,
aptr->database );

        rc = SQLSetConnectOption (o_hdbc2,
SQL_PACKET_SIZE, aptr->pack_size);
        if (rc != SUCCEEDED)
            HandleErrorDBC(o_hdbc2);

        rc = SQLDriverConnect ( o_hdbc2,
NULL,
(SQLCHAR*)&szDriverString[0],
SQL_NTS,
(SQLCHAR*)&szDriverStringOut[0],
sizeof(szDriverStringOut),
&cbDriverStringOut,
SQL_DRIVER_NOPROMPT);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc2);

        // Connection 7
        sprintf( szDriverString , "DRIVER={SQL
Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s",
aptr->server,
aptr->user,
aptr->password,
aptr->database );

        rc = SQLSetConnectOption (o_hdbc3,
SQL_PACKET_SIZE, aptr->pack_size);

```

```

        if (rc != SUCCEEDED)
            HandleErrorDBC(o_hdbc3);

        rc = SQLDriverConnect ( o_hdbc3,
NULL,
(SQLCHAR*)&szDriverString[0],
SQL_NTS,
(SQLCHAR*)&szDriverStringOut[0],
sizeof(szDriverStringOut),
&cbDriverStringOut,
SQL_DRIVER_NOPROMPT);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc3);
}
//=====
//
// Function name: BuildIndex
//
//=====

void BuildIndex(char *index_script)
{
    char    cmd[256];

    printf("Starting index creation: %s\n",index_script);

    sprintf(cmd, "isql -S%s -U%s -P%s -e -i%s\\%s.sql >
%s\\%s.log",
aptr->server,
aptr->user,
aptr->password,
aptr->index_script_path,
index_script,
LogBase,
index_script);

    system(cmd);

    printf("Finished index creation: %s\n",index_script);
}

void HandleErrorDBC (SQLHDBC hdbc1)
{
    SQLCHAR    SqlState[6],
Msg[SQL_MAX_MESSAGE_LENGTH];
    SQLINTEGER NativeError;
    SQLSMALLINT i, MsgLen;
    SQLRETURN rc2;
    char    timebuf[128];
    char    datebuf[128];
    FILE    *fp1;

    i = 1;
    while (( rc2 = SQLGetDiagRec(SQL_HANDLE_DBC,
hdbc1, 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);

    LOGFILE_CREATE("\\tpccldr.err")
    fp1 = fopen(LogFile,"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 HandleErrorSTMT (HSTMT hstmt1)
{
    SQLCHAR                SqlState[6],
Msg[SQL_MAX_MESSAGE_LENGTH];
    SQLINTEGER NativeError;
    SQLSMALLINT i, MsgLen;
    SQLRETURN rc2;
    char                timebuf[128];
    char                datebuf[128];
    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);

        LOGFILE_CREATE("\\tpccldr.err")
        fp1 = fopen(LogFile,"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 : CheckDataBase
//
//=====

void CheckDataBase()
{
    RETCODE rc;

    char                szDriverString[300];
    char                szDriverStringOut[1024];
    char                TablesBitMap[9] =
{"000000000"};
    int                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_UIINTEGER);
    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 %s\ directory for database
creation errors.\n", LogBase);

        // 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;

```

```

// iterate through the bitmap to display which
table(s) is actually missing
for (i = 0; i <= 8; i++)
{
    switch(i)
    {
        case 0:
            if (TablesBitMap[i] == '0')
            {
                printf("The
Warehouse table is missing or damaged.\n");
                ExitFlag = 1;
            }
            break;
        case 1:
            if (TablesBitMap[i] == '0')
            {
                printf("The
District table is missing or damaged.\n");
                ExitFlag = 1;
            }
            break;
        case 2:
            if (TablesBitMap[i] == '0')
            {
                printf("The
Customer table is missing or damaged.\n");
                ExitFlag = 1;
            }
            break;
        case 3:
            if (TablesBitMap[i] == '0')
            {
                printf("The
History table is missing or damaged.\n");
                ExitFlag = 1;
            }
            break;
        case 4:
            if (TablesBitMap[i] == '0')
            {
                printf("The
New_Order table is missing or damaged.\n");
                ExitFlag = 1;
            }
            break;
        case 5:
            if (TablesBitMap[i] == '0')
            {
                printf("The
Orders table is missing or damaged.\n");
                ExitFlag = 1;
            }
            break;
        case 6:
            if (TablesBitMap[i] == '0')
            {
                printf("The
Order_Line table is missing or damaged.\n");
                ExitFlag = 1;
            }
            break;
        case 7:
            if (TablesBitMap[i] == '0')
            {
                printf("The
Item table is missing or damaged.\n");
                ExitFlag = 1;
            }
    }
}

}
break;
case 8:
    if (TablesBitMap[i] == '0')
    {
        printf("The
Stock table is missing or damaged.\n");
        ExitFlag = 1;
    }
    break;
}
}

// if one or more tables are missing, display
message and exit the loader
if (ExitFlag = 1)
{
    printf("\nExiting TPC-C Loader!\n");
    printf("\nCheck %s\\ directory for
database\n", LogBase);
    printf("or table creation errors.\n");

    // cleanup database connections and
handles
SQLFreeHandle(SQL_HANDLE_STMT,v_hstmt);
SQLDisconnect(v_hdbc);
SQLFreeHandle(SQL_HANDLE_DBC,v_hdbc);
    exit(1);
}

// cleanup database connections and handles
SQLFreeHandle(SQL_HANDLE_STMT,v_hstmt);
SQLDisconnect(v_hdbc);
SQLFreeHandle(SQL_HANDLE_DBC,v_hdbc);

return;
}

tpccldr.mak
# Microsoft Developer Studio Generated NMAKE File, Format
Version 4.10
# ** DO NOT EDIT **

# TARGETTYPE "Win32 (x86) Console Application" 0x0103
!IF "$(CFG)" == ""
CFG=tpccldr - Win32 Debug
!MESSAGE No configuration specified. Defaulting to tpccldr - Win32
Debug.
!ENDIF

!IF "$(CFG)" != "tpccldr - Win32 Release" && "$(CFG)" !=\
"tpccldr - Win32 Debug"
!MESSAGE Invalid configuration "$(CFG)" specified.
!MESSAGE You can specify a configuration when running NMAKE
on this makefile
!MESSAGE by defining the macro CFG on the command line. For
example:
!MESSAGE
!MESSAGE NMAKE /f "tpccldr.mak" CFG="tpccldr - Win32 Debug"
!MESSAGE
!MESSAGE Possible choices for configuration are:

```

```

!MESSAGE
!MESSAGE "tpccldr - Win32 Release" (based on "Win32 (x86)
Console Application")
!MESSAGE "tpccldr - Win32 Debug" (based on "Win32 (x86)
Console Application")
!MESSAGE
!ERROR An invalid configuration is specified.
!ENDIF

!IF "$(OS)" == "Windows_NT"
NULL=
!ELSE
NULL=nul
!ENDIF
#####
#####
# Begin Project
# PROP Target_Last_Scanned "tpccldr - Win32 Debug"
RSC=rc.exe
CPP=cl.exe

!IF "$(CFG)" == "tpccldr - Win32 Release"

# PROP BASE Use_MFC 0
# PROP BASE Use_Debug_Libraries 0
# PROP BASE Output_Dir "Release"
# PROP BASE Intermediate_Dir "Release"
# PROP BASE Target_Dir ""
# PROP Use_MFC 0
# PROP Use_Debug_Libraries 0
# PROP Output_Dir "bin"
# PROP Intermediate_Dir "objects"
# PROP Target_Dir ""
OUTDIR=.\\bin
INTDIR=.\\objects

ALL : "$(OUTDIR)\tpccldr.exe"

CLEAN :
    -@erase "$(INTDIR)\getargs.obj"
    -@erase "$(INTDIR)\random.obj"
    -@erase "$(INTDIR)\strings.obj"
    -@erase "$(INTDIR)\time.obj"
    -@erase "$(INTDIR)\tpccldr.obj"
    -@erase "$(OUTDIR)\tpccldr.exe"

"$(OUTDIR)":
    if not exist "$(OUTDIR)\$(NULL)" mkdir "$(OUTDIR)"

"$(INTDIR)":
    if not exist "$(INTDIR)\$(NULL)" mkdir "$(INTDIR)"

# ADD BASE CPP /nologo /W3 /GX /O2 /D "WIN32" /D "NDEBUG"
/D "_CONSOLE" /YX /c
# ADD CPP /nologo /MT /W3 /GX /O2 /I "c:\mssql\dblib\include" /D
"NDEBUG" /D "WIN32" /D "_CONSOLE" /D "DBNTWIN32" /c
# SUBTRACT CPP /YX
CPP_PROJ=/nologo /MT /W3 /GX /O2 /I "c:\mssql\dblib\include" /D
"NDEBUG" /D\
"WIN32" /D "_CONSOLE" /D "DBNTWIN32" /Fo "$(INTDIR)\"/c
CPP_OBJS=.\\objects/
CPP_SBRS=.\\
# ADD BASE RSC /I 0x409 /d "NDEBUG"
# ADD RSC /I 0x409 /d "NDEBUG"
BSC32=bscmake.exe
# ADD BASE BSC32 /nologo
# ADD BSC32 /nologo
BSC32_FLAGS=/nologo /o "$(OUTDIR)\tpccldr.bsc"

```

```

BSC32_SBRS= \

LINK32=link.exe
# 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 odbccp32.lib /nologo /subsystem:console /machine:I386
# 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 uuid.lib odbc32.lib odbccp32.lib /nologo
/subsystem:console /pdb:none /machine:I386
LINK32_FLAGS=c:\mssql\dblib\lib\ntwdblib.lib kernel32.lib
user32.lib gdi32.lib\
winspool.lib comdlg32.lib advapi32.lib shell32.lib ole32.lib
oleaut32.lib\
uuid.lib odbc32.lib odbccp32.lib /nologo /subsystem:console
/pdb:none\
/machine:I386 /out:"$(OUTDIR)\tpccldr.exe"
LINK32_OBJS= \
    "$(INTDIR)\getargs.obj" \
    "$(INTDIR)\random.obj" \
    "$(INTDIR)\strings.obj" \
    "$(INTDIR)\time.obj" \
    "$(INTDIR)\tpccldr.obj"

"$(OUTDIR)\tpccldr.exe": "$(OUTDIR)" $(DEF_FILE)
$(LINK32_OBJS)
    $(LINK32) @<<
    $(LINK32_FLAGS)$(LINK32_OBJS)
<<

!ELSEIF "$(CFG)" == "tpccldr - Win32 Debug"

# 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
# PROP Use_Debug_Libraries 1
# PROP Output_Dir "bin"
# PROP Intermediate_Dir "objects"
# PROP Target_Dir ""
OUTDIR=.\\bin
INTDIR=.\\objects

ALL : "$(OUTDIR)\tpccldr.exe"

CLEAN :
    -@erase "$(INTDIR)\getargs.obj"
    -@erase "$(INTDIR)\random.obj"
    -@erase "$(INTDIR)\strings.obj"
    -@erase "$(INTDIR)\time.obj"
    -@erase "$(INTDIR)\tpccldr.obj"
    -@erase "$(OUTDIR)\tpccldr.exe"

"$(OUTDIR)":
    if not exist "$(OUTDIR)\$(NULL)" mkdir "$(OUTDIR)"

"$(INTDIR)":
    if not exist "$(INTDIR)\$(NULL)" mkdir "$(INTDIR)"

# ADD BASE CPP /nologo /W3 /Gm /GX /Zi /Od /D "WIN32" /D
"_DEBUG" /D "_CONSOLE" /YX /c
# ADD CPP /nologo /MTd /W3 /Gm /GX /Zi /Od /I
"c:\mssql\dblib\include" /D "_DEBUG" /D "WIN32" /D "_CONSOLE"
/D "DBNTWIN32" /c

```

```

# SUBTRACT CPP /YX
CPP_PROJ=/nologo /MTd /W3 /Gm /GX /Zi /Od /I
"c:\mssql\dblib\include" /D\
 "_DEBUG" /D "WIN32" /D "_CONSOLE" /D "DBNTWIN32"
/Fo"${INTDIR}/"
/Fd"${INTDIR}/" /c
CPP_OBJS=.objects/
CPP_SBRS=.
# ADD BASE RSC /I 0x409 /d "_DEBUG"
# ADD RSC /I 0x409 /d "_DEBUG"
BSC32=bscmake.exe
# ADD BASE BSC32 /nologo
# ADD BSC32 /nologo
BSC32_FLAGS=/nologo /o"${OUTDIR}/tpccldr.bsc"
BSC32_SBRS= \

LINK32=link.exe
# 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 odbccp32.lib /nologo /subsystem:console /debug
/machine:I386
# 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 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 gdi32.lib\
winspool.lib comdlg32.lib advapi32.lib shell32.lib ole32.lib
oleaut32.lib\
uuid.lib odbc32.lib odbccp32.lib /nologo /subsystem:console
/pdb:none /debug\
/machine:I386 /out:"${OUTDIR}/tpccldr.exe"
LINK32_OBJS= \
    "${INTDIR}\getargs.obj" \
    "${INTDIR}\random.obj" \
    "${INTDIR}\strings.obj" \
    "${INTDIR}\time.obj" \
    "${INTDIR}\tpccldr.obj"

"${OUTDIR}/tpccldr.exe": "${OUTDIR}" $(DEF_FILE)
$(LINK32_OBJS)
$(LINK32) @<<
$(LINK32_FLAGS)$(LINK32_OBJS)
<<

!ENDIF

.c{$(CPP_OBJS)}.obj:
    $(CPP) $(CPP_PROJ) $<

.cpp{$(CPP_OBJS)}.obj:
    $(CPP) $(CPP_PROJ) $<

.cxx{$(CPP_OBJS)}.obj:
    $(CPP) $(CPP_PROJ) $<

.c{$(CPP_SBRS)}.sbr:
    $(CPP) $(CPP_PROJ) $<

.cpp{$(CPP_SBRS)}.sbr:
    $(CPP) $(CPP_PROJ) $<

.cxx{$(CPP_SBRS)}.sbr:
    $(CPP) $(CPP_PROJ) $<

#####
#####
# Begin Target

```

```

# Name "tpccldr - Win32 Release"
# Name "tpccldr - Win32 Debug"

!IF "$(CFG)" == "tpccldr - Win32 Release"

!ELSEIF "$(CFG)" == "tpccldr - Win32 Debug"

!ENDIF

#####
#####
# Begin Source File

SOURCE=.src\random.c
DEP_CPP_RAND0=\
    ".src\tpcc.h" \
    "\mssql\dblib\include\sqldb.h" \
    "\mssql\dblib\include\sqlfront.h"

"${INTDIR}\random.obj": $(SOURCE) $(DEP_CPP_RAND0)
"${INTDIR}"
$(CPP) $(CPP_PROJ) $(SOURCE)

# End Source File
#####
#####
# Begin Source File

SOURCE=.src\strings.c
DEP_CPP_STRIN=\
    ".src\tpcc.h" \
    "\mssql\dblib\include\sqldb.h" \
    "\mssql\dblib\include\sqlfront.h"

"${INTDIR}\strings.obj": $(SOURCE) $(DEP_CPP_STRIN)
"${INTDIR}"
$(CPP) $(CPP_PROJ) $(SOURCE)

# End Source File
#####
#####
# Begin Source File

SOURCE=.src\time.c
DEP_CPP_TIME_=\
    ".src\tpcc.h" \
    "\mssql\dblib\include\sqldb.h" \
    "\mssql\dblib\include\sqlfront.h"

"${INTDIR}\time.obj": $(SOURCE) $(DEP_CPP_TIME_)
"${INTDIR}"
$(CPP) $(CPP_PROJ) $(SOURCE)

# End Source File
#####
#####
# Begin Source File

SOURCE=.src\tpccldr.c
DEP_CPP_TPCCL=\
    ".src\tpcc.h" \

```

```
"\mssql\dblib\include\sqldb.h"\
"\mssql\dblib\include\sqlfront.h"\

"${INTDIR}\tpccldr.obj" : $(SOURCE) $(DEP_CPP_TPCCL)
"${INTDIR}"
$(CPP) $(CPP_PROJ) $(SOURCE)

# End Source File
#####
#####
# Begin Source File

SOURCE=. \src\getargs.c
DEP_CPP_GETAR=\
    ".\src\tpcc.h"\
    "\mssql\dblib\include\sqldb.h"\
    "\mssql\dblib\include\sqlfront.h"\

"${INTDIR}\getargs.obj" : $(SOURCE) $(DEP_CPP_GETAR)
"${INTDIR}"
$(CPP) $(CPP_PROJ) $(SOURCE)

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

# Appendix C: Tunable Parameters

## Database Startup Parameters

Microsoft SQL Server 2000 Startup Parameters

```
c:\Program Files\Microsoft SQL Server\MSSQL\Binn\sqlservr.exe -c -x
-t3502 -g800
```

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
-g800 Specifies the amount of memory that is set aside for allocations
not from the buffer pool
```

## Database Information

```
1> 2> 3> 4> 5> 6> 7> 8> 9> 10> 11>
-- File:  VERSION.SQL
--      Microsoft TPC-C Benchmark Kit Ver. 4.30
--      Copyright Microsoft, 2000
-- Purpose: Returns SQL Server version string
```

```
print ''
select convert(char(30), getdate(),21)
print ''
```

```
-----
2001-03-19 12:39:54.107
```

(1 row affected)

```
1> 2> 3> 4> 5> Server Name:
```

```
print 'Server Name:'
select @@servername
print ''
```

```
-----
SGNODE01
```

(1 row affected)

```
1> 2> 3> 4> 5> Server Version:
```

```
print 'Server Version:'
select @@version
print ''
```

```
-----
```

```
-----
-----
Microsoft SQL Server 2000 - 8.00.208 (Intel X86)
Sep 27 2000 14:06:38
Cop
yright (c) 1988-2000 Microsoft Corporation
Enterprise Edition on Windo
ws NT 5.0 (Build 2195: Service Pack 1)
```

(1 row affected)

```
1> 2>
1> 2> 3> 4> 5> 6> 7> 8> 9> 10>
-- File:  CONFIG.SQL
--      Microsoft TPC-C Benchmark Kit Ver. 4.30
--      Copyright Microsoft, 2000
-- Purpose: Collects SQL Server configuration parameters
```

```
print ''
select convert(char(30), getdate(),21)
print ''
```

```
-----
2001-03-19 12:39:54.590
```

(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	0	2147483647	255	255
allow updates	0	1	0	0
awe enabled	0	1	1	1
c2 audit mode	0	1	0	0
cost threshold for parallelism	0	32767	5	5
cursor threshold	-1	2147483647	-1	-1
default full-text language	0	2147483647	1033	1033
default language	0	9999	0	0
fill factor (%)	0	100	0	0
index create memory (KB)	0	704 2147483647	0	0
lightweight pooling	0	1	0	0
locks	5000	2147483647	0	0
max degree of parallelism	0	32	0	0
max server memory (MB)	4	2147483647	2147483647	2147483647
max text repl size (B)	0	2147483647	65536	65536
max worker threads	32	32767	550	550
media retention	0	365	0	0
min memory per query (KB)	1024	2147483647	1024	1024
min server memory (MB)	0	2147483647	0	0



```

nested triggers          0      1      1      1
network packet size (B) 512    65536   4096
4096
open objects            0 2147483647      0      0
priority boost          0      1      1      1
query governor cost limit      0 2147483647      0      0

query wait (s)          -1 2147483647     -1     -1
recovery interval (min)  0    32767     56     56
remote access           0      1      1      1
remote login timeout (s)  0 2147483647      0      0

remote proc trans       0      1      0      0
remote query timeout (s)  0 2147483647      0      0

scan for startup proc   0      1      0      0
set working set size    0      1      0      0
show advanced options   0      1      1      1
two digit year cutoff   1753   9999     2049   2049

user connections        0    32767      0      0
user options            0    32767      0      0

```

1>

## Software Update

Software Update

SQL Server QFE 208 was applied to all 4 server nodes and is available.  
Windows 2000 SP1 was applied to all 4 server nodes and is available.

## Microsoft Windows 2000 Datacenter Configuration

### boot.ini

```

[boot loader]
timeout=5
default=multi(0)disk(0)rdisk(0)partition(1)\WINNT
[operating systems]
multi(0)disk(0)rdisk(0)partition(1)\WINNT="Microsoft Windows
2000 Datacenter Server /PAE" /fastdetect /PAE
multi(0)disk(0)rdisk(0)partition(1)\WINNT="Microsoft Windows
2000 Datacenter Server" /fastdetect

```

### Enabling AWE

The following procedure was used on each node to enable SQL Server access to AWE memory:

- 1) Ran gpedit.msc to open the Group Policy Editor window.
- 2) Selected Computer Configuration -> Windows Settings -> Security Settings -> Local Policies -> User Rights Assignment.
- 3) Double-clicked on Lock Pages in Memory in the right hand pane of the editor.
- 4) Clicked Add.
- 5) Selected the Administrators group.
- 6) Click OK to set the local security policy setting.
- 7) Click OK to return to the Group Policy Editor.
- 8) Close the editor. Reboot the node.

## Enabling VIA

The following procedure was used on each node to enable VIA support for SQL Server:

- 1) Open the SQL Client Network Utility.
- 2) Enable VIA as a protocol.
- 3) Select the Alias page. Press the Add button.
- 4) Check the VIA radio button.
- 5) Create a server alias.
- 6) Close the SQL Client Network Utility.
- 7) Open the SQL Server Network Utility.
- 8) Add VIA to the list of enabled protocols.
- 9) Close the SQL Server Network Utility.

## System Information Report for Server Nodes

This hardware configuration inventory was obtained from 1 of 4 nodes.

All 4 nodes are identically configured.

System Information report written at: 03/14/2001 08:00:13 AM  
[System Information]

[ Following are sub-categories of this main category ]

[System Summary]

Item	Value
OS Name	Microsoft Windows 2000 Datacenter Server
Version	5.0.2195 Service Pack 1 Build 2195
OS Manufacturer	Microsoft Corporation
System Name	SGNODE01
System Manufacturer	IBM
System Model	IBM Server -[8681]-
System Type	X86-based PC
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
BIOS Version	IBM BIOS Ver 6.0
Windows Directory	C:\WINNT
System Directory	C:\WINNT\System32
Boot Device	\Device\Harddisk0\Partition1
Locale	United States
User Name	SGNODE01\Administrator
Time Zone	Eastern Standard Time
Total Physical Memory	33,291,576 KB
Available Physical Memory	32,866,272 KB
Total Virtual Memory	68,305,188 KB
Available Virtual Memory	67,782,752 KB
Page File Space	35,013,612 KB
Page File	C:\pagefile.sys

[Hardware Resources]

[ Following are sub-categories of this main category ]

[Conflicts/Sharing]

Resource Device  
 IRQ 51 Adaptec AIC-7896/AIC-7897 PCI Ultra2 SCSI Controller  
 IRQ 51 Adaptec AIC-7896/AIC-7897 PCI Ultra2 SCSI Controller

[DMA]

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

[Forced Hardware]

Device PNP Device ID  
 No Forced Hardware

[I/O]

Address Range	Device	Status
0x0000-0x0CF7	PCI bus	OK
0x0000-0x0CF7	Direct memory access controller	OK
0x0D00-0x5FFF	PCI bus	OK
0x2000-0x20FF	Other PCI Bridge Device	OK
0x2180-0x218F	Other PCI Bridge Device	OK
0x03B0-0x03BB	S3 Inc. Trio3D	OK
0x03C0-0x03DF	S3 Inc. Trio3D	OK
0x2200-0x22FF	IBM ServeRAID 4H Controller	OK
0x21C0-0x21FF	IBM 10/100 Ethernet Server Adapter	OK
0x2300-0x23FF	Adaptec AIC-7896/AIC-7897 PCI Ultra2 SCSI Controller	OK
0x2400-0x24FF	Adaptec AIC-7896/AIC-7897 PCI Ultra2 SCSI Controller	OK
0x0A79-0x0A79	ISAPNP Read Data Port	OK
0x0279-0x0279	ISAPNP Read Data Port	OK
0x02F4-0x02F7	ISAPNP Read Data Port	OK
0x002E-0x002F	Motherboard resources	OK
0x0438-0x0439	Motherboard resources	OK
0x0430-0x0437	Motherboard resources	OK
0x0060-0x0060	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard	OK
0x0064-0x0064	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard	OK
0x03F0-0x03F5	Standard floppy disk controller	OK
0x03F7-0x03F7	Standard floppy disk controller	OK
0x0378-0x037F	Printer Port (LPT1)	OK
0x03F8-0x03FF	Communications Port (COM1)	OK
0x02F8-0x02FF	Communications Port (COM2)	OK
0x0020-0x0021	Advanced programmable interrupt controller	OK
0x00A0-0x00A1	Advanced programmable interrupt controller	OK
0x0080-0x008F	Direct memory access controller	OK
0x00C0-0x00DF	Direct memory access controller	OK
0x0040-0x0043	System timer	OK
0x0070-0x0073	System CMOS/real time clock	OK
0x0061-0x0061	System speaker	OK
0x00F0-0x00FF	Numeric data processor	OK
0x04D0-0x04D1	Motherboard resources	OK
0x00E8-0x00EF	Motherboard resources	OK
0x5FF0-0x5FFF	Intel(r) 82371AB/EB PCI Bus Master IDE Controller	OK
0x01F0-0x01F7	Primary IDE Channel	OK
0x03F6-0x03F6	Primary IDE Channel	OK
0x5FC0-0x5FDF	Intel 82371AB/EB PCI to USB Universal Host Controller	OK
0x6000-0xAFFF	PCI bus	OK
0x6000-0xAFFF	IBM ServeRAID 4H Controller	OK
0x6100-0x61FF	IBM ServeRAID 4H Controller	OK
0xB000-0xCFFF	PCI bus	OK

0xB000-0xCFFF	IBM ServeRAID 4H Controller	OK
0xB100-0xB1FF	IBM ServeRAID 4H Controller	OK
0xD000-0xFFFF	PCI bus	OK
0xD000-0xFFFF	IBM ServeRAID 4H Controller	OK
0xD100-0xD1FF	IBM ServeRAID 4H Controller	OK

[IRQs]

IRQ Number	Device
9	Microsoft ACPI-Compliant System
5	Other PCI Bridge Device
25	IBM ServeRAID 4H Controller
26	cLAN Host Adapter
27	IBM 10/100 Ethernet Server Adapter
51	Adaptec AIC-7896/AIC-7897 PCI Ultra2 SCSI Controller
51	Adaptec AIC-7896/AIC-7897 PCI Ultra2 SCSI Controller
1	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard
12	PS/2 Compatible Mouse
6	Standard floppy disk controller
4	Communications Port (COM1)
3	Communications Port (COM2)
8	System CMOS/real time clock
13	Numeric data processor
14	Primary IDE Channel
49	Intel 82371AB/EB PCI to USB Universal Host Controller
19	IBM ServeRAID 4H Controller
20	IBM ServeRAID 4H Controller
23	IBM ServeRAID 4H Controller
24	IBM ServeRAID 4H Controller
21	IBM ServeRAID 4H Controller
22	IBM ServeRAID 4H Controller

[Memory]

Range	Device	Status
0xA000-0xBFFFF	PCI bus	OK
0xA0000-0xBFFFF	S3 Inc. Trio3D	OK
0xC8000-0xDFFFF	PCI bus	OK
0xF000000-0xFC1FFFFF	PCI bus	OK
0xFEC00000-0xFFFFFFFF	PCI bus	OK
0xFEC00000-0xFFFFFFFF	Advanced programmable interrupt controller	OK
0xFC1FFC00-0xFC1FFCFF	Other PCI Bridge Device	OK
0xFC1E0000-0xFC1EFFFF	Other PCI Bridge Device	OK
0xF8000000-0xFBFFFFFF	S3 Inc. Trio3D	OK
0xFC000000-0xFC0FFFFF	IBM ServeRAID 4H Controller	OK
0xFC1C0000-0xFC1DFFFF	cLAN Host Adapter	OK
0xF7E00000-0xF7FFFFFF	cLAN Host Adapter	OK
0xF6000000-0xF6FFFFFF	cLAN Host Adapter	OK
0xFC1B0000-0xFC1BFFFF	cLAN Host Adapter	OK
0xFC1FE000-0xFC1FEFFF	IBM 10/100 Ethernet Server Adapter	OK
0xFC180000-0xFC19FFFF	IBM 10/100 Ethernet Server Adapter	OK
0xFC1FD000-0xFC1FDFFF	Adaptec AIC-7896/AIC-7897 PCI Ultra2 SCSI Controller	OK
0xFC1FC000-0xFC1FCFFF	Adaptec AIC-7896/AIC-7897 PCI Ultra2 SCSI Controller	OK
0xFC200000-0xFD7FFFFF	PCI bus	OK
0xFD700000-0xFD7FFFFF	IBM ServeRAID 4H Controller	OK
0xFD600000-0xFD6FFFFF	IBM ServeRAID 4H Controller	OK
0xFD800000-0xFE1FFFFF	PCI bus	OK
0xFE100000-0xFE1FFFFF	IBM ServeRAID 4H Controller	OK
0xFE000000-0xFE0FFFFF	IBM ServeRAID 4H Controller	OK
0xFE200000-0xFEBFFFFFF	PCI bus	OK
0xFEB00000-0xFEBFFFFFF	IBM ServeRAID 4H Controller	OK
0xFE000000-0xFE0FFFFF	IBM ServeRAID 4H Controller	OK

[Components]

[ Following are sub-categories of this main category ]

[Multimedia]

[ Following are sub-categories of this main category ]

[Audio Codecs]

Codec Version	Manufacturer Size	Description Creation Date	Status	File
2.05.53	195.00 KB (199,680 bytes)	Intel Corporation IAC25_32.AX 7/27/2000 8:00:00 PM	OK	c:\winnt\system32\iac25_32.ax
4.4.3385	33.27 KB (34,064 bytes)	Microsoft Corporation LHACM.ACM 2/8/2001 4:08:52 PM	OK	c:\winnt\system32\lhacm.acm
5.00.2134.1	14.77 KB (15,120 bytes)	Microsoft Corporation MSADP32.ACM 7/27/2000 8:00:00 PM	OK	c:\winnt\system32\msadp32.acm
4.4.3385	106.77 KB (109,328 bytes)	Microsoft Corporation MSG723.ACM 2/8/2001 4:08:51 PM	OK	c:\winnt\system32\msg723.acm
5.00.2134.1	22.27 KB (22,800 bytes)	Microsoft Corporation MSGSM32.ACM 7/27/2000 8:00:00 PM	OK	c:\winnt\system32\msgsm32.acm
1.01	9.27 KB (9,488 bytes)	DSP GROUP, INC. TSSOFT32.ACM 7/27/2000 8:00:00 PM	OK	c:\winnt\system32\tsoft32.acm
5.00.2134.1	10.27 KB (10,512 bytes)	Microsoft Corporation MSG711.ACM 7/27/2000 8:00:00 PM	OK	c:\winnt\system32\msg711.acm
5.00.2134.1	16.27 KB (16,656 bytes)	Microsoft Corporation IMAADP32.ACM 7/27/2000 8:00:00 PM	OK	c:\winnt\system32\imaadp32.acm

[Video Codecs]

Codec Version	Manufacturer Size	Description Creation Date	Status	File
5.10	737.50 KB (755,200 bytes)	Intel Corporation IR50_32.DLL 7/27/2000 8:00:00 PM	OK	c:\winnt\system32\ir50_32.dll
4.4.3385	163.77 KB (167,696 bytes)	Microsoft Corporation MSH261.DRV 2/8/2001 4:08:52 PM	OK	c:\winnt\system32\msh261.drv
4.4.3385	252.27 KB (258,320 bytes)	Microsoft Corporation MSH263.DRV 2/8/2001 4:08:22 PM	OK	c:\winnt\system32\msh263.drv
5.00.2134.1	27.27 KB (27,920 bytes)	Microsoft Corporation MSVIDC32.DLL 7/27/2000 8:00:00 PM	OK	c:\winnt\system32\msvidc32.dll
5.00.2134.1	10.77 KB (11,024 bytes)	Microsoft Corporation MSRLE32.DLL 7/27/2000 8:00:00 PM	OK	c:\winnt\system32\msrle32.dll
Not Available	194.50 KB (199,168 bytes)	Intel(R) Corporation IR32_32.DLL 7/27/2000 8:00:00 PM	Not Available	c:\winnt\system32\ir32_32.dll
1.10.0.6	108.00 KB (110,592 bytes)	Radius Inc. ICCVID.DLL 7/27/2000 8:00:00 PM	OK	c:\winnt\system32\iccvd.dll

[CD-ROM]

Item Value

Drive D:

Description	CD-ROM Drive
Media Loaded	False
Media Type	CD-ROM
Name	LG CD-ROM CRD-8400B
Manufacturer	(Standard CD-ROM drives)
Status	OK
Transfer Rate	Not Available
SCSI Target ID	0
PNP Device ID	IDE\CDROMLG_CD-ROM_CRD-8400B_____1.08____\4249204D393932303830202020202020202020

[Sound Device]

Item Value  
No sound devices

[Display]

Item	Name	S3 Inc. Trio3D
Value	PNP Device ID	PCI\VEN_5333&DEV_8904&SUBSYS_00DB1014&REV_01\3&267A616A&0&20
	Adapter Type	S3 Trio3D, S3 compatible
	Adapter Description	S3 Inc. Trio3D
	Adapter RAM	4.00 MB (4,194,304 bytes)
	Installed Drivers	s3mt3d.sys
	Driver Version	5.01.526.0007
	INF File	s3trio3d.inf (S3Inc section)
	Color Planes	1
	Color Table Entries	65536
	Resolution	800 x 600 x 60 hertz
	Bits/Pixel	16

[Infrared]

Item Value  
No infrared devices

[Input]

[ Following are sub-categories of this main category ]

[Keyboard]

Item	Description	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard
Value	Name	Enhanced (101- or 102-key)
	Layout	00000409
	PNP Device ID	ACPI\PNP0303\4&43B47AD&0
	NumberOfFunctionKeys	12

[Pointing Device]

Item	Hardware Type	PS/2 Compatible Mouse
Value	Number of Buttons	3
	Status	OK
	PNP Device ID	ACPI\PNP0F13\4&43B47AD&0
	Power Management Supported	False
	Double Click Threshold	6
	Handedness	Right Handed Operation

[Modem]

Item Value  
No modems

[Network]

[ Following are sub-categories of this main category ]

[Adapter]

Item Value  
Name [00000000] RAS Async Adapter  
Adapter Type Not Available  
Product Name RAS Async Adapter  
Installed True  
PNP Device ID Not Available  
Last Reset 3/12/2001 6:29:18 AM  
Index 0  
Service Name AsyncMac  
IP Address Not Available  
IP Subnet Not Available  
Default IP Gateway Not Available  
DHCP Enabled False  
DHCP Server Not Available  
DHCP Lease Expires Not Available  
DHCP Lease Obtained Not Available  
MAC Address Not Available  
Service Name Not Available

Name [00000001] WAN Miniport (L2TP)  
Adapter Type Not Available  
Product Name WAN Miniport (L2TP)  
Installed True  
PNP Device ID ROOT\MS\_L2TPMINIPORT\0000  
Last Reset 3/12/2001 6:29:18 AM  
Index 1  
Service Name Rasl2tp  
IP Address Not Available  
IP Subnet Not Available  
Default IP Gateway Not Available  
DHCP Enabled False  
DHCP Server Not Available  
DHCP Lease Expires Not Available  
DHCP Lease Obtained Not Available  
MAC Address Not Available  
Service Name Rasl2tp  
Driver c:\winnt\system32\drivers\rasl2tp.sys (50320, 5.00.2179.1)

Name [00000002] WAN Miniport (PPTP)  
Adapter Type Wide Area Network (WAN)  
Product Name WAN Miniport (PPTP)  
Installed True  
PNP Device ID ROOT\MS\_PPTPMINIPORT\0000  
Last Reset 3/12/2001 6:29:18 AM  
Index 2  
Service Name PptpMiniport  
IP Address Not Available  
IP Subnet Not Available  
Default IP Gateway Not Available  
DHCP Enabled False  
DHCP Server Not Available  
DHCP Lease Expires Not Available  
DHCP Lease Obtained Not Available  
MAC Address 50:50:54:50:30:30  
Service Name PptpMiniport

Driver c:\winnt\system32\drivers\rasptp.sys (47376, 5.00.2160.1)

Name [00000003] Direct Parallel  
Adapter Type Not Available  
Product Name Direct Parallel  
Installed True  
PNP Device ID ROOT\MS\_PTMINIPORT\0000  
Last Reset 3/12/2001 6:29:18 AM  
Index 3  
Service Name Raspti  
IP Address Not Available  
IP Subnet Not Available  
Default IP Gateway Not Available  
DHCP Enabled False  
DHCP Server Not Available  
DHCP Lease Expires Not Available  
DHCP Lease Obtained Not Available  
MAC Address Not Available  
Service Name Raspti  
Driver c:\winnt\system32\drivers\raspti.sys (16880, 5.00.2146.1)

Name [00000004] WAN Miniport (IP)  
Adapter Type Not Available  
Product Name WAN Miniport (IP)  
Installed True  
PNP Device ID ROOT\MS\_NDISWANIP\0000  
Last Reset 3/12/2001 6:29:18 AM  
Index 4  
Service Name NdisWan  
IP Address Not Available  
IP Subnet Not Available  
Default IP Gateway Not Available  
DHCP Enabled False  
DHCP Server Not Available  
DHCP Lease Expires Not Available  
DHCP Lease Obtained Not Available  
MAC Address Not Available  
Service Name NdisWan  
Driver c:\winnt\system32\drivers\ndiswan.sys (89808, 5.00.2184.1)

Name [00000005] IBM 10/100 Ethernet Server Adapter  
Adapter Type Ethernet 802.3  
Product Name IBM 10/100 Ethernet Server Adapter  
Installed True  
PNP Device ID PCI\VEN\_8086&DEV\_1229&SUBSYS\_01F11014&REV\_0C\3&267A616A&0&60  
Last Reset 3/12/2001 6:29:18 AM  
Index 5  
Service Name E100B  
IP Address 192.168.125.201  
IP Subnet 255.255.255.0  
Default IP Gateway Not Available  
DHCP Enabled False  
DHCP Server Not Available  
DHCP Lease Expires Not Available  
DHCP Lease Obtained Not Available  
MAC Address 00:03:47:0A:6A:CA  
Service Name E100B  
IRQ Number 27  
I/O Port 0x21C0-0x21FF  
Driver c:\winnt\system32\drivers\e100bnt5.sys (119056, 5.40.11.0000)

Name [00000006] cLAN Host Adapter  
Adapter Type Ethernet 802.3  
Product Name cLAN Host Adapter

Installed True  
 PNP Device ID  
 PCI\VEN\_135B&DEV\_0001&SUBSYS\_00000000&REV\_00\3&267  
 A616A&0&58  
 Last Reset 3/12/2001 6:29:18 AM  
 Index 6  
 Service Name GNINDIS  
 IP Address 192.168.120.1  
 IP Subnet 255.255.255.0  
 Default IP Gateway Not Available  
 DHCP Enabled False  
 DHCP Server Not Available  
 DHCP Lease Expires Not Available  
 DHCP Lease Obtained Not Available  
 MAC Address 00:90:FA:00:1B:18  
 Service Name GNINDIS  
 IRQ Number 26  
 Driver c:\winnt\system32\drivers\gnindis.sys (22598, 4.1.1)

[Protocol]

Item Value  
 Name MSAFD Tcpi [TCP/IP]  
 ConnectionlessService False  
 GuaranteesDelivery True  
 GuaranteesSequencing True  
 MaximumAddressSize 16 bytes  
 MaximumMessageSize 0 bytes  
 MessageOriented False  
 MinimumAddressSize 16 bytes  
 PseudoStreamOriented False  
 SupportsBroadcasting False  
 SupportsConnectData False  
 SupportsDisconnectData False  
 SupportsEncryption False  
 SupportsExpeditedData True  
 SupportsGracefulClosing True  
 SupportsGuaranteedBandwidth False  
 SupportsMulticasting False  
  
 Name MSAFD Tcpi [UDP/IP]  
 ConnectionlessService True  
 GuaranteesDelivery False  
 GuaranteesSequencing False  
 MaximumAddressSize 16 bytes  
 MaximumMessageSize 65467 bytes  
 MessageOriented True  
 MinimumAddressSize 16 bytes  
 PseudoStreamOriented False  
 SupportsBroadcasting True  
 SupportsConnectData False  
 SupportsDisconnectData False  
 SupportsEncryption False  
 SupportsExpeditedData False  
 SupportsGracefulClosing False  
 SupportsGuaranteedBandwidth False  
 SupportsMulticasting True  
  
 Name RSVP UDP Service Provider  
 ConnectionlessService True  
 GuaranteesDelivery False  
 GuaranteesSequencing False  
 MaximumAddressSize 16 bytes  
 MaximumMessageSize 65467 bytes  
 MessageOriented True  
 MinimumAddressSize 16 bytes  
 PseudoStreamOriented False

SupportsBroadcasting True  
 SupportsConnectData False  
 SupportsDisconnectData False  
 SupportsEncryption True  
 SupportsExpeditedData False  
 SupportsGracefulClosing False  
 SupportsGuaranteedBandwidth False  
 SupportsMulticasting True  
  
 Name RSVP TCP Service Provider  
 ConnectionlessService False  
 GuaranteesDelivery True  
 GuaranteesSequencing True  
 MaximumAddressSize 16 bytes  
 MaximumMessageSize 0 bytes  
 MessageOriented False  
 MinimumAddressSize 16 bytes  
 PseudoStreamOriented False  
 SupportsBroadcasting False  
 SupportsConnectData False  
 SupportsDisconnectData False  
 SupportsEncryption True  
 SupportsExpeditedData True  
 SupportsGracefulClosing True  
 SupportsGuaranteedBandwidth False  
 SupportsMulticasting False  
  
 Name MSAFD NetBIOS  
 [\Device\NetBT\_Tcpi\_{D2A2954E-53B0-4242-8EBA-3AD3DDDA9E08}] SEQUENCE 3  
 ConnectionlessService False  
 GuaranteesDelivery True  
 GuaranteesSequencing True  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize 20 bytes  
 PseudoStreamOriented False  
 SupportsBroadcasting False  
 SupportsConnectData False  
 SupportsDisconnectData False  
 SupportsEncryption False  
 SupportsExpeditedData False  
 SupportsGracefulClosing False  
 SupportsGuaranteedBandwidth False  
 SupportsMulticasting False  
  
 Name MSAFD NetBIOS  
 [\Device\NetBT\_Tcpi\_{D2A2954E-53B0-4242-8EBA-3AD3DDDA9E08}] DATAGRAM 3  
 ConnectionlessService True  
 GuaranteesDelivery False  
 GuaranteesSequencing False  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize 20 bytes  
 PseudoStreamOriented False  
 SupportsBroadcasting True  
 SupportsConnectData False  
 SupportsDisconnectData False  
 SupportsEncryption False  
 SupportsExpeditedData False  
 SupportsGracefulClosing False  
 SupportsGuaranteedBandwidth False  
 SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT\_Tcpip\_{CBD78306-A3A1-4444-B08A-916C948A2B49}] SEQUENCEPACKET 0

ConnectionlessService False

GuaranteesDelivery True

GuaranteesSequencing True

MaximumAddressSize 20 bytes

MaximumMessageSize 64000 bytes

MessageOriented True

MinimumAddressSize 20 bytes

PseudoStreamOriented False

SupportsBroadcasting False

SupportsConnectData False

SupportsDisconnectData False

SupportsEncryption False

SupportsExpeditedData False

SupportsGracefulClosing False

SupportsGuaranteedBandwidth False

SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT\_Tcpip\_{CBD78306-A3A1-4444-B08A-916C948A2B49}] DATAGRAM 0

ConnectionlessService True

GuaranteesDelivery False

GuaranteesSequencing False

MaximumAddressSize 20 bytes

MaximumMessageSize 64000 bytes

MessageOriented True

MinimumAddressSize 20 bytes

PseudoStreamOriented False

SupportsBroadcasting True

SupportsConnectData False

SupportsDisconnectData False

SupportsEncryption False

SupportsExpeditedData False

SupportsGracefulClosing False

SupportsGuaranteedBandwidth False

SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT\_Tcpip\_{6F39C5C3-FA2B-4B96-8C5C-24070D7B7CB2}] SEQUENCEPACKET 1

ConnectionlessService False

GuaranteesDelivery True

GuaranteesSequencing True

MaximumAddressSize 20 bytes

MaximumMessageSize 64000 bytes

MessageOriented True

MinimumAddressSize 20 bytes

PseudoStreamOriented False

SupportsBroadcasting False

SupportsConnectData False

SupportsDisconnectData False

SupportsEncryption False

SupportsExpeditedData False

SupportsGracefulClosing False

SupportsGuaranteedBandwidth False

SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT\_Tcpip\_{6F39C5C3-FA2B-4B96-8C5C-24070D7B7CB2}] DATAGRAM 1

ConnectionlessService True

GuaranteesDelivery False

GuaranteesSequencing False

MaximumAddressSize 20 bytes

MaximumMessageSize 64000 bytes

MessageOriented True

MinimumAddressSize 20 bytes

PseudoStreamOriented False

SupportsBroadcasting True

SupportsConnectData False

SupportsDisconnectData False

SupportsEncryption False

SupportsExpeditedData False

SupportsGracefulClosing False

SupportsGuaranteedBandwidth False

SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT\_Tcpip\_{F409C2C7-7706-4373-94CC-4422E208FE86}] SEQUENCEPACKET 2

ConnectionlessService False

GuaranteesDelivery True

GuaranteesSequencing True

MaximumAddressSize 20 bytes

MaximumMessageSize 64000 bytes

MessageOriented True

MinimumAddressSize 20 bytes

PseudoStreamOriented False

SupportsBroadcasting False

SupportsConnectData False

SupportsDisconnectData False

SupportsEncryption False

SupportsExpeditedData False

SupportsGracefulClosing False

SupportsGuaranteedBandwidth False

SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT\_Tcpip\_{F409C2C7-7706-4373-94CC-4422E208FE86}] DATAGRAM 2

ConnectionlessService True

GuaranteesDelivery False

GuaranteesSequencing False

MaximumAddressSize 20 bytes

MaximumMessageSize 64000 bytes

MessageOriented True

MinimumAddressSize 20 bytes

PseudoStreamOriented False

SupportsBroadcasting True

SupportsConnectData False

SupportsDisconnectData False

SupportsEncryption False

SupportsExpeditedData False

SupportsGracefulClosing False

SupportsGuaranteedBandwidth False

SupportsMulticasting False

[WinSock]

Item	Value
File	c:\winnt\system32\winsock.dll
Version	3.10
Size	2.80 KB (2,864 bytes)
File	c:\winnt\system32\wssock32.dll

Version 5.00.2195.1207  
Size 21.27 KB (21,776 bytes)

[Ports]

[ Following are sub-categories of this main category ]

[Serial]

Item	Value
Name	COM1
Status	OK
PNP Device ID	ACPI\PNP0501\1
Maximum Input Buffer Size	0
Maximum Output Buffer Size	False
Settable Baud Rate	True
Settable Data Bits	True
Settable Flow Control	True
Settable Parity	True
Settable Parity Check	True
Settable Stop Bits	True
Settable RLSD	True
Supports RLSD	True
Supports 16 Bit Mode	False
Supports Special Characters	False
Baud Rate	9600
Bits/Byte	8
Stop Bits	1
Parity	None
Busy	0
Abort Read/Write on Error	0
Binary Mode Enabled-1	0
Continue XMit on XOff	0
CTS Outflow Control	0
Discard NULL Bytes	0
DSR Outflow Control	0
DSR Sensitivity	0
DTR Flow Control Type	Enable
EOF Character	0
Error Replace Character	0
Error Replacement Enabled	0
Event Character	0
Parity Check Enabled	0
RTS Flow Control Type	Enable
XOff Character	19
XOffXMit Threshold	512
XOn Character	17
XOnXMit Threshold	2048
XOnXOff InFlow Control	0
XOnXOff OutFlow Control	0
IRQ Number	4
I/O Port	0x03F8-0x03FF
Driver	c:\winnt\system32\drivers\serial.sys (62448, 5.00.2134.1)

Name	COM2
Status	OK
PNP Device ID	ACPI\PNP0501\2
Maximum Input Buffer Size	0
Maximum Output Buffer Size	False
Settable Baud Rate	True
Settable Data Bits	True
Settable Flow Control	True
Settable Parity	True
Settable Parity Check	True
Settable Stop Bits	True
Settable RLSD	True
Supports RLSD	True

Supports 16 Bit Mode	False
Supports Special Characters	False
Baud Rate	9600
Bits/Byte	8
Stop Bits	1
Parity	None
Busy	0
Abort Read/Write on Error	0
Binary Mode Enabled-1	0
Continue XMit on XOff	0
CTS Outflow Control	0
Discard NULL Bytes	0
DSR Outflow Control	0
DSR Sensitivity	0
DTR Flow Control Type	Enable
EOF Character	0
Error Replace Character	0
Error Replacement Enabled	0
Event Character	0
Parity Check Enabled	0
RTS Flow Control Type	Enable
XOff Character	19
XOffXMit Threshold	512
XOn Character	17
XOnXMit Threshold	2048
XOnXOff InFlow Control	0
XOnXOff OutFlow Control	0
IRQ Number	3
I/O Port	0x02F8-0x02FF
Driver	c:\winnt\system32\drivers\serial.sys (62448, 5.00.2134.1)

[Parallel]

Item	Value
Name	LPT1
PNP Device ID	ACPI\PNP0400\1

[Storage]

[ Following are sub-categories of this main category ]

[Drives]

Item	Value
Drive	A:
Description	3 1/2 Inch Floppy Drive
Drive	C:
Description	Local Fixed Disk
Compressed	False
File System	NTFS
Size	8.47 GB (9,097,424,896 bytes)
Free Space	4.76 GB (5,111,955,456 bytes)
Volume Name	
Volume Serial Number	C8DF6769
Partition	Disk #0, Partition #0
Partition Size	8.47 GB (9,097,428,992 bytes)
Starting Offset	16384 bytes
Drive Description	Disk drive
Drive Manufacturer	(Standard disk drives)
Drive Model	IBM ServeRAID Failover Virtual Disk Device
Drive BytesPerSector	512
Drive MediaLoaded	True
Drive MediaType	Fixed hard disk media
Drive Partitions	1
Drive SCSI Bus	0

Drive SCSILogicalUnit 0  
 Drive SCSIPort 3  
 Drive SCSTargetId 0  
 Drive SectorsPerTrack 32  
 Drive Size 9099542528 bytes  
 Drive TotalCylinders 4339  
 Drive TotalSectors 17772544  
 Drive TotalTracks 555392  
 Drive TracksPerCylinder 128

Drive F:  
 Description Local Fixed Disk  
 Compressed Not Available  
 File System Not Available  
 Size Not Available  
 Free SpaceNot Available  
 Volume Name Not Available  
 Volume Serial Number Not Available  
 Partition Disk #1, Partition #0  
 Partition Size 101.70 GB (109,196,607,488 bytes)  
 Starting Offset 2097152 bytes  
 Drive Description Disk drive  
 Drive Manufacturer (Standard disk drives)  
 Drive Model IBM ServeRAID Failover Virtual Disk Device  
 Drive BytesPerSector 512  
 Drive MediaLoaded True  
 Drive MediaType Fixed hard disk media  
 Drive Partitions 1  
 Drive SCsIBus 0  
 Drive SCSILogicalUnit 0  
 Drive SCSIPort 3  
 Drive SCSTargetId 1  
 Drive SectorsPerTrack 32  
 Drive Size 109200801792 bytes  
 Drive TotalCylinders 52071  
 Drive TotalSectors 213282816  
 Drive TotalTracks 6665088  
 Drive TracksPerCylinder 128

Drive G:  
 Description Local Fixed Disk  
 Compressed Not Available  
 File System Not Available  
 Size Not Available  
 Free SpaceNot Available  
 Volume Name Not Available  
 Volume Serial Number Not Available  
 Partition Disk #2, Partition #0  
 Partition Size 474.59 GB (509,588,997,120 bytes)  
 Starting Offset 8225280 bytes  
 Drive Description Disk drive  
 Drive Manufacturer (Standard disk drives)  
 Drive Model IBM ServeRAID Failover Virtual Disk Device  
 Drive BytesPerSector 512  
 Drive MediaLoaded True  
 Drive MediaType Fixed hard disk media  
 Drive Partitions 3  
 Drive SCsIBus 0  
 Drive SCSILogicalUnit 0  
 Drive SCSIPort 4  
 Drive SCSTargetId 0  
 Drive SectorsPerTrack 63  
 Drive Size 509597222400 bytes  
 Drive TotalCylinders 61955  
 Drive TotalSectors 995307075  
 Drive TotalTracks 15798525  
 Drive TracksPerCylinder 255

Drive H:  
 Description Local Fixed Disk  
 Compressed Not Available  
 File System Not Available  
 Size Not Available  
 Free SpaceNot Available  
 Volume Name Not Available  
 Volume Serial Number Not Available  
 Partition Disk #2, Partition #0  
 Partition Size 474.59 GB (509,588,997,120 bytes)  
 Starting Offset 8225280 bytes  
 Drive Description Disk drive  
 Drive Manufacturer (Standard disk drives)  
 Drive Model IBM ServeRAID Failover Virtual Disk Device  
 Drive BytesPerSector 512  
 Drive MediaLoaded True  
 Drive MediaType Fixed hard disk media  
 Drive Partitions 3  
 Drive SCsIBus 0  
 Drive SCSILogicalUnit 0  
 Drive SCSIPort 4  
 Drive SCSTargetId 0  
 Drive SectorsPerTrack 63  
 Drive Size 509597222400 bytes  
 Drive TotalCylinders 61955  
 Drive TotalSectors 995307075  
 Drive TotalTracks 15798525  
 Drive TracksPerCylinder 255

Drive I:  
 Description Local Fixed Disk  
 Compressed Not Available  
 File System Not Available  
 Size Not Available  
 Free SpaceNot Available  
 Volume Name Not Available  
 Volume Serial Number Not Available  
 Partition Disk #3, Partition #0  
 Partition Size 237.29 GB (254,782,996,480 bytes)  
 Starting Offset 2097152 bytes  
 Drive Description Disk drive  
 Drive Manufacturer (Standard disk drives)  
 Drive Model IBM ServeRAID Failover Virtual Disk Device  
 Drive BytesPerSector 512  
 Drive MediaLoaded True  
 Drive MediaType Fixed hard disk media  
 Drive Partitions 2  
 Drive SCsIBus 0  
 Drive SCSILogicalUnit 0  
 Drive SCSIPort 5  
 Drive SCSTargetId 0  
 Drive SectorsPerTrack 32  
 Drive Size 254787190784 bytes  
 Drive TotalCylinders 121492  
 Drive TotalSectors 497631232  
 Drive TotalTracks 15550976  
 Drive TracksPerCylinder 128

Drive J:  
 Description Local Fixed Disk  
 Compressed Not Available  
 File System Not Available  
 Size Not Available  
 Free SpaceNot Available  
 Volume Name Not Available  
 Volume Serial Number Not Available  
 Partition Disk #3, Partition #0  
 Partition Size 237.29 GB (254,782,996,480 bytes)



Starting Offset 2097152 bytes  
 Drive Description Disk drive  
 Drive Manufacturer (Standard disk drives)  
 Drive Model IBM ServeRAID Failover Virtual Disk Device  
 Drive BytesPerSector 512  
 Drive MediaLoaded True  
 Drive MediaType Fixed hard disk media  
 Drive Partitions 2  
 Drive SCSIBus 0  
 Drive SCSILogicalUnit 0  
 Drive SCSIPort 5  
 Drive SCSTargetId 0  
 Drive SectorsPerTrack 32  
 Drive Size 254787190784 bytes  
 Drive TotalCylinders 121492  
 Drive TotalSectors 497631232  
 Drive TotalTracks 15550976  
 Drive TracksPerCylinder 128

Drive K:  
 Description Local Fixed Disk  
 Compressed Not Available  
 File System Not Available  
 Size Not Available  
 Free SpaceNot Available  
 Volume Name Not Available  
 Volume Serial Number Not Available  
 Partition Disk #4, Partition #0  
 Partition Size 237.27 GB (254,769,822,720 bytes)  
 Starting Offset 8225280 bytes  
 Drive Description Disk drive  
 Drive Manufacturer (Standard disk drives)  
 Drive Model IBM ServeRAID Failover Virtual Disk Device  
 Drive BytesPerSector 512  
 Drive MediaLoaded True  
 Drive MediaType Fixed hard disk media  
 Drive Partitions 3  
 Drive SCSIBus 0  
 Drive SCSILogicalUnit 0  
 Drive SCSIPort 6  
 Drive SCSTargetId 0  
 Drive SectorsPerTrack 63  
 Drive Size 254786273280 bytes  
 Drive TotalCylinders 30976  
 Drive TotalSectors 497629440  
 Drive TotalTracks 7898880  
 Drive TracksPerCylinder 255

Drive L:  
 Description Local Fixed Disk  
 Compressed Not Available  
 File System Not Available  
 Size Not Available  
 Free SpaceNot Available  
 Volume Name Not Available  
 Volume Serial Number Not Available  
 Partition Disk #4, Partition #0  
 Partition Size 237.27 GB (254,769,822,720 bytes)  
 Starting Offset 8225280 bytes  
 Drive Description Disk drive  
 Drive Manufacturer (Standard disk drives)  
 Drive Model IBM ServeRAID Failover Virtual Disk Device  
 Drive BytesPerSector 512  
 Drive MediaLoaded True  
 Drive MediaType Fixed hard disk media  
 Drive Partitions 3  
 Drive SCSIBus 0  
 Drive SCSILogicalUnit 0

Drive SCSIPort 6  
 Drive SCSTargetId 0  
 Drive SectorsPerTrack 63  
 Drive Size 254786273280 bytes  
 Drive TotalCylinders 30976  
 Drive TotalSectors 497629440  
 Drive TotalTracks 7898880  
 Drive TracksPerCylinder 255

Drive M:  
 Description Local Fixed Disk  
 Compressed Not Available  
 File System Not Available  
 Size Not Available  
 Free SpaceNot Available  
 Volume Name Not Available  
 Volume Serial Number Not Available  
 Partition Disk #5, Partition #0  
 Partition Size 237.29 GB (254,782,996,480 bytes)  
 Starting Offset 2097152 bytes  
 Drive Description Disk drive  
 Drive Manufacturer (Standard disk drives)  
 Drive Model IBM ServeRAID Failover Virtual Disk Device  
 Drive BytesPerSector 512  
 Drive MediaLoaded True  
 Drive MediaType Fixed hard disk media  
 Drive Partitions 3  
 Drive SCSIBus 0  
 Drive SCSILogicalUnit 0  
 Drive SCSIPort 7  
 Drive SCSTargetId 0  
 Drive SectorsPerTrack 32  
 Drive Size 254787190784 bytes  
 Drive TotalCylinders 121492  
 Drive TotalSectors 497631232  
 Drive TotalTracks 15550976  
 Drive TracksPerCylinder 128

Drive N:  
 Description Local Fixed Disk  
 Compressed Not Available  
 File System Not Available  
 Size Not Available  
 Free SpaceNot Available  
 Volume Name Not Available  
 Volume Serial Number Not Available  
 Partition Disk #5, Partition #0  
 Partition Size 237.29 GB (254,782,996,480 bytes)  
 Starting Offset 2097152 bytes  
 Drive Description Disk drive  
 Drive Manufacturer (Standard disk drives)  
 Drive Model IBM ServeRAID Failover Virtual Disk Device  
 Drive BytesPerSector 512  
 Drive MediaLoaded True  
 Drive MediaType Fixed hard disk media  
 Drive Partitions 3  
 Drive SCSIBus 0  
 Drive SCSILogicalUnit 0  
 Drive SCSIPort 7  
 Drive SCSTargetId 0  
 Drive SectorsPerTrack 32  
 Drive Size 254787190784 bytes  
 Drive TotalCylinders 121492  
 Drive TotalSectors 497631232  
 Drive TotalTracks 15550976  
 Drive TracksPerCylinder 128

Drive O:

Description Local Fixed Disk  
 Compressed Not Available  
 File System Not Available  
 Size Not Available  
 Free Space Not Available  
 Volume Name Not Available  
 Volume Serial Number Not Available  
 Partition Disk #6, Partition #0  
 Partition Size 237.29 GB (254,782,996,480 bytes)  
 Starting Offset 2097152 bytes  
 Drive Description Disk drive  
 Drive Manufacturer (Standard disk drives)  
 Drive Model IBM ServeRAID Failover Virtual Disk Device  
 Drive BytesPerSector 512  
 Drive MediaLoaded True  
 Drive MediaType Fixed hard disk media  
 Drive Partitions 3  
 Drive SCSI Bus 0  
 Drive SCSI Logical Unit 0  
 Drive SCSI Port 8  
 Drive SCSI Target ID 0  
 Drive SectorsPerTrack 32  
 Drive Size 254787190784 bytes  
 Drive TotalCylinders 121492  
 Drive TotalSectors 497631232  
 Drive TotalTracks 15550976  
 Drive TracksPerCylinder 128

Drive P:  
 Description Local Fixed Disk  
 Compressed Not Available  
 File System Not Available  
 Size Not Available  
 Free Space Not Available  
 Volume Name Not Available  
 Volume Serial Number Not Available  
 Partition Disk #6, Partition #0  
 Partition Size 237.29 GB (254,782,996,480 bytes)  
 Starting Offset 2097152 bytes  
 Drive Description Disk drive  
 Drive Manufacturer (Standard disk drives)  
 Drive Model IBM ServeRAID Failover Virtual Disk Device  
 Drive BytesPerSector 512  
 Drive MediaLoaded True  
 Drive MediaType Fixed hard disk media  
 Drive Partitions 3  
 Drive SCSI Bus 0  
 Drive SCSI Logical Unit 0  
 Drive SCSI Port 8  
 Drive SCSI Target ID 0  
 Drive SectorsPerTrack 32  
 Drive Size 254787190784 bytes  
 Drive TotalCylinders 121492  
 Drive TotalSectors 497631232  
 Drive TotalTracks 15550976  
 Drive TracksPerCylinder 128

Drive Q:  
 Description Local Fixed Disk  
 Compressed Not Available  
 File System Not Available  
 Size Not Available  
 Free Space Not Available  
 Volume Name Not Available  
 Volume Serial Number Not Available  
 Partition Disk #7, Partition #0  
 Partition Size 237.29 GB (254,782,996,480 bytes)  
 Starting Offset 2097152 bytes

Drive Description Disk drive  
 Drive Manufacturer (Standard disk drives)  
 Drive Model IBM ServeRAID Failover Virtual Disk Device  
 Drive BytesPerSector 512  
 Drive MediaLoaded True  
 Drive MediaType Fixed hard disk media  
 Drive Partitions 3  
 Drive SCSI Bus 0  
 Drive SCSI Logical Unit 0  
 Drive SCSI Port 9  
 Drive SCSI Target ID 0  
 Drive SectorsPerTrack 32  
 Drive Size 254787190784 bytes  
 Drive TotalCylinders 121492  
 Drive TotalSectors 497631232  
 Drive TotalTracks 15550976  
 Drive TracksPerCylinder 128

Drive R:  
 Description Local Fixed Disk  
 Compressed Not Available  
 File System Not Available  
 Size Not Available  
 Free Space Not Available  
 Volume Name Not Available  
 Volume Serial Number Not Available  
 Partition Disk #7, Partition #0  
 Partition Size 237.29 GB (254,782,996,480 bytes)  
 Starting Offset 2097152 bytes  
 Drive Description Disk drive  
 Drive Manufacturer (Standard disk drives)  
 Drive Model IBM ServeRAID Failover Virtual Disk Device  
 Drive BytesPerSector 512  
 Drive MediaLoaded True  
 Drive MediaType Fixed hard disk media  
 Drive Partitions 3  
 Drive SCSI Bus 0  
 Drive SCSI Logical Unit 0  
 Drive SCSI Port 9  
 Drive SCSI Target ID 0  
 Drive SectorsPerTrack 32  
 Drive Size 254787190784 bytes  
 Drive TotalCylinders 121492  
 Drive TotalSectors 497631232  
 Drive TotalTracks 15550976  
 Drive TracksPerCylinder 128

Drive S:  
 Description Local Fixed Disk  
 Compressed False  
 File System NTFS  
 Size 117.19 GB (125,829,099,520 bytes)  
 Free Space 24.04 GB (25,817,296,896 bytes)  
 Volume Name backup2\_1&2  
 Volume Serial Number 84B7B1CA  
 Partition Disk #6, Partition #0  
 Partition Size 237.29 GB (254,782,996,480 bytes)  
 Starting Offset 2097152 bytes  
 Drive Description Disk drive  
 Drive Manufacturer (Standard disk drives)  
 Drive Model IBM ServeRAID Failover Virtual Disk Device  
 Drive BytesPerSector 512  
 Drive MediaLoaded True  
 Drive MediaType Fixed hard disk media  
 Drive Partitions 3  
 Drive SCSI Bus 0  
 Drive SCSI Logical Unit 0  
 Drive SCSI Port 8

Drive SCSITargetId 0  
 Drive SectorsPerTrack 32  
 Drive Size 254787190784 bytes  
 Drive TotalCylinders 121492  
 Drive TotalSectors 497631232  
 Drive TotalTracks 15550976  
 Drive TracksPerCylinder 128

Drive T:  
 Description Local Fixed Disk  
 Compressed False  
 File System NTFS  
 Size 117.19 GB (125,829,099,520 bytes)  
 Free Space 24.04 GB (25,817,362,432 bytes)  
 Volume Name backup2\_3&4  
 Volume Serial Number ACCCCC87  
 Partition Disk #7, Partition #0  
 Partition Size 237.29 GB (254,782,996,480 bytes)  
 Starting Offset 2097152 bytes  
 Drive Description Disk drive  
 Drive Manufacturer (Standard disk drives)  
 Drive Model IBM ServeRAID Failover Virtual Disk Device  
 Drive BytesPerSector 512  
 Drive MediaLoaded True  
 Drive MediaType Fixed hard disk media  
 Drive Partitions 3  
 Drive SCSIBus 0  
 Drive SCSILogicalUnit 0  
 Drive SCSIPort 9  
 Drive SCSITargetId 0  
 Drive SectorsPerTrack 32  
 Drive Size 254787190784 bytes  
 Drive TotalCylinders 121492  
 Drive TotalSectors 497631232  
 Drive TotalTracks 15550976  
 Drive TracksPerCylinder 128

Drive U:  
 Description Local Fixed Disk  
 Compressed False  
 File System NTFS  
 Size 117.19 GB (125,830,299,648 bytes)  
 Free Space 24.02 GB (25,792,114,688 bytes)  
 Volume Name backup1\_1&2  
 Volume Serial Number 088D0CA3  
 Partition Disk #4, Partition #0  
 Partition Size 237.27 GB (254,769,822,720 bytes)  
 Starting Offset 8225280 bytes  
 Drive Description Disk drive  
 Drive Manufacturer (Standard disk drives)  
 Drive Model IBM ServeRAID Failover Virtual Disk Device  
 Drive BytesPerSector 512  
 Drive MediaLoaded True  
 Drive MediaType Fixed hard disk media  
 Drive Partitions 3  
 Drive SCSIBus 0  
 Drive SCSILogicalUnit 0  
 Drive SCSIPort 6  
 Drive SCSITargetId 0  
 Drive SectorsPerTrack 63  
 Drive Size 254786273280 bytes  
 Drive TotalCylinders 30976  
 Drive TotalSectors 497629440  
 Drive TotalTracks 7898880  
 Drive TracksPerCylinder 255

Drive V:  
 Description Local Fixed Disk

Compressed False  
 File System NTFS  
 Size 117.19 GB (125,829,099,520 bytes)  
 Free Space 24.04 GB (25,816,969,216 bytes)  
 Volume Name backup1\_3&4  
 Volume Serial Number E8A224E6  
 Partition Disk #5, Partition #0  
 Partition Size 237.29 GB (254,782,996,480 bytes)  
 Starting Offset 2097152 bytes  
 Drive Description Disk drive  
 Drive Manufacturer (Standard disk drives)  
 Drive Model IBM ServeRAID Failover Virtual Disk Device  
 Drive BytesPerSector 512  
 Drive MediaLoaded True  
 Drive MediaType Fixed hard disk media  
 Drive Partitions 3  
 Drive SCSIBus 0  
 Drive SCSILogicalUnit 0  
 Drive SCSIPort 7  
 Drive SCSITargetId 0  
 Drive SectorsPerTrack 32  
 Drive Size 254787190784 bytes  
 Drive TotalCylinders 121492  
 Drive TotalSectors 497631232  
 Drive TotalTracks 15550976  
 Drive TracksPerCylinder 128

Drive W:  
 Description Local Fixed Disk  
 Compressed False  
 File System NTFS  
 Size 996.19 MB (1,044,577,280 bytes)  
 Free Space 476.63 MB (499,786,752 bytes)  
 Volume Name MSDTC  
 Volume Serial Number 60EEFCD6  
 Partition Disk #2, Partition #0  
 Partition Size 474.59 GB (509,588,997,120 bytes)  
 Starting Offset 8225280 bytes  
 Drive Description Disk drive  
 Drive Manufacturer (Standard disk drives)  
 Drive Model IBM ServeRAID Failover Virtual Disk Device  
 Drive BytesPerSector 512  
 Drive MediaLoaded True  
 Drive MediaType Fixed hard disk media  
 Drive Partitions 3  
 Drive SCSIBus 0  
 Drive SCSILogicalUnit 0  
 Drive SCSIPort 4  
 Drive SCSITargetId 0  
 Drive SectorsPerTrack 63  
 Drive Size 509597222400 bytes  
 Drive TotalCylinders 61955  
 Drive TotalSectors 995307075  
 Drive TotalTracks 15798525  
 Drive TracksPerCylinder 255

Drive E:  
 Description Network Connection  
 Provider Name \\192.168.125.251\e\$\shogun

[SCSI]

Item	Value
Name	IBM ServeRAID 4H Controller
Caption	IBM ServeRAID 4H Controller
Driver	ipsraidn
Status	OK

PNP Device ID  
 PCI\VEN\_1014&DEV\_002E&SUBSYS\_022E1014&REV\_10\3&267  
 A616A&0&50  
 Device ID  
 PCI\VEN\_1014&DEV\_002E&SUBSYS\_022E1014&REV\_10\3&267  
 A616A&0&50  
 Device Map Not Available  
 Index Not Available  
 Max Number Controlled Not Available  
 IRQ Number 25  
 I/O Port 0x2200-0x22FF  
 Driver c:\winnt\system32\drivers\ipsraidn.sys (102807, 4.70.08)

Name Adaptec AIC-7896/AIC-7897 PCI Ultra2 SCSI Controller  
 Caption Adaptec AIC-7896/AIC-7897 PCI Ultra2 SCSI Controller  
 Driver aic78u2  
 Status OK

PNP Device ID  
 PCI\VEN\_9005&DEV\_005F&SUBSYS\_080F9005&REV\_00\3&267  
 A616A&0&70  
 Device ID  
 PCI\VEN\_9005&DEV\_005F&SUBSYS\_080F9005&REV\_00\3&267  
 A616A&0&70  
 Device Map Not Available  
 Index Not Available  
 Max Number Controlled Not Available  
 IRQ Number 51  
 I/O Port 0x2300-0x23FF  
 Driver c:\winnt\system32\drivers\aic78u2.sys (65168, v3.00a)

Name Adaptec AIC-7896/AIC-7897 PCI Ultra2 SCSI Controller  
 Caption Adaptec AIC-7896/AIC-7897 PCI Ultra2 SCSI Controller  
 Driver aic78u2  
 Status OK

PNP Device ID  
 PCI\VEN\_9005&DEV\_005F&SUBSYS\_080F9005&REV\_00\3&267  
 A616A&0&71  
 Device ID  
 PCI\VEN\_9005&DEV\_005F&SUBSYS\_080F9005&REV\_00\3&267  
 A616A&0&71  
 Device Map Not Available  
 Index Not Available  
 Max Number Controlled Not Available  
 IRQ Number 51  
 I/O Port 0x2400-0x24FF  
 Driver c:\winnt\system32\drivers\aic78u2.sys (65168, v3.00a)

Name IBM ServeRAID 4H Controller  
 Caption IBM ServeRAID 4H Controller  
 Driver ipsraidn  
 Status OK  
 PNP Device ID  
 PCI\VEN\_1014&DEV\_002E&SUBSYS\_022E1014&REV\_10\3&172  
 E68DD&0&20  
 Device ID  
 PCI\VEN\_1014&DEV\_002E&SUBSYS\_022E1014&REV\_10\3&172  
 E68DD&0&20  
 Device Map Not Available  
 Index Not Available  
 Max Number Controlled Not Available  
 IRQ Number 19  
 I/O Port 0x6000-0xAFFF  
 Driver c:\winnt\system32\drivers\ipsraidn.sys (102807, 4.70.08)

Name IBM ServeRAID 4H Controller  
 Caption IBM ServeRAID 4H Controller  
 Driver ipsraidn  
 Status OK

PNP Device ID  
 PCI\VEN\_1014&DEV\_002E&SUBSYS\_022E1014&REV\_10\3&172  
 E68DD&0&28  
 Device ID  
 PCI\VEN\_1014&DEV\_002E&SUBSYS\_022E1014&REV\_10\3&172  
 E68DD&0&28  
 Device Map Not Available  
 Index Not Available  
 Max Number Controlled Not Available  
 IRQ Number 20  
 I/O Port 0x6100-0x61FF  
 Driver c:\winnt\system32\drivers\ipsraidn.sys (102807, 4.70.08)

Name IBM ServeRAID 4H Controller  
 Caption IBM ServeRAID 4H Controller  
 Driver ipsraidn  
 Status OK

PNP Device ID  
 PCI\VEN\_1014&DEV\_002E&SUBSYS\_022E1014&REV\_10\3&23C  
 0707C&0&40  
 Device ID  
 PCI\VEN\_1014&DEV\_002E&SUBSYS\_022E1014&REV\_10\3&23C  
 0707C&0&40  
 Device Map Not Available  
 Index Not Available  
 Max Number Controlled Not Available  
 IRQ Number 23  
 I/O Port 0xB000-0xCFFF  
 Driver c:\winnt\system32\drivers\ipsraidn.sys (102807, 4.70.08)

Name IBM ServeRAID 4H Controller  
 Caption IBM ServeRAID 4H Controller  
 Driver ipsraidn  
 Status OK

PNP Device ID  
 PCI\VEN\_1014&DEV\_002E&SUBSYS\_022E1014&REV\_10\3&23C  
 0707C&0&48  
 Device ID  
 PCI\VEN\_1014&DEV\_002E&SUBSYS\_022E1014&REV\_10\3&23C  
 0707C&0&48  
 Device Map Not Available  
 Index Not Available  
 Max Number Controlled Not Available  
 IRQ Number 24  
 I/O Port 0xB100-0xB1FF  
 Driver c:\winnt\system32\drivers\ipsraidn.sys (102807, 4.70.08)

Name IBM ServeRAID 4H Controller  
 Caption IBM ServeRAID 4H Controller  
 Driver ipsraidn  
 Status OK  
 PNP Device ID  
 PCI\VEN\_1014&DEV\_002E&SUBSYS\_022E1014&REV\_10\3&146  
 CA173&0&30  
 Device ID  
 PCI\VEN\_1014&DEV\_002E&SUBSYS\_022E1014&REV\_10\3&146  
 CA173&0&30  
 Device Map Not Available  
 Index Not Available  
 Max Number Controlled Not Available  
 IRQ Number 21  
 I/O Port 0xD000-0xFFFF  
 Driver c:\winnt\system32\drivers\ipsraidn.sys (102807, 4.70.08)

Name IBM ServeRAID 4H Controller  
 Caption IBM ServeRAID 4H Controller  
 Driver ipsraidn  
 Status OK

PNP Device ID  
 PCI\VEN\_1014&DEV\_002E&SUBSYS\_022E1014&REV\_10\3&146  
 CA173&0&38  
 Device ID  
 PCI\VEN\_1014&DEV\_002E&SUBSYS\_022E1014&REV\_10\3&146  
 CA173&0&38  
 Device Map Not Available  
 Index Not Available  
 Max Number Controlled Not Available  
 IRQ Number 22  
 I/O Port 0xD100-0xD1FF  
 Driver c:\winnt\system32\drivers\ipsraidn.sys (102807, 4.70.08)

Name IBM ServeRAID Failover Virtual Adapter  
 Caption IBM ServeRAID Failover Virtual Adapter  
 Driver twintail  
 Status OK

PNP Device ID ROOT\TWINTAIL\0000  
 Device ID ROOT\TWINTAIL\0000  
 Device Map Not Available  
 Index Not Available  
 Max Number Controlled Not Available  
 Driver c:\winnt\system32\drivers\twintail.sys (26032, 4.70.06)

[Printing]

Name Port Name Server Name  
 No printing information

[Problem Devices]

Device	PNP Device ID	Error Code
Other PCI Bridge Device		
PCI\VEN_1014&DEV_00DC&SUBSYS_00DC1014&REV_02\3&26 7A616A&0&10	28	
Not Available	ACPI\IBM37D0\4&43B47AD&0	28

[USB]

Device	PNP Device ID
Intel 82371AB/EB PCI to USB Universal Host Controller	
PCI\VEN_8086&DEV_7112&SUBSYS_00000000&REV_01\3&267 A616A&0&7A	
USB Root Hub	USB\ROOT_HUB\4&B5B4E1B&0

[Software Environment]

[ Following are sub-categories of this main category ]

[Drivers]

Name	Description	File	Type	Started
Start Mode	State	Status	Error Control	
Accept Pause	Accept Stop			
abiosdsk	Abiosdsk	Not Available	Kernel Driver	
False	Disabled	Stopped OK	Ignore	False
False				
acpi	Microsoft ACPI Driver		Kernel Driver	
c:\winnt\system32\drivers\acpi.sys				
True	Boot	Running OK	Normal	False
True				
acpiec	ACPIEC	c:\winnt\system32\drivers\acpiec.sys	Kernel Driver	
False	False	Disabled	Stopped OK	
Normal	False	False		
adpu160m	adpu160m	Not Available	Kernel Driver	
False	Disabled	Stopped OK	Normal	False
False				

afd	AFD Networking Support Environment				
c:\winnt\system32\drivers\afd.sys	Kernel Driver	True			
Auto	Running OK	Normal	False	True	
aic116x	aic116x	Not Available	Kernel Driver		
False	Disabled	Stopped OK	Normal	False	
False					
aic78u2	aic78u2	c:\winnt\system32\drivers\aic78u2.sys	Kernel Driver		
True	Boot	Running OK			
Normal	False	True			
aic78xx	aic78xx	Not Available	Kernel Driver		
False	Disabled	Stopped OK	Normal	False	
False					
ami0nt	ami0nt	Not Available	Kernel Driver		
False	Disabled	Stopped OK	Normal	False	
False					
asc	asc	Not Available	Kernel Driver		
False	Disabled	Stopped OK	Normal	False	
False					
asc3550	asc3550	Not Available	Kernel Driver		
False	Disabled	Stopped OK	Normal	False	
False					
asynmac	RAS Asynchronous Media Driver		Kernel Driver		
c:\winnt\system32\drivers\asynmac.sys					
False	Manual	Stopped OK	Normal	False	
False					
atapi	Standard IDE/ESDI Hard Disk Controller		Kernel Driver		
c:\winnt\system32\drivers\atapi.sys					
True	Boot	Running OK	Normal	False	
True					
atdisk	Atdisk	Not Available	Kernel Driver		
False	Disabled	Stopped OK	Ignore	False	
False					
atmarpc	ATM ARP Client Protocol		Kernel Driver		
c:\winnt\system32\drivers\atmarpc.sys					
False	Manual	Stopped OK	Normal	False	
False					
audstub	Audio Stub Driver		Kernel Driver		
c:\winnt\system32\drivers\audstub.sys					
True	Manual	Running OK	Normal	False	
True					
beep	Beep	c:\winnt\system32\drivers\beep.sys	Kernel Driver		
True	System	Running OK			
Normal	False	True			
cdaudio	Cdaudio	c:\winnt\system32\drivers\cdaudio.sys	Kernel Driver		
False	System	Stopped OK			
Ignore	False	False			
cdfs	Cdfs	c:\winnt\system32\drivers\cdfs.sys	System Driver		File
True	Disabled	Running OK			
Normal	False	True			
cdrom	CD-ROM Driver	c:\winnt\system32\drivers\cdrom.sys	Kernel Driver		
True	System	Running OK			
Normal	False	True			
changer	Changer	Not Available	Kernel Driver		
False	System	Stopped OK	Ignore	False	
False					
cpqarray	Cpqarray	Not Available	Kernel Driver		
False	Disabled	Stopped OK	Normal	False	
False					
cpqarray2	cpqarray2	Not Available	Kernel Driver		
False	Disabled	Stopped OK	Normal	False	
False					
cpqfcalm	cpqfcalm	Not Available	Kernel Driver		
False	Disabled	Stopped OK	Normal	False	
False					
dac960nt	dac960nt	Not Available	Kernel Driver		
False	Disabled	Stopped OK	Normal	False	
False					

deckzpsx	deckzpsx	Not Available	Kernel Driver	False	Manual	Stopped	OK	Normal	False
False	Disabled	Stopped	OK	Normal	False				
False									
dfsdriver	DfsDriver	c:\winnt\system32\drivers\dfs.sys	File System Driver	Kernel Driver	False	Manual	Stopped	OK	
True	Boot	Running	OK	Normal					
False	True								
disk	Disk Driver	c:\winnt\system32\drivers\disk.sys	Kernel Driver	Kernel Driver	False	Manual	Stopped	OK	Normal
True	Boot	Running	OK	Normal	False				False
Normal	False	True							
diskperf	Diskperf	c:\winnt\system32\drivers\diskperf.sys	Kernel Driver	Kernel Driver	False	Manual	Running	OK	
Normal	False	False	Disabled	Stopped	OK				
dmboot	dmboot	c:\winnt\system32\drivers\dmboot.sys	Kernel Driver	Kernel Driver	True	Manual	Running	OK	
Normal	False	False	Disabled	Stopped	OK				
dmio	Logical Disk Manager Driver	c:\winnt\system32\drivers\dmio.sys	Kernel Driver	Kernel Driver	True	Manual	Running	OK	Normal
True	Boot	Running	OK	Normal	False				False
True									
dmload	dmload	c:\winnt\system32\drivers\dmload.sys	Kernel Driver	Kernel Driver	True	Manual	Running	OK	
Normal	False	True	Boot	Running	OK				
e100b	Intel(R) PRO Adapter Driver	c:\winnt\system32\drivers\e100bnt5.sys	Kernel Driver	Kernel Driver	True	Manual	Running	OK	Normal
True	Manual	Running	OK	Normal	False				False
True									
efs	EFS	c:\winnt\system32\drivers\efs.sys	File System Driver	Kernel Driver	True	Manual	Running	OK	Normal
False	True	Disabled	Running	OK	Normal				
fastfat	Fastfat	c:\winnt\system32\drivers\fastfat.sys	File System Driver	Kernel Driver	True	Manual	Running	OK	
Normal	False	True	Disabled	Running	OK				
fdc	Floppy Disk Controller Driver	c:\winnt\system32\drivers\fdc.sys	Kernel Driver	Kernel Driver	True	Manual	Running	OK	Normal
Manual	Running	OK	Normal	False	True				False
fireport	fireport	Not Available	Kernel Driver	Kernel Driver	False	Manual	Stopped	OK	Normal
False	Disabled	Stopped	OK	Normal	False				False
False									
flpydisk	Floppy Disk Driver	c:\winnt\system32\drivers\flpydisk.sys	Kernel Driver	Kernel Driver	True	Manual	Running	OK	Normal
True	Manual	Running	OK	Normal	False				False
True									
ftdisk	Volume Manager Driver	c:\winnt\system32\drivers\ftdisk.sys	Kernel Driver	Kernel Driver	True	Manual	Running	OK	Normal
True	Boot	Running	OK	Normal	False				False
True									
gnindis	cLAN NDIS Driver	c:\winnt\system32\drivers\gnindis.sys	Kernel Driver	Kernel Driver	True	Manual	Running	OK	Normal
Normal	False	True	Auto	Running	OK				
gnivia	cLAN VIA Driver	c:\winnt\system32\drivers\gnivia.sys	Kernel Driver	Kernel Driver	True	Manual	Running	OK	Normal
Normal	False	True	Auto	Running	OK				
gpc	Generic Packet Classifier	c:\winnt\system32\drivers\msgpc.sys	Kernel Driver	Kernel Driver	True	Manual	Running	OK	Normal
True	Manual	Running	OK	Normal	False				False
True									
i8042prt	i8042 Keyboard and PS/2 Mouse Port Driver	c:\winnt\system32\drivers\i8042prt.sys	Kernel Driver	Kernel Driver	True	Manual	Running	OK	Normal
True	System	Running	OK	Normal	False				False
True									
ini910u	ini910u	Not Available	Kernel Driver	Kernel Driver	False	Manual	Stopped	OK	Normal
False	Disabled	Stopped	OK	Normal	False				False
False									
intelide	IntelIde	c:\winnt\system32\drivers\intelide.sys	Kernel Driver	Kernel Driver	True	Manual	Running	OK	Normal
Normal	False	True	Boot	Running	OK				False
True									
ipfilterdriver	IP Traffic Filter Driver	c:\winnt\system32\drivers\ipfltdrv.sys	Kernel Driver	Kernel Driver	True	Manual	Running	OK	Normal
True									False
False									
ipinip	IP in IP Tunnel Driver	c:\winnt\system32\drivers\ipinip.sys	Kernel Driver	Kernel Driver	True	Manual	Running	OK	Normal
Normal	False	False	Manual	Stopped	OK				
ipnat	IP Network Address Translator	c:\winnt\system32\drivers\ipnat.sys	Kernel Driver	Kernel Driver	True	Manual	Running	OK	Normal
False	Manual	Stopped	OK	Normal	False				False
False									
ipsecc	IPSEC driver	c:\winnt\system32\drivers\ipsecc.sys	Kernel Driver	Kernel Driver	True	Manual	Running	OK	Normal
Normal	False	True	Manual	Running	OK				
ipsperf	IBM ServeRAID Performance Driver	c:\winnt\system32\drivers\ipsperf.sys	Kernel Driver	Kernel Driver	True	Manual	Running	OK	Normal
True	Boot	Running	OK	Normal	False				False
True									
ipsraidn	IBM ServeRAID Device Driver	c:\winnt\system32\drivers\ipsraidn.sys	Kernel Driver	Kernel Driver	True	Manual	Running	OK	Normal
True	Boot	Running	OK	Normal	False				False
True									
isapnp	PnP ISA/EISA Bus Driver	c:\winnt\system32\drivers\isapnp.sys	Kernel Driver	Kernel Driver	True	Manual	Running	OK	Critical
True	Boot	Running	OK	Critical	False				False
True									
kbdclass	Keyboard Class Driver	c:\winnt\system32\drivers\kbdclass.sys	Kernel Driver	Kernel Driver	True	Manual	Running	OK	Normal
True	System	Running	OK	Normal	False				False
True									
ksecdd	KSecDD	c:\winnt\system32\drivers\ksecdd.sys	Kernel Driver	Kernel Driver	True	Manual	Running	OK	Normal
Normal	False	True	Boot	Running	OK				
lbrtfdc	lbrtfdc	Not Available	Kernel Driver	Kernel Driver	False	Manual	Stopped	OK	Ignore
False	System	Stopped	OK	Ignore	False				False
False									
lp6nds35	lp6nds35	Not Available	Kernel Driver	Kernel Driver	False	Manual	Stopped	OK	Normal
False	Disabled	Stopped	OK	Normal	False				False
False									
mmdd	mmdd	c:\winnt\system32\drivers\mmdd.sys	Kernel Driver	Kernel Driver	True	Manual	Running	OK	Normal
Ignore	False	True	System	Running	OK				
modem	Modem	c:\winnt\system32\drivers\modem.sys	Kernel Driver	Kernel Driver	True	Manual	Running	OK	Normal
Ignore	False	False	Manual	Stopped	OK				False
mouclass	Mouse Class Driver	c:\winnt\system32\drivers\mouclass.sys	Kernel Driver	Kernel Driver	True	Manual	Running	OK	Normal
True	System	Running	OK	Normal	False				False
True									
mountmgr	MountMgr	c:\winnt\system32\drivers\mountmgr.sys	Kernel Driver	Kernel Driver	True	Manual	Running	OK	Normal
Normal	False	True	Boot	Running	OK				
mraid35x	mraid35x	Not Available	Kernel Driver	Kernel Driver	False	Manual	Stopped	OK	Normal
False	Disabled	Stopped	OK	Normal	False				False
False									
mrxsmb	MRXSMB	c:\winnt\system32\drivers\mrxsmb.sys	System Driver	System Driver	True	Manual	Running	OK	Normal
Normal	False	True	System	Running	OK				False
msfs	Msfs	c:\winnt\system32\drivers\msfs.sys	System Driver	System Driver	True	Manual	Running	OK	Normal
Normal	False	True	System	Running	OK				False
mskssrv	Microsoft Streaming Service Proxy	c:\winnt\system32\drivers\mskssrv.sys	Kernel Driver	Kernel Driver	False	Manual	Running	OK	Normal
False	Manual	Running	OK	Normal	False				False
False									
mspclock	Microsoft Streaming Clock Proxy	c:\winnt\system32\drivers\mspclock.sys	Kernel Driver	Kernel Driver	False	Manual	Running	OK	Normal
False	Manual	Running	OK	Normal	False				False
False									



```

True      Manual  Running  OK      Normal  False
True
serial    Serial port driver      c:\winnt\system32\drivers\serial.sys
Kernel Driver      True      System  Running  OK
Ignore  False      True
sfloppy  Sfloppy  c:\winnt\system32\drivers\sfloppy.sys
Kernel Driver      False     System  Stopped  OK
Ignore  False     False
sglfb    sglfb    Not Available      Kernel Driver
False    System  Stopped  OK      Normal  False
False
simbad   Simbad   Not Available      Kernel Driver
False    Disabled Stopped  OK      Normal  False
False
srv      Srv      c:\winnt\system32\drivers\srvsrv.sys File System
Driver    True     Manual  Running  OK      Normal
False     True
swenum   Software Bus Driver
c:\winnt\system32\drivers\swenum.sys      Kernel Driver
True     Manual  Running  OK      Normal  False
True
symc810  symc810  Not Available      Kernel Driver
False    Disabled Stopped  OK      Normal  False
False
symc8xx  symc8xx  Not Available      Kernel Driver
False    Disabled Stopped  OK      Normal  False
False
sym_hi   sym_hi   Not Available      Kernel Driver
False    Disabled Stopped  OK      Normal  False
False
tcpip    TCP/IP Protocol Driver
c:\winnt\system32\drivers\tcpip.sys      Kernel Driver
True     System  Running  OK      Normal  False
True
tdasync  TDASYNC      c:\winnt\system32\drivers\tdasync.sys
Kernel Driver      False     Manual  Stopped  OK
Ignore  False     False
tdipx    TDIPX      c:\winnt\system32\drivers\tdipx.sys
Kernel Driver      False     Manual  Stopped  OK
Ignore  False     False
tdnetb   TDNETB      c:\winnt\system32\drivers\tdnetb.sys
Kernel Driver      False     Manual  Stopped  OK
Ignore  False     False
tdpipe   TDPIPE      c:\winnt\system32\drivers\tdpipe.sys
Kernel Driver      False     Manual  Stopped  OK
Ignore  False     False
tdspx    TDSPX      c:\winnt\system32\drivers\tdspx.sys
Kernel Driver      False     Manual  Stopped  OK
Ignore  False     False
tdtcp    TDTCP      c:\winnt\system32\drivers\tdtcp.sys
Kernel Driver      False     Manual  Stopped  OK
Ignore  False     False
termdd   Terminal Device Driver
c:\winnt\system32\drivers\termdd.sys      Kernel Driver
False    Disabled Stopped  OK      Normal  False
False
tga      tga        Not Available      Kernel Driver
False    System  Stopped  OK      Ignore   False
False
twintail IBM ServeRAID Failover Driver
c:\winnt\system32\drivers\twintail.sys      Kernel Driver
True     Boot    Running  OK      Normal  False
True
udfs     Udfs      c:\winnt\system32\drivers\udfs.sys      File
System Driver      False     Disabled Stopped  OK
Normal  False     False
uhcd     Microsoft USB Universal Host Controller Driver
c:\winnt\system32\drivers\uhcd.sys      Kernel Driver

```

```

True      Manual  Running  OK      Normal  False
True
update    Microcode Update Driver
c:\winnt\system32\drivers\update.sys      Kernel Driver
True     Manual  Running  OK      Normal  False
True
usbhub    Microsoft USB Standard Hub Driver
c:\winnt\system32\drivers\usbhub.sys      Kernel Driver
True     Manual  Running  OK      Normal  False
True
vgasave  VgaSave   c:\winnt\system32\drivers\vga.sys
Kernel Driver      True     System  Running  OK
Ignore  False     True
wanarp    Remote Access IP ARP Driver
c:\winnt\system32\drivers\wanarp.sys      Kernel Driver
True     Manual  Running  OK      Normal  False
True
wdica     WDICA     Not Available      Kernel Driver
False    Manual  Stopped  OK      Ignore   False
False

```

[Environment Variables]

Variable	Value	User Name
ComSpec	%SystemRoot%\system32\cmd.exe	<SYSTEM>
NUMBER_OF_PROCESSORS	8	<SYSTEM>
OS	Windows_NT	<SYSTEM>
Os2LibPath	%SystemRoot%\system32\os2\dll;	
<SYSTEM>		
Path	%SystemRoot%\system32;%SystemRoot%;%SystemRoot%\System32\Wbem;C:\Program Files\Microsoft SQL Server\MSSQL\BINN;C:\Program Files\Microsoft SQL Server\80\Tools\BINN	<SYSTEM>
PATHEXT	.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH	
<SYSTEM>		
PROCESSOR_ARCHITECTURE	x86	<SYSTEM>
PROCESSOR_IDENTIFIER	x86 Family 6 Model 10 Stepping 4, GenuineIntel	<SYSTEM>
PROCESSOR_LEVEL	6	<SYSTEM>
PROCESSOR_REVISION	0a04	<SYSTEM>
TEMP	%SystemRoot%\TEMP	<SYSTEM>
TMP	%SystemRoot%\TEMP	<SYSTEM>
windir	%SystemRoot%	<SYSTEM>
TEMP	%USERPROFILE%\Local Settings\Temp	
SGNODE01\Administrator		
TMP	%USERPROFILE%\Local Settings\Temp	
SGNODE01\Administrator		

[Jobs]

[ Following are sub-categories of this main category ]

[Print]

Document Size	Owner	Notify	Status	Time
Submitted	Start Time	Until Time	Elapsed Time	Pages Printed
Job ID	Priority	Parameters	Driver Name	Print Processor
Host Print Queue	Data	Type	Name	
No print jobs				

[Network Connections]

Local Name	Remote Name	Type	Status
User Name			



E: \\192.168.125.251\e\$\shogun Disk OK  
SGNODE01\Administrator

[Running Tasks]

Name	Path	Process ID	Priority	Min Working Set
Max Working Set	Start Time	Version	Size	File Date
system idle process	Not Available	0	0	Not
Available	Not Available	Not Available	Unknown	Unknown
Unknown	Unknown			
system	Not Available	8	8	0
1413120	Not Available	Unknown	Unknown	Unknown
smss.exe	c:\winnt\system32\smss.exe	184	11	
204800	1413120	3/12/2001 11:29:54 AM	5.00.2195.31	
44.27 KB (45,328 bytes)		7/27/2000 8:00:00 PM		
csrss.exe	Not Available	212	13	Not Available
Not Available	3/12/2001 11:29:58 AM		Unknown	
Unknown	Unknown			
winlogon.exe	c:\winnt\system32\winlogon.exe	232	13	
204800	1413120	3/12/2001 11:30:00 AM		
5.00.2195.1600		172.77 KB (176,912 bytes)	7/27/2000	
8:00:00 PM				
services.exe	c:\winnt\system32\services.exe			
260	9	204800 1413120 3/12/2001		
11:30:01 AM		5.00.2134.1	86.77 KB (88,848 bytes)	7/27/2000 8:00:00 PM
lsass.exe	c:\winnt\system32\lsass.exe	272		
13		204800 1413120 3/12/2001		
11:30:01 AM		5.00.2195.1620	32.77 KB (33,552 bytes)	7/27/2000 8:00:00 PM
gnconmgr.exe	c:\winnt\system32\gnconmgr.exe			
408	8	204800 1413120 3/12/2001		
11:30:03 AM		4.1.1	128.05 KB (131,119 bytes)	2/13/2001 9:01:54 AM
svchost.exe	c:\winnt\system32\svchost.exe	468		
8		204800 1413120 3/12/2001		
11:30:04 AM		5.00.2134.1	7.77 KB (7,952 bytes)	7/27/2000 8:00:00 PM
spoolsv.exe	c:\winnt\system32\spoolsv.exe	496		
8		204800 1413120 3/12/2001		
11:30:04 AM		5.00.2161.1	43.77 KB (44,816 bytes)	2/8/2001 10:57:40 AM
msdtc.exe	c:\winnt\system32\msdtc.exe	524		
8		204800 1413120 3/12/2001		
11:30:04 AM		1999.9.3421.3	6.77 KB (6,928 bytes)	2/8/2001 11:06:14 AM
svchost.exe	c:\winnt\system32\svchost.exe	800		
8		204800 1413120 3/12/2001		
11:30:12 AM		5.00.2134.1	7.77 KB (7,952 bytes)	7/27/2000 8:00:00 PM
llssrv.exe	c:\winnt\system32\llssrv.exe	824		
9		204800 1413120 3/12/2001		
11:30:12 AM		5.00.2167.1	114.27 KB (117,008 bytes)	7/27/2000 8:00:00 PM
proconsv.exe	c:\winnt\system32\proconsv.exe			
892	8	204800 1413120 3/12/2001		
11:30:13 AM		1, 0, 1, 23	72.27 KB (74,000 bytes)	2/8/2001 11:06:30 AM
regsvc.exe	c:\winnt\system32\regsvc.exe	956		
8		204800 1413120 3/12/2001		
11:30:14 AM		5.00.2195.31	65.27 KB (66,832 bytes)	7/27/2000 8:00:00 PM
proconmd8.exe	c:\winnt\system32\proconmd8.exe			
968	8	204800 1413120 3/12/2001		
11:30:14 AM		1, 0, 1, 23	24.27 KB (24,848 bytes)	2/8/2001 11:06:30 AM
rshsvc.exe	c:\winnt\system32\rshsvc.exe	848		
8		204800 1413120 3/12/2001		

11:30:17 AM Not Available 53.50 KB (54,784 bytes) 2/15/2001 11:03:08 AM

mstask.exe c:\winnt\system32\mstask.exe  
1028 8 204800 1413120 3/12/2001  
11:30:17 AM 4.71.2137.1 115.27 KB  
(118,032 bytes) 2/8/2001 4:08:40 PM

raiderv.exe c:\program  
files\raidman\raiderv.exe 1076 8  
204800 1413120 3/12/2001 11:30:17 AM Not  
Available 40.50 KB (41,472 bytes)  
2/12/2001 9:16:15 AM

tcpsvcs.exe c:\winnt\system32\tcpsvcs.exe  
1116 8 204800 1413120 3/12/2001  
11:30:17 AM 5.00.2134.1 24.77 KB (25,360  
bytes) 7/27/2000 8:00:00 PM

winmgmt.exe  
c:\winnt\system32\wbem\winmgmt.exe 1136 8  
204800 1413120 3/12/2001 11:30:17 AM  
1.50.1085.0009 192.08 KB (196,685 bytes)  
7/27/2000 8:00:00 PM

wins.exe c:\winnt\system32\wins.exe  
1152 8 204800 1413120 3/12/2001  
11:30:17 AM 5.00.2195.1600 147.77 KB  
(151,312 bytes) 2/8/2001 11:06:08 AM

dfssvc.exe c:\winnt\system32\dfssvc.exe  
1268 8 204800 1413120 3/12/2001  
11:30:22 AM 5.00.2195.1340 87.77 KB (89,872  
bytes) 7/27/2000 8:00:00 PM

explorer.exe c:\winnt\explorer.exe 1372 8  
204800 1413120 3/12/2001 11:31:13 AM  
5.00.3103.1000 237.27 KB (242,960 bytes)  
7/27/2000 8:00:00 PM

sqlmangr.exe c:\program files\microsoft sql  
server\80\tools\bin\sqlmangr.exe 1432 8  
204800 1413120 3/12/2001 11:31:14 AM  
2000.080.0194.00 68.00 KB (69,632 bytes)  
2/13/2001 3:21:26 PM

svchost.exe c:\winnt\system32\svchost.exe  
1560 8 204800 1413120 3/12/2001  
11:32:01 AM 5.00.2134.1 7.77 KB (7,952  
bytes) 7/27/2000 8:00:00 PM

cmd.exe c:\winnt\system32\cmd.exe 1364 8  
204800 1413120 3/12/2001 11:32:47 AM  
5.00.2195.1600 230.77 KB (236,304 bytes)  
7/27/2000 8:00:00 PM

cmd.exe c:\winnt\system32\cmd.exe 1088 8  
204800 1413120 3/12/2001 12:13:34 PM  
5.00.2195.1600 230.77 KB (236,304 bytes)  
7/27/2000 8:00:00 PM

calc.exe c:\winnt\system32\calc.exe  
1684 8 204800 1413120 3/12/2001 5:16:48  
PM 5.00.2134.1 89.27 KB (91,408 bytes)  
2/8/2001 11:06:31 AM

mmc.exe c:\winnt\system32\mmc.exe 1500 8  
204800 1413120 3/14/2001 7:56:47 AM  
5.00.2153.1 589.27 KB (603,408 bytes)  
7/27/2000 8:00:00 PM

rsvp.exe c:\winnt\system32\rsvp.exe  
2968 8 204800 1413120 3/14/2001 7:59:37  
AM 5.00.2167.1 172.77 KB (176,912 bytes)  
7/27/2000 8:00:00 PM

[Loaded Modules]

Name	Version	Size	File Date
Manufacturer	Path		

traffic.dll	5.00.2139.1	30.77 KB (31,504 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\traffic.dll
rsvp.exe	5.00.2167.1	172.77 KB (176,912 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\rsvp.exe
wbemprox.dll	1.50.1085.0015	40.08 KB (41,040 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\wbem\wbemprox.dll
mlang.dll	5.00.3103.1000	510.77 KB (523,024 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\mlang.dll
rassapi.dll	5.00.2188.1	14.27 KB (14,608 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\rassapi.dll
adsnt.dll	5.00.2195.1600	194.27 KB (198,928 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\adsnt.dll
dbghelp.dll	5.00.2195.1620	159.27 KB (163,088 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\dbghelp.dll
localesec.dll	5.00.2195.1340	227.27 KB (232,720 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\localesec.dll
devmgr.dll	5.00.2166.1	215.77 KB (220,944 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\devmgr.dll
filemgmt.dll	5.00.2134.1	287.27 KB (294,160 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\filemgmt.dll
pdh.dll	5.00.2195.1600	143.27 KB (146,704 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\pdh.dll
smlogcfg.dll	5.00.2163.1	273.27 KB (279,824 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\smlogcfg.dll
cabinet.dll	5.00.2147.1	54.77 KB (56,080 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\cabinet.dll
msinfo32.dll	5.00.2177.1	312.27 KB (319,760 bytes)	2/8/2001 4:08:48 PM	Microsoft Corporation	c:\program files\microsoft shared\msinfo\msinfo32.dll
riched20.dll	5.30.23.1203	421.27 KB (431,376 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\riched20.dll
riched32.dll	5.00.2134.1	3.77 KB (3,856 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\riched32.dll
els.dll	5.00.2175.1	151.27 KB (154,896 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\els.dll
ntsmmgr.dll	1,0,0,1	427.77 KB (438,032 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation and HighGround Systems, Inc.	c:\winnt\system32\ntsmmgr.dll
mmfutil.dll	1.50.1085.0000	32.06 KB (32,829 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\mmfutil.dll
logdrive.dll	1.50.1085.0000	200.06 KB (204,863 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\logdrive.dll
dfrgres.dll	5.00.2150.1	27.50 KB (28,160 bytes)	7/27/2000 8:00:00 PM	Executive Software International, Inc.	c:\winnt\system32\dfrgres.dll
dfrgsnap.dll	5.00.2195.31	41.77 KB (42,768 bytes)	7/27/2000 8:00:00 PM	Executive Software International, Inc.	c:\winnt\system32\dfrgsnap.dll
dmdskres.dll	2195.1600.297.3	119.50 KB (122,368 bytes)	7/27/2000 8:00:00 PM	Microsoft Corp., VERITAS Software	c:\winnt\system32\dmdskres.dll
dmutil.dll	2195.23.297.2	42.27 KB (43,280 bytes)	7/27/2000 8:00:00 PM	VERITAS Software Corp.	c:\winnt\system32\dmutil.dll
ntmsapi.dll	5.00.1948.1	50.27 KB (51,472 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\ntmsapi.dll
dmdskmgr.dll	2195.1600.297.3	160.27 KB (164,112 bytes)	7/27/2000 8:00:00 PM	Microsoft Corp., VERITAS Software	c:\winnt\system32\dmdskmgr.dll
mycomput.dll	5.00.2134.1	107.77 KB (110,352 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\mycomput.dll
mmcndmgr.dll	5.00.2178.1	815.27 KB (834,832 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\mmcndmgr.dll
mmc.exe	5.00.2153.1	589.27 KB (603,408 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\mmc.exe
calc.exe	5.00.2134.1	89.27 KB (91,408 bytes)	2/8/2001 11:06:31 AM	Microsoft Corporation	c:\winnt\system32\calc.exe
cmd.exe	5.00.2195.1600	230.77 KB (236,304 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\cmd.exe
tapisrv.dll	5.00.2195.1600	168.77 KB (172,816 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\tapisrv.dll
sqlmgr.rll	2000.080.0194.00	96.00 KB (98,304 bytes)	2/13/2001 3:21:26 PM	Microsoft Corporation	c:\program files\microsoft sql server\80\tools\binn\resources\1033\sqlmgr.rll
sqlsvr.rll	2000.080.0194.00	24.00 KB (24,576 bytes)	2/13/2001 1:51:05 PM	Microsoft Corporation	c:\program files\microsoft sql server\80\tools\binn\resources\1033\sqlsvr.rll
odbcint.dll	3.520.6526.0	88.00 KB (90,112 bytes)	2/13/2001 1:46:25 PM	Microsoft Corporation	c:\winnt\system32\odbcint.dll
sqlresld.dll	2000.080.0194.00	28.06 KB (28,738 bytes)	2/13/2001 1:51:00 PM	Microsoft Corporation	c:\program files\microsoft sql server\80\tools\binn\sqlresld.dll

odbcbcpl.dll	2000.080.0194.00	28.07 KB	(28,742 bytes)	2/13/2001 1:46:32 PM	Microsoft Corporation	c:\winnt\system32\odbcbcpl.dll
sqlsvcl.dll	2000.080.0194.00	92.06 KB	(94,272 bytes)	2/13/2001 1:51:00 PM	Microsoft Corporation	c:\program files\microsoft sql server\80\tools\bin\sqlsvcl.dll
sqlunirl.dll	2000.080.0194.00	176.06 KB	(180,290 bytes)	8/6/2000 2:51:56 AM	Microsoft Corporation	c:\winnt\system32\sqlunirl.dll
odbc32.dll	3.520.6526.0	216.27 KB	(221,456 bytes)	2/13/2001 1:46:25 PM	Microsoft Corporation	c:\winnt\system32\odbc32.dll
w95scm.dll	2000.080.0194.00	48.06 KB	(49,216 bytes)	2/13/2001 1:50:59 PM	Microsoft Corporation	c:\program files\microsoft sql server\80\tools\bin\w95scm.dll
sqlmangr.exe	2000.080.0194.00	68.00 KB	(69,632 bytes)	2/13/2001 3:21:26 PM	Microsoft Corporation	c:\program files\microsoft sql server\80\tools\bin\sqlmangr.exe
hhsetup.dll	4.74.8702	66.27 KB	(67,856 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\hhsetup.dll
mmschext.dll	5.00.2153.1	24.27 KB	(24,848 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\mmschext.dll
shdoclc.dll	5.00.3103.1000	324.50 KB	(332,288 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\shdoclc.dll
wininet.dll	5.00.3103.1000	456.77 KB	(467,728 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\wininet.dll
urlmon.dll	5.00.3103.1000	440.77 KB	(451,344 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\urlmon.dll
browseic.dll	5.00.3103.1000	34.50 KB	(35,328 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\browseic.dll
faxshell.dll	5.00.2134.1	8.27 KB	(8,464 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\faxshell.dll
msacm32.dll	5.00.2134.1	65.27 KB	(66,832 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\msacm32.dll
avifil32.dll	5.00.2134.1	76.27 KB	(78,096 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\avifil32.dll
msvfw32.dll	5.00.2134.1	113.77 KB	(116,496 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\msvfw32.dll
docprop2.dll	5.00.2195.1387	308.77 KB	(316,176 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\docprop2.dll
linkinfo.dll	5.00.2195.1387	16.77 KB	(17,168 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\linkinfo.dll
powrprof.dll	5.00.3103.1000	13.27 KB	(13,584 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\powrprof.dll
batmeter.dll	5.00.2920.0000	20.27 KB	(20,752 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\batmeter.dll
stobject.dll	5.00.2195.1387	79.27 KB	(81,168 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\stobject.dll
webcheck.dll	5.00.3103.1000	251.77 KB	(257,808 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\webcheck.dll
ntshrui.dll	5.00.2134.1	46.77 KB	(47,888 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\ntshrui.dll
mydocs.dll	5.00.3103.1000	57.77 KB	(59,152 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\mydocs.dll
browseui.dll	5.00.3103.1000	788.77 KB	(807,696 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\browseui.dll
shdocvw.dll	5.00.3103.1000	1.05 MB	(1,104,144 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\shdocvw.dll
explorer.exe	5.00.3103.1000	237.27 KB	(242,960 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\explorer.exe
dfssvc.exe	5.00.2195.1340	87.77 KB	(89,872 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\dfssvc.exe
wins.exe	5.00.2195.1600	147.77 KB	(151,312 bytes)	2/8/2001 11:06:08 AM	Microsoft Corporation	c:\winnt\system32\wins.exe
wshnetbs.dll	5.00.2134.1	7.77 KB	(7,952 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\wshnetbs.dll
rapilib.dll	5.00.2167.1	25.27 KB	(25,872 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\rapilib.dll
rsvsp.dll	5.00.2167.1	74.77 KB	(76,560 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\rsvsp.dll
provthrd.dll	1.50.1085.0000	68.07 KB	(69,708 bytes)	2/8/2001 4:08:39 PM	Microsoft Corporation	c:\winnt\system32\wbem\provthrd.dll
ntevt.dll	1.50.1085.0000	192.06 KB	(196,669 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\wbem\ntevt.dll
perfos.dll	5.00.2155.1	21.27 KB	(21,776 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\perfos.dll
framedyn.dll	1.50.1085.0000	164.05 KB	(167,992 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\wbem\framedyn.dll
cimwin32.dll	1.50.1085.0016	1.02 MB	(1,073,232 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\wbem\cimwin32.dll
wbemsvcl.dll	1.50.1085.0007	40.07 KB	(41,036 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\wbem\wbemsvcl.dll
wbemess.dll	1.50.1085.0007	364.07 KB	(372,804 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\wbem\wbemess.dll

Microsoft Corporation	psapi.dll	5.00.2134.1	28.27 KB (28,944 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\psapi.dll
c:\winnt\system32\wbem\wbemess.dll	proconsvcs.exe	1, 0, 1, 23	72.27 KB (74,000 bytes)	2/8/2001 11:06:30 AM	Sequent Computer Systems, Inc.	c:\winnt\system32\proconsvcs.exe
fastprox.dll 1.50.1085.0007 144.08 KB (147,536 bytes) 7/27/2000 8:00:00 PM	llsrpc.dll	5.00.2149.1	45.77 KB (46,864 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\llsrpc.dll
Microsoft Corporation	llssrv.exe	5.00.2167.1	114.27 KB (117,008 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\llssrv.exe
c:\winnt\system32\wbem\fastprox.dll	rasdlg.dll	5.00.2194.1	514.27 KB (526,608 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\rasdlg.dll
wbemcore.dll 1.50.1085.0008 628.07 KB (643,140 bytes) 7/27/2000 8:00:00 PM	netcfgx.dll	5.00.2195.1618	533.77 KB (546,576 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\netcfgx.dll
Microsoft Corporation	rasmans.dll	5.00.2195.27	146.77 KB (150,288 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\rasmans.dll
c:\winnt\system32\wbem\wbemcore.dll	wmi.dll	5.00.2195.1600	6.27 KB (6,416 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\wmi.dll
wbemcomn.dll 1.50.1085.0007 692.07 KB (708,675 bytes) 7/27/2000 8:00:00 PM	netshell.dll	5.00.2195.1600	456.77 KB (467,728 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\netshell.dll
Microsoft Corporation	netman.dll	5.00.2195.1600	89.27 KB (91,408 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\netman.dll
c:\winnt\system32\wbem\wbemcomn.dll	ntmsdba.dll	5.00.2195.1600	167.77 KB (171,792 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\ntmsdba.dll
winmgmt.exe 1.50.1085.0009 192.08 KB (196,685 bytes) 7/27/2000 8:00:00 PM	sens.dll	5.00.2163.1	36.77 KB (37,648 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\sens.dll
Microsoft Corporation	ntmssvc.dll	5.00.2187.1	390.77 KB (400,144 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\ntmssvc.dll
c:\winnt\system32\wbem\winmgmt.exe	es.dll	2000.2.3465.0	228.77 KB (234,256 bytes)	10/16/2000 2:51:22 PM	Microsoft Corporation	c:\winnt\system32\es.dll
simptcp.dll 5.00.2134.1 19.27 KB (19,728 bytes) 2/8/2001 11:06:09 AM	mtxoci.dll	2000.2.3465.0	101.77 KB (104,208 bytes)	10/16/2000 2:51:32 PM	Microsoft Corporation	c:\winnt\system32\mtxoci.dll
Microsoft Corporation	resutils.dll	5.00.2195.1613	39.77 KB (40,720 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\resutils.dll
c:\winnt\system32\simptcp.dll	clusapi.dll	5.00.2195.1613	54.27 KB (55,568 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\clusapi.dll
tcpsvcs.exe 5.00.2134.1 24.77 KB (25,360 bytes) 7/27/2000 8:00:00 PM	msvc50.dll	5.00.7051	552.50 KB (565,760 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\msvc50.dll
Microsoft Corporation	xolehlp.dll	1999.9.3421.3	17.27 KB (17,680 bytes)	2/8/2001 11:06:15 AM	Microsoft Corporation	c:\winnt\system32\xolehlp.dll
c:\winnt\system32\tcpsvcs.exe						
nfstcppr.dll Not Available 104.00 KB (106,496 bytes) 2/12/2001 9:16:03 AM						
Not Available c:\program files\raidman\nfstcppr.dll						
nfstjdll.dll Not Available 237.50 KB (243,200 bytes) 2/12/2001 9:16:03 AM						
Not Available c:\program files\raidman\nfstjdll.dll						
net.dll Not Available 22.50 KB (23,040 bytes) 2/12/2001 9:16:12 AM						
Not Available c:\program files\raidman\jre\bin\net.dll						
zip.dll Not Available 40.00 KB (40,960 bytes) 2/12/2001 9:16:12 AM						
Not Available c:\program files\raidman\jre\bin\zip.dll						
javai.dll Not Available 513.50 KB (525,824 bytes) 2/12/2001 9:16:12 AM						
Not Available c:\program files\raidman\jre\bin\javai.dll						
raidserv.exe Not Available 40.50 KB (41,472 bytes) 2/12/2001 9:16:15 AM						
Not Available c:\program files\raidman\raidserv.exe						
msidle.dll 5.00.3103.1000 6.27 KB (6,416 bytes) 7/27/2000 8:00:00 PM						
Microsoft Corporation						
c:\winnt\system32\msidle.dll						
mstask.exe 4.71.2137.1 115.27 KB (118,032 bytes) 2/8/2001 4:08:40 PM						
Microsoft Corporation						
c:\winnt\system32\mstask.exe						
rshsvcs.dll Not Available 18.00 KB (18,432 bytes) 2/15/2001 11:03:08 AM						
Not Available c:\winnt\system32\rshsvcs.dll						
rshsvcs.exe Not Available 53.50 KB (54,784 bytes) 2/15/2001 11:03:08 AM						
Not Available c:\winnt\system32\rshsvcs.exe						
proconcmd8.exe 1, 0, 1, 23 24.27 KB (24,848 bytes) 2/8/2001 11:06:30 AM						
Sequent Computer Systems, Inc.						
c:\winnt\system32\proconcmd8.exe						
regsvcs.exe 5.00.2195.31 65.27 KB (66,832 bytes) 7/27/2000 8:00:00 PM						
Microsoft Corporation						
c:\winnt\system32\regsvcs.exe						
ntmarta.dll 5.00.2158.1 98.77 KB (101,136 bytes) 7/27/2000 8:00:00 PM						
Microsoft Corporation						
c:\winnt\system32\ntmarta.dll						

msdtclog.dll	1999.9.3421.3	89.77 KB (91,920 bytes)	2/8/2001 11:06:14 AM	Microsoft Corporation	c:\winnt\system32\msdtclog.dll
mtxclu.dll	2000.2.3465.0	51.27 KB (52,496 bytes)	10/16/2000 2:51:30 PM	Microsoft Corporation	c:\winnt\system32\mtxclu.dll
msdtcprx.dll	2000.2.3465.0	626.27 KB (641,296 bytes)	10/16/2000 2:51:24 PM	Microsoft Corporation	c:\winnt\system32\msdtcprx.dll
txfaux.dll	2000.2.3465.0	373.77 KB (382,736 bytes)	10/16/2000 2:51:32 PM	Microsoft Corporation	c:\winnt\system32\txfaux.dll
msdtctm.dll	2000.2.3465.0	1.07 MB (1,120,528 bytes)	10/16/2000 2:51:28 PM	Microsoft Corporation	c:\winnt\system32\msdtctm.dll
msdtc.exe	1999.9.3421.3	6.77 KB (6,928 bytes)	2/8/2001 11:06:14 AM	Microsoft Corporation	c:\winnt\system32\msdtc.exe
inetpp.dll	5.00.2195.25	65.27 KB (66,832 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\inetpp.dll
win32spl.dll	5.00.2195.1340	92.27 KB (94,480 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\win32spl.dll
usbmon.dll	5.00.2165.1	11.27 KB (11,536 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\usbmon.dll
tcpmon.dll	5.00.2195.1225	40.77 KB (41,744 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\tcpmon.dll
pjlmon.dll	5.00.2165.1	12.77 KB (13,072 bytes)	6/9/2000 2:06:42 PM	Microsoft Corporation	c:\winnt\system32\pjlmon.dll
cnbjmon.dll	5.00.2134.1	43.77 KB (44,816 bytes)	6/9/2000 2:05:56 PM	Microsoft Corporation	c:\winnt\system32\cnbjmon.dll
localspl.dll	5.00.2195.1340	246.27 KB (252,176 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\localspl.dll
spoolss.dll	5.00.2161.1	61.77 KB (63,248 bytes)	2/8/2001 10:57:40 AM	Microsoft Corporation	c:\winnt\system32\spoolss.dll
spoolsv.exe	5.00.2161.1	43.77 KB (44,816 bytes)	2/8/2001 10:57:40 AM	Microsoft Corporation	c:\winnt\system32\spoolsv.exe
rpcss.dll	5.00.2195.2500	180.27 KB (184,592 bytes)	10/16/2000 3:52:26 PM	Microsoft Corporation	c:\winnt\system32\rpcss.dll
svchost.exe	5.00.2134.1	7.77 KB (7,952 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\svchost.exe
vipl.dll	4.1.1	80.00 KB (81,920 bytes)	2/13/2001 9:01:54 AM	Giganet Incorporated	c:\winnt\system32\vipl.dll
gnconmgr.exe	4.1.1	128.05 KB (131,119 bytes)	2/13/2001 9:01:54 AM	Giganet Incorporated	c:\winnt\system32\gnconmgr.exe
dssbase.dll	5.00.2195.1391	141.27 KB (144,656 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\dssbase.dll
oakley.dll	5.00.2195.1600	378.77 KB (387,856 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\oakley.dll
mfc42u.dll	6.00.8665.0	972.05 KB (995,384 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\mfc42u.dll
polagent.dll	5.00.2195.1600	93.27 KB (95,504 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\polagent.dll
scecli.dll	5.00.2191.1	105.27 KB (107,792 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\scecli.dll
atl.dll	3.00.8449	57.56 KB (58,938 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\atl.dll
certcli.dll	5.00.2175.1	132.27 KB (135,440 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\certcli.dll
ntdsatq.dll	5.00.2195.1284	31.27 KB (32,016 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\ntdsatq.dll
ntdsa.dll	5.00.2195.1600	987.27 KB (1,010,960 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\ntdsa.dll
kdcsvc.dll	5.00.2195.1284	133.77 KB (136,976 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\kdcsvc.dll
sfmapi.dll	5.00.2134.1	38.77 KB (39,696 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\sfmapi.dll
rassfm.dll	5.00.2195.1179	21.27 KB (21,776 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\rassfm.dll
schannel.dll	5.00.2195.1163	137.27 KB (140,560 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\schannel.dll
netlogon.dll	5.00.2195.1600	348.27 KB (356,624 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\netlogon.dll
msv1_0.dll	5.00.2195.1620	92.77 KB (94,992 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\msv1_0.dll
kerberos.dll	5.00.2195.1378	197.77 KB (202,512 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\kerberos.dll
msprivs.dll	5.00.2195.1600	41.50 KB (42,496 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\msprivs.dll
samsrv.dll	5.00.2195.1609	343.27 KB (351,504 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\samsrv.dll
lsasrv.dll	5.00.2195.1620	475.27 KB (486,672 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\lsasrv.dll
lsass.exe	5.00.2195.1620	32.77 KB (33,552 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\lsass.exe

esent.dll	6.0.3940.4	1.08 MB	(1,135,888 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\esent.dll
msi.dll	1.11.1314.0	1.72 MB	(1,798,928 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\msi.dll
appmgmts.dll	5.00.2168.1	117.77 KB	(120,592 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\appmgmts.dll
xactsrv.dll	5.00.2134.1	90.27 KB	(92,432 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\xactsrv.dll
wmicore.dll	5.00.2178.1	70.77 KB	(72,464 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\wmicore.dll
rasadhlp.dll	5.00.2168.1	7.27 KB	(7,440 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\rasadhlp.dll
winrnr.dll	5.00.2195.1175	19.27 KB	(19,728 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\winrnr.dll
rnr20.dll	5.00.2195.1207	35.77 KB	(36,624 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\rnr20.dll
wshtcpip.dll	5.00.2134.1	17.27 KB	(17,680 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\wshtcpip.dll
msafd.dll	5.00.2195.1614	102.77 KB	(105,232 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\msafd.dll
mswsock.dll	5.00.2195.1207	62.77 KB	(64,272 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\mswsock.dll
msgsvc.dll	5.00.2181.1	33.77 KB	(34,576 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\msgsvc.dll
browser.dll	5.00.2142.1	48.27 KB	(49,424 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\browser.dll
alrsvc.dll	5.00.2134.1	17.77 KB	(18,192 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\alrsvc.dll
trkwks.dll	5.00.2166.1	88.77 KB	(90,896 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\trkwks.dll
seclogon.dll	5.00.2135.1	15.77 KB	(16,144 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\seclogon.dll
psbase.dll	5.00.2195.1600	111.77 KB	(114,448 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\psbase.dll
cryptsvc.dll	5.00.2181.1	61.77 KB	(63,248 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\cryptsvc.dll
cryptdll.dll	5.00.2135.1	43.27 KB	(44,304 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\cryptdll.dll
wkssvc.dll	5.00.2195.1175	95.27 KB	(97,552 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\wkssvc.dll
srvsvc.dll	5.00.2178.1	79.27 KB	(81,168 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\srvsvc.dll
cfgmgr32.dll	5.00.2195.1608	16.77 KB	(17,168 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\cfgmgr32.dll
dmserver.dll	2195.23.297.2	11.77 KB	(12,048 bytes)	7/27/2000 8:00:00 PM	VERITAS Software Corp.	c:\winnt\system32\dmserver.dll
winsta.dll	5.00.2195.32	36.27 KB	(37,136 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\winsta.dll
lmhsvc.dll	5.00.2134.1	9.27 KB	(9,488 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\lmhsvc.dll
dnssrslvr.dll	5.00.2195.1600	88.77 KB	(90,896 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\dnssrslvr.dll
tapi32.dll	5.00.2195.1600	124.27 KB	(127,248 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\tapi32.dll
rasman.dll	5.00.2188.1	54.77 KB	(56,080 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\rasman.dll
rasapi32.dll	5.00.2188.1	187.77 KB	(192,272 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\rasapi32.dll
rtutils.dll	5.00.2168.1	43.77 KB	(44,816 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\rtutils.dll
adslrpc.dll	5.00.2195.1600	125.77 KB	(128,784 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\adslrpc.dll
activeds.dll	5.00.2172.1	172.77 KB	(176,912 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\activeds.dll
mprapi.dll	5.00.2181.1	79.27 KB	(81,168 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\mprapi.dll
iphlpapi.dll	5.00.2173.2	67.77 KB	(69,392 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\iphlpapi.dll
icmp.dll	5.00.2134.1	7.27 KB	(7,440 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\icmp.dll
dhcpcsvc.dll	5.00.2195.1600	85.27 KB	(87,312 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\dhcpcsvc.dll
eventlog.dll	5.00.2178.1	43.77 KB	(44,816 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\eventlog.dll
ntdsapi.dll	5.00.2195.1175	56.27 KB	(57,616 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\ntdsapi.dll
scesrv.dll	5.00.2188.1	225.77 KB	(231,184 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\scesrv.dll
umpnpmgr.dll	5.00.2182.1	86.27 KB	(88,336 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\umpnpmgr.dll
services.exe	5.00.2134.1	86.77 KB	(88,848 bytes)	7/27/2000 8:00:00 PM	Microsoft Corporation	c:\winnt\system32\services.exe
clbcatq.dll	2000.2.3465.0	496.27 KB	(508,176 bytes)	10/16/2000 2:51:16 PM		

Microsoft Corporation  
c:\winnt\system32\clbcatq.dll  
oleaut32.dll 2.40.4514 600.27 KB  
(614,672 bytes) 7/27/2000 8:00:00 PM  
Microsoft Corporation  
c:\winnt\system32\oleaut32.dll  
netmsg.dll 5.00.2137.1 152.50 KB  
(156,160 bytes) 7/27/2000 8:00:00 PM  
Microsoft Corporation  
c:\winnt\system32\netmsg.dll  
comdlg32.dll 5.00.3103.1000 236.77 KB  
(242,448 bytes) 7/27/2000 8:00:00 PM  
Microsoft Corporation  
c:\winnt\system32\comdlg32.dll  
netui2.dll 5.00.2134.1 280.27 KB  
(286,992 bytes) 7/27/2000 8:00:00 PM  
Microsoft Corporation  
c:\winnt\system32\netui2.dll  
mprui.dll 5.00.2134.1 54.77 KB (56,080  
bytes) 7/27/2000 8:00:00 PM Microsoft  
Corporation c:\winnt\system32\mprui.dll  
netui1.dll 5.00.2134.1 210.27 KB  
(215,312 bytes) 7/27/2000 8:00:00 PM  
Microsoft Corporation  
c:\winnt\system32\netui1.dll  
netui0.dll 5.00.2134.1 70.27 KB (71,952  
bytes) 7/27/2000 8:00:00 PM Microsoft  
Corporation c:\winnt\system32\netui0.dll  
ntlanman.dll 5.00.2157.1 35.27 KB (36,112  
bytes) 7/27/2000 8:00:00 PM Microsoft  
Corporation c:\winnt\system32\ntlanman.dll  
mpr.dll 5.00.2195.1340 53.27 KB (54,544 bytes)  
7/27/2000 8:00:00 PM Microsoft Corporation  
c:\winnt\system32\mpr.dll  
cscui.dll 5.00.2195.1387 227.27 KB  
(232,720 bytes) 7/27/2000 8:00:00 PM  
Microsoft Corporation  
c:\winnt\system32\cscui.dll  
winspool.drv 5.00.2195.1340 109.77 KB  
(112,400 bytes) 7/27/2000 8:00:00 PM  
Microsoft Corporation  
c:\winnt\system32\winspool.drv  
wincard.dll 5.00.2134.1 77.27 KB (79,120  
bytes) 7/27/2000 8:00:00 PM Microsoft  
Corporation c:\winnt\system32\wincard.dll  
wlnotify.dll 5.00.2195.1163 53.27 KB (54,544  
bytes) 7/27/2000 8:00:00 PM Microsoft  
Corporation c:\winnt\system32\wlnotify.dll  
csddl.dll 5.00.2195.1600 98.27 KB (100,624  
bytes) 7/27/2000 8:00:00 PM Microsoft  
Corporation c:\winnt\system32\csddl.dll  
lz32.dll 5.00.2134.1 9.77 KB (10,000  
bytes) 7/27/2000 8:00:00 PM Microsoft  
Corporation c:\winnt\system32\lz32.dll  
version.dll 5.00.2134.1 15.77 KB (16,144  
bytes) 7/27/2000 8:00:00 PM Microsoft  
Corporation c:\winnt\system32\version.dll  
rsabase.dll 5.00.2195.1391 129.27 KB  
(132,368 bytes) 7/27/2000 8:00:00 PM  
Microsoft Corporation  
c:\winnt\system32\rsabase.dll  
mscat32.dll 5.131.2134.1 7.77 KB (7,952 bytes)  
7/27/2000 8:00:00 PM Microsoft Corporation  
c:\winnt\system32\mscat32.dll  
ole32.dll 5.00.2195.2500 916.77 KB (938,768 bytes)  
10/16/2000 3:52:52 PM Microsoft Corporation  
c:\winnt\system32\ole32.dll

imagehlp.dll 5.00.2195.1620 120.77 KB (123,664  
bytes) 7/27/2000 8:00:00 PM Microsoft Corporation  
c:\winnt\system32\imagehlp.dll  
msasn1.dll 5.00.2134.1 51.27 KB (52,496 bytes)  
7/27/2000 8:00:00 PM Microsoft Corporation  
c:\winnt\system32\msasn1.dll  
crypt32.dll 5.131.2195.1340 464.77 KB (475,920 bytes)  
7/27/2000 8:00:00 PM Microsoft Corporation  
c:\winnt\system32\crypt32.dll  
wintrust.dll 5.131.2143.1 162.27 KB (166,160  
bytes) 7/27/2000 8:00:00 PM Microsoft Corporation  
c:\winnt\system32\wintrust.dll  
setupapi.dll 5.00.2195.1608 552.77 KB (566,032  
bytes) 7/27/2000 8:00:00 PM Microsoft Corporation  
c:\winnt\system32\setupapi.dll  
winmm.dll 5.00.2161.1 184.77 KB (189,200 bytes)  
7/27/2000 8:00:00 PM Microsoft Corporation  
c:\winnt\system32\winmm.dll  
comctl32.dll 5.81 537.77 KB (550,672 bytes)  
7/27/2000 8:00:00 PM Microsoft Corporation  
c:\winnt\system32\comctl32.dll  
shlwapi.dll 5.00.3103.1000 282.27 KB (289,040 bytes)  
7/27/2000 8:00:00 PM Microsoft Corporation  
c:\winnt\system32\shlwapi.dll  
shell32.dll 5.00.3103.1000 2.25 MB (2,358,032 bytes)  
7/27/2000 8:00:00 PM Microsoft Corporation  
c:\winnt\system32\shell32.dll  
msgina.dll 5.00.2195.1600 323.27 KB (331,024 bytes)  
7/27/2000 8:00:00 PM Microsoft Corporation  
c:\winnt\system32\msgina.dll  
wssock32.dll 5.00.2195.1207 21.27 KB (21,776 bytes)  
7/27/2000 8:00:00 PM Microsoft Corporation  
c:\winnt\system32\wssock32.dll  
dnsapi.dll 5.00.2195.1600 127.77 KB (130,832 bytes)  
7/27/2000 8:00:00 PM Microsoft Corporation  
c:\winnt\system32\dnsapi.dll  
wldap32.dll 5.00.2195.1175 155.27 KB (158,992  
bytes) 7/27/2000 8:00:00 PM Microsoft Corporation  
c:\winnt\system32\wldap32.dll  
ws2help.dll 5.00.2134.1 17.77 KB (18,192 bytes)  
7/27/2000 8:00:00 PM Microsoft Corporation  
c:\winnt\system32\ws2help.dll  
ws2\_32.dll 5.00.2195.1340 68.77 KB (70,416 bytes)  
7/27/2000 8:00:00 PM Microsoft Corporation  
c:\winnt\system32\ws2\_32.dll  
samlib.dll 5.00.2195.1175 46.27 KB (47,376 bytes)  
7/27/2000 8:00:00 PM Microsoft Corporation  
c:\winnt\system32\samlib.dll  
netrap.dll 5.00.2134.1 11.27 KB (11,536 bytes)  
7/27/2000 8:00:00 PM Microsoft Corporation  
c:\winnt\system32\netrap.dll  
netapi32.dll 5.00.2195.1600 303.27 KB (310,544  
bytes) 7/27/2000 8:00:00 PM Microsoft Corporation  
c:\winnt\system32\netapi32.dll  
profmap.dll 5.00.2181.1 29.27 KB (29,968 bytes)  
7/27/2000 8:00:00 PM Microsoft Corporation  
c:\winnt\system32\profmap.dll  
secur32.dll 5.00.2195.1600 47.27 KB (48,400 bytes)  
7/27/2000 8:00:00 PM Microsoft Corporation  
c:\winnt\system32\secur32.dll  
sfc.dll 5.00.2195.1618 85.27 KB (87,312 bytes)  
7/27/2000 8:00:00 PM Microsoft Corporation  
c:\winnt\system32\sfc.dll  
nddeapi.dll 5.00.2137.1 15.27 KB (15,632 bytes)  
7/27/2000 8:00:00 PM Microsoft Corporation  
c:\winnt\system32\nddeapi.dll

userenv.dll 5.00.2195.1600 359.27 KB (367,888 bytes)  
 7/27/2000 8:00:00 PM Microsoft Corporation  
 c:\winnt\system32\userenv.dll  
 user32.dll 5.00.2195.1600 392.77 KB (402,192 bytes)  
 7/27/2000 8:00:00 PM Microsoft Corporation  
 c:\winnt\system32\user32.dll  
 gdi32.dll 5.00.2195.1340 228.77 KB (234,256 bytes)  
 7/27/2000 8:00:00 PM Microsoft Corporation  
 c:\winnt\system32\gdi32.dll  
 rpctr4.dll 5.00.2195.1615 436.27 KB (446,736 bytes)  
 7/27/2000 8:00:00 PM Microsoft Corporation  
 c:\winnt\system32\rpctr4.dll  
 advapi32.dll 5.00.2195.1600 349.27 KB (357,648 bytes)  
 7/27/2000 8:00:00 PM Microsoft Corporation  
 c:\winnt\system32\advapi32.dll  
 kernel32.dll 5.00.2195.1600 713.27 KB (730,384 bytes)  
 7/27/2000 8:00:00 PM Microsoft Corporation  
 c:\winnt\system32\kernel32.dll  
 msvcr7.dll 6.10.8637.0 288.09 KB (295,000 bytes)  
 7/27/2000 8:00:00 PM Microsoft Corporation  
 c:\winnt\system32\msvcr7.dll  
 winlogon.exe 5.00.2195.1600 172.77 KB (176,912 bytes)  
 7/27/2000 8:00:00 PM Microsoft Corporation  
 c:\winnt\system32\winlogon.exe  
 sfcdll.dll 5.00.2195.1620 885.27 KB (906,512 bytes)  
 7/27/2000 8:00:00 PM Microsoft Corporation  
 c:\winnt\system32\sfcdll.dll  
 ntdll.dll 5.00.2195.1600 475.27 KB (486,672 bytes)  
 7/27/2000 8:00:00 PM Microsoft Corporation  
 c:\winnt\system32\ntdll.dll  
 smss.exe 5.00.2195.31 44.27 KB (45,328 bytes)  
 7/27/2000 8:00:00 PM Microsoft Corporation  
 c:\winnt\system32\smss.exe

[Services]

Display Name	Name	State	Start Mode	Start Name
Service Type	Path	Error Control	Start Name	
Tag ID				
Alerter	Alerter	Running	Auto	Share Process
c:\winnt\system32\services.exe		Normal	LocalSystem	0
Application Management	AppMgmt	Running	Manual	
Share Process	c:\winnt\system32\services.exe	Normal		
LocalSystem	0			
Computer Browser	Browser	Running	Auto	Share Process
c:\winnt\system32\services.exe		Normal	LocalSystem	0
Indexing Service	cisvc	Stopped	Manual	Share Process
c:\winnt\system32\cisvc.exe		Normal	LocalSystem	0
ClipBook	ClipSrv	Stopped	Manual	Own Process
c:\winnt\system32\clipsrv.exe		Normal	LocalSystem	0
Distributed File System	Dfs	Running	Auto	
Own Process	c:\winnt\system32\dfsrv.exe	Normal		
LocalSystem	0			
DHCP Client	Dhcp	Running	Auto	Share Process
c:\winnt\system32\services.exe		Normal	LocalSystem	0
Logical Disk Manager Administrative Service				dmadmin
Stopped	Manual	Share Process		
c:\winnt\system32\dmadmin.exe	/com	Normal	LocalSystem	0
Logical Disk Manager				dmserver
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		
Fax Service	Fax	Stopped	Manual	Own Process
c:\winnt\system32\faxsvc.exe		Normal	LocalSystem	0
cLAN Connection Manager	GniConMgr	Running		
Auto	Own Process			c:\winnt\system32\gnconmgr.exe
Normal	LocalSystem	0		
Intersite Messaging	IsmServ	Stopped	Disabled	Own Process
c:\winnt\system32\ismserv.exe		Normal	LocalSystem	0
Kerberos Key Distribution Center	kdc	Stopped		
Disabled	Share Process			c:\winnt\system32\lsass.exe
Normal	LocalSystem	0		
Server	lanmanserver	Running	Auto	Share Process
c:\winnt\system32\services.exe		Normal	LocalSystem	0
Workstation	lanmanworkstation	Running	Auto	
Share Process	c:\winnt\system32\services.exe	Normal		
LocalSystem	0			
License Logging Service	LicenseService	Running		
Auto	Own Process			c:\winnt\system32\llssrv.exe
Normal	LocalSystem	0		
TCP/IP NetBIOS Helper Service	LmHosts	Running	Auto	
Share Process	c:\winnt\system32\services.exe	Normal		
LocalSystem	0			
Messenger	Messenger	Running	Auto	Share Process
c:\winnt\system32\services.exe		Normal	LocalSystem	0
NetMeeting Remote Desktop Sharing	mnmsrvc	Stopped		
Manual	Own Process			c:\winnt\system32\mnmsrvc.exe
Normal	LocalSystem	0		
Distributed Transaction Coordinator	MSDTC	Running		
Auto	Own Process			c:\winnt\system32\msdtc.exe
Normal	LocalSystem	0		
Windows Installer	MSIServer	Stopped	Manual	Share Process
c:\winnt\system32\msiexec.exe /v		Normal	LocalSystem	0
MSSQLSERVER	MSSQLSERVER	Stopped	Manual	
Own Process	c:\progra~1\microso~2\mssql\binn\sqlservr.exe			
Normal	LocalSystem	0		
MSSQLServerADHelper	MSSQLServerADHelper			
Stopped	Manual	Own Process		c:\program files\microsoft sql server\80\tools\binn\sqladhlp.exe
Normal	LocalSystem	0		
Network DDE	NetDDE	Stopped	Manual	Share Process
c:\winnt\system32\netdde.exe		Normal	LocalSystem	0
Network DDE DSDM	NetDDEdsdm	Stopped	Manual	
Share Process	c:\winnt\system32\netdde.exe	Normal		
LocalSystem	0			
Net Logon	Netlogon	Stopped	Manual	Share Process
c:\winnt\system32\lsass.exe		Normal	LocalSystem	0
Network Connections	Netman	Running	Manual	Share Process
c:\winnt\system32\svchost.exe -k netsvcs		Normal	LocalSystem	0
File Replication	NtFrs	Stopped	Manual	Own Process
c:\winnt\system32\ntfrs.exe		Ignore	LocalSystem	0
NT LM Security Support Provider	NtLmSsp	Running		
Manual	Share Process			c:\winnt\system32\lsass.exe
Normal	LocalSystem	0		
Removable Storage	NtmsSvc	Running	Auto	Share Process
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
IPSEC Policy Agent	PolicyAgent	Running	Auto	
Share Process	c:\winnt\system32\lsass.exe	Normal		
LocalSystem	0			
Process Control Service	ProcCon	Running	Auto	
Own Process	c:\winnt\system32\proconsvr.exe			
Normal	LocalSystem	0		



Protected Storage	ProtectedStorage	Running	Auto	
Share Process	c:\winnt\system32\services.exe	Normal		
LocalSystem	0			
Remote Access Auto Connection Manager	RasAuto	Stopped		
Manual Share Process	c:\winnt\system32\svchost.exe -k			
netshvc	Normal LocalSystem	0		
Remote Access Connection Manager	RasMan	Stopped		
Manual Share Process	c:\winnt\system32\svchost.exe -k			
netshvc	Normal LocalSystem	0		
Routing and Remote Access	RemoteAccess	Stopped		
Disabled Share Process	c:\winnt\system32\svchost.exe -k			
netshvc	Normal LocalSystem	0		
Remote Registry Service	RemoteRegistry	Running		
Auto Own Process	c:\winnt\system32\regsvc.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 rpsvc			
Normal LocalSystem	0			
Remote Shell Service RshSvc	Running	Auto	Own Process	
c:\winnt\system32\rshsvc.exe	Normal LocalSystem	0		
QoS RSVP	RSVP	Running	Manual	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		
Task Scheduler	Schedule	Running	Auto	Share Process
c:\winnt\system32\mstask.exe	Normal LocalSystem	0		
RunAs Service	seclogon	Running	Auto	Share Process
c:\winnt\system32\services.exe	Ignore LocalSystem	0		
System Event Notification	SENS	Running	Auto	
Share Process	c:\winnt\system32\svchost.exe -k netshvc			
Normal LocalSystem	0			
ServeRAID Manager Agent	ServeRAIDManagerAgent			
Running Auto Own Process	c:\program			
files\raidman\raidserv.exe	Normal LocalSystem	0		
Internet Connection Sharing	SharedAccess	Stopped		
Manual Share Process	c:\winnt\system32\svchost.exe -k			
netshvc	Normal LocalSystem	0		
Simple TCP/IP Services	SimpTcp	Running	Auto	
Share Process	c:\winnt\system32\tcpshvc.exe	Normal		
LocalSystem	0			
Print Spooler	Spooler	Running	Auto	Own Process
c:\winnt\system32\spoolsv.exe	Normal LocalSystem	0		
SQLSERVERAGENT	SQLSERVERAGENT			
Stopped Manual Own Process				
c:\progra~1\microso~2\mssql\bin\sqlagent.exe	Normal			
LocalSystem	0			
Performance Logs and Alerts	SysmonLog	Stopped		
Manual Own Process	c:\winnt\system32\smlogsvc.exe			
Normal LocalSystem	0			
Telephony TapiSrv	Running	Manual	Share Process	
c:\winnt\system32\svchost.exe -k tapisrv	Normal LocalSystem	0		
Terminal Services	TermService	Stopped	Disabled	
Own Process	c:\winnt\system32\termsrv.exe	Normal		
LocalSystem	0			
Telnet	TlntSvr	Stopped	Manual	Own Process
c:\winnt\system32\tlntsvr.exe	Normal LocalSystem	0		
Distributed Link Tracking ServerTrkSvr	Stopped	Manual		
Share Process	c:\winnt\system32\services.exe	Normal		
LocalSystem	0			

Distributed Link Tracking Client TrkWks	Running	Auto		
Share Process	c:\winnt\system32\services.exe	Normal		
LocalSystem	0			
Uninterruptible Power Supply	UPS	Stopped	Manual	
Own Process	c:\winnt\system32\ups.exe	Normal		
LocalSystem	0			
Utility Manager	UtilMan	Stopped	Manual	Own Process
c:\winnt\system32\utilman.exe	Normal LocalSystem	0		
Windows Time	W32Time	Stopped	Manual	Share Process
c:\winnt\system32\services.exe	Normal LocalSystem	0		
Windows Management Instrumentation	WinMgmt	Running		
Auto Own Process				
c:\winnt\system32\wbem\winmgmt.exe	Ignore	LocalSystem		
0				
Windows Internet Name Service (WINS)	WINS	Running		
Auto Own Process	c:\winnt\system32\wins.exe			
Normal LocalSystem	0			
Windows Management Instrumentation Driver Extensions				
Wmi	Running	Manual	Share Process	
c:\winnt\system32\services.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	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\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	
ServeRAID Manager	All Users:ServeRAID Manager	All Users	
Startup	All Users:Startup	All Users	
Accessories	SGNODE01\Administrator:Accessories		
SGNODE01\Administrator			
Accessories\Accessibility			
SGNODE01\Administrator:Accessories\Accessibility			
SGNODE01\Administrator			
Accessories\Entertainment			
SGNODE01\Administrator:Accessories\Entertainment			
SGNODE01\Administrator			
Accessories\System Tools			
SGNODE01\Administrator:Accessories\System Tools			
SGNODE01\Administrator			
Administrative Tools	SGNODE01\Administrator:Administrative Tools		
SGNODE01\Administrator			
Startup	SGNODE01\Administrator:Startup		
SGNODE01\Administrator			

[Startup Programs]

Program	Command	User Name	Location
synctime	synctime.cmd	SGNODE01\Administrator	
Startup			

Service Manager  
 c:\progra~1\micro~2\80\tools\bin\sqlmgr.exe /n All Users  
 Common Startup

[OLE Registration]

Object	Local Server
Sound (OLE2)	sndrec32.exe
Media Clip	mplay32.exe
Video Clip	mplay32.exe /avi
MIDI Sequence	mplay32.exe /mid
Sound	Not Available
Media Clip	Not Available
Image Document	"C:\Program Files\Windows NT\Accessories\ImageVue\KodakImg.exe"
WordPad Document	"%ProgramFiles%\Windows NT\Accessories\WORDPAD.EXE"
Windows Media Services DRM Storage object	Not Available
Bitmap Image	mspaint.exe

[Internet Explorer 5]

[ Following are sub-categories of this main category ]

[Summary]

Item	Value
Version	5.00.3103.1000
Build	53103.1000
Product ID	51891-000-0000007-05809
Application Path	C:\Program Files\Internet Explorer
Language	English (United States)
Active Printer	Not Available
Cipher Strength	56-bit
Content Advisor	Disabled
IEAK Install	No

[File Versions]

File	Version	Size	Date	Path	Company
advapi32.dll	5.0.2195.1600	349 KB	7/27/2000 7:00:00 PM	C:\WINNT\system32	Microsoft Corporation
advpack.dll	5.0.3103.1000	87 KB	7/27/2000 7:00:00 PM	C:\WINNT\system32	Microsoft Corporation
browsecl.dll	5.0.3103.1000	35 KB	7/27/2000 7:00:00 PM	C:\WINNT\system32	Microsoft Corporation
browseui.dll	5.0.3103.1000	789 KB	7/27/2000 7:00:00 PM	C:\WINNT\system32	Microsoft Corporation
ckcnv.exe	5.0.2195.1600	9 KB	7/27/2000 7:00:00 PM	C:\WINNT\system32	Microsoft Corporation
comctl32.dll	5.81.3103.1000	538 KB	7/27/2000 7:00:00 PM	C:\WINNT\system32	Microsoft Corporation
crypt32.dll	5.131.2195.1340	465 KB	7/27/2000 7:00:00 PM	C:\WINNT\system32	Microsoft Corporation
enhsg.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
iemigrat.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
iesetup.dll	5.0.3103.1000	57 KB	7/27/2000 7:00:00 PM	C:\WINNT\system32	Microsoft Corporation
ieexplore.exe	5.0.3103.1000	59 KB	7/27/2000 7:00:00 PM	C:\Program Files\Internet Explorer	Microsoft Corporation
imagehlp.dll	5.0.2195.1620	121 KB	7/27/2000 7:00:00 PM	C:\WINNT\system32	Microsoft Corporation
imghelp.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available

inseng.dll	5.0.3103.1000	72 KB	7/27/2000 7:00:00 PM	C:\WINNT\system32	Microsoft Corporation
jobexec.dll	5.0.0.1	47 KB	7/27/2000 7:00:00 PM	C:\WINNT\system32	Microsoft Corporation
jscrip.dll	5.1.0.5010	476 KB	7/27/2000 7:00:00 PM	C:\WINNT\system32	Microsoft Corporation
jsproxy.dll	5.0.3103.1000	13 KB	7/27/2000 7:00:00 PM	C:\WINNT\system32	Microsoft Corporation
msahtml.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
mshhtml.dll	5.0.3103.1000	2292 KB	7/27/2000 7:00:00 PM	C:\WINNT\system32	Microsoft Corporation
msjava.dll	5.0.3310.0	922 KB	7/27/2000 7:00:00 PM	C:\WINNT\system32	Microsoft Corporation
msoss.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
msxml.dll	8.0.5226.0	506 KB	7/27/2000 7:00:00 PM	C:\WINNT\system32	Microsoft Corporation
occache.dll	5.0.3103.1000	86 KB	7/27/2000 7:00:00 PM	C:\WINNT\system32	Microsoft Corporation
ole32.dll	5.0.2195.2500	917 KB	10/16/2000 2:52:52 PM	C:\WINNT\system32	Microsoft Corporation
oleaut32.dll	2.40.4514.1	600 KB	7/27/2000 7:00:00 PM	C:\WINNT\system32	Microsoft Corporation
olepro32.dll	5.0.4514.1	160 KB	7/27/2000 7:00:00 PM	C:\WINNT\system32	Microsoft Corporation
rsabase.dll	5.0.2195.1391	129 KB	7/27/2000 7:00:00 PM	C:\WINNT\system32	Microsoft Corporation
rsaenh.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
rsapi32.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
rsasig.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
schannel.dll	5.0.2195.0	137 KB	7/27/2000 7:00:00 PM	C:\WINNT\system32	Microsoft Corporation
shdoc401.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
shdocvw.dll	5.0.3103.1000	1078 KB	7/27/2000 7:00:00 PM	C:\WINNT\system32	Microsoft Corporation
shell32.dll	5.0.3103.1000	2303 KB	7/27/2000 7:00:00 PM	C:\WINNT\system32	Microsoft Corporation
shlwapi.dll	5.0.3103.1000	282 KB	7/27/2000 7:00:00 PM	C:\WINNT\system32	Microsoft Corporation
url.dll	5.0.3103.1000	82 KB	7/27/2000 7:00:00 PM	C:\WINNT\system32	Microsoft Corporation
urlmon.dll	5.0.3103.1000	441 KB	7/27/2000 7:00:00 PM	C:\WINNT\system32	Microsoft Corporation
vbscript.dll	5.1.0.5010	428 KB	7/27/2000 7:00:00 PM	C:\WINNT\system32	Microsoft Corporation
webcheck.dll	5.0.3103.1000	252 KB	7/27/2000 7:00:00 PM	C:\WINNT\system32	Microsoft Corporation
win.com	5.0.2134.1	24 KB	7/27/2000 7:00:00 PM	C:\WINNT\system32	Microsoft Corporation
wininet.dll	5.0.3103.1000	457 KB	7/27/2000 7:00:00 PM	C:\WINNT\system32	Microsoft Corporation
winsock.dll	3.10.0.103	3 KB	7/27/2000 7:00:00 PM	C:\WINNT\system32	Microsoft Corporation
wintrust.dll	5.131.2143.1	162 KB	7/27/2000 7:00:00 PM	C:\WINNT\system32	Microsoft Corporation
wsock.vxd	<File Missing>	Not Available	Not Available	Not Available	Not Available
wsock32.dll	5.0.2195.1207	21 KB	7/27/2000 7:00:00 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	Not Available
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	8675 MB
Available Disk Space	4875 MB
Maximum Cache Size	271 MB
Available Cache Size	271 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	2/8/2001 to 1/15/2101	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

## Client Configuration

### Transaction Monitor: COM+ Settings on Clients

SYTPTUNE.TXT

Transaction Monitor: COM+ Settings on Clients

TPCC COM+ Component settings

TPCC.AllRemoteTxns.1

Activation  
 Enable Object pooling  
 Minimum pool size 10/8  
 Maximum pool size 10/8  
 Creation timeout (ms) 300000  
 Enable object construction  
 Object construction  
 Constructor string 'dummy string (do not remove)'  
 Enable Just In Time Activation  
 Disable Component supports events and statistics  
 Disable Must be activated in caller's context

TPCC.AllTxns.1

Activation  
 Enable Object pooling  
 Minimum pool size 40/32  
 Maximum pool size 40/32  
 Creation timeout (ms) 300000  
 Enable object construction  
 Object construction  
 Constructor string 'dummy string (do not remove)'  
 Enable Just In Time Activation  
 Disable Component supports events and statistics  
 Disable Must be activated in caller's context

### Windows Registry Editor Version 5.00

Windows Registry Editor Version 5.00

[HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\InternetInfo]

[HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\InternetInfo\Parameters]

"ListenBackLog"=dword:00000096  
 "DispatchEntries"=hex(7):4c,00,44,00,41,00,50,00,53,00,56,00,43,00,00,00,53,00,4d,00,54,00,50,00,53,00,56,00,43,00,00,00,00,00  
 "PoolThreadLimit"=dword:000000be  
 "ThreadTimeout"=dword:00015180

[HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\InternetInfo\Performance]

"Library"="infoctrs.dll"  
 "Open"="OpenINFOPerformanceData"  
 "Close"="CloseINFOPerformanceData"  
 "Collect"="CollectINFOPerformanceData"  
 "Last Counter"=dword:00000bcc  
 "Last Help"=dword:00000bcd  
 "First Counter"=dword:00000b8c  
 "First Help"=dword:00000b8d

"Library Validation  
Code"=hex:c2,f1,19,d5,83,58,c0,01,10,25,00,00,00,00,00  
"WbemAdapFileTime"=hex:00,20,2b,88,67,40,bf,01  
"WbemAdapFileSize"=dword:00002510  
"WbemAdapStatus"=dword:00000000

Windows Registry Editor Version 5.00

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\TPCC]
"Path"="c:\inetpub\wwwroot\"
"NumberOfDeliveryThreads"=dword:00000009
"MaxConnections"=dword:00002710
"MaxPendingDeliveries"=dword:000003e8
"DB_Protocol"="ODBC"
"TxnMonitor"="COM"
"DbServer"="serv1"
"DbName"="tpcc"
"DbUser"="sa"
"DbPassword"=""
"COM_SinglePool"="YES"
```

Windows Registry Editor Version 5.00

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W
3SVC]
"Type"=dword:00000020
"Start"=dword:00000002
"ErrorControl"=dword:00000001
"ImagePath"=hex(2):43,00,3a,00,5c,00,57,00,49,00,4e,00,4e,00,54,00,
5c,00,53,00,\
79,00,73,00,74,00,65,00,6d,00,33,00,32,00,5c,00,69,00,6e,00,65,00,7
4,00,73,\
```

```
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
"DisplayName"="World Wide Web Publishing Service"
"DependOnService"=hex(7):49,00,49,00,53,00,41,00,44,00,4d,00,49,0
0,4e,00,00,00,\
00,00
"DependOnGroup"=hex(7):00,00
"ObjectName"="LocalSystem"
"Description"="Provides Web connectivity and administration through
the Internet Information Services snap-in."
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W
3SVC\ASP]
"NOTE"="This is for backward compatibility only."
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W
3SVC\ASP\Parameters]
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W
3SVC\Parameters]
"MajorVersion"=dword:00000005
"MinorVersion"=dword:00000000
"InstallPath"="C:\\WINNT\\System32\\inetsrv"
"CertMapList"="C:\\WINNT\\System32\\inetsrv\\iisrmap.dll"
"AccessDeniedMessage"="Error: Access is Denied."
"Filter DLLs"=""
"LogFileDirectory"="C:\\WINNT\\System32\\LogFiles"
"AcceptExOutstanding"=dword:00000028
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W
3SVC\Parameters\Script Map]
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W
3SVC\Parameters\Virtual Roots]
"/"="c:\\inetpub\\wwwroot,205"
"/Scripts"="c:\\inetpub\\scripts,1"
"/IISHelp"="c:\\winnt\\help\\iishelp,1"
"/IISAdmin"="C:\\WINNT\\System32\\inetsrv\\iisadmin,1"
"/IISSamples"="c:\\inetpub\\iissamples,1"
"/MSADC"="c:\\program files\\common files\\system\\msadc,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\W
3SVC\Performance]
"Library"="w3ctrs.dll"
"Open"="OpenW3PerformanceData"
"Close"="CloseW3PerformanceData"
"Collect"="CollectW3PerformanceData"
"Last Counter"=dword:00000c70
"Last Help"=dword:00000c71
"First Counter"=dword:00000bce
"First Help"=dword:00000bcf
"Library Validation  
Code"=hex:b0,e3,f5,d7,83,58,c0,01,10,3d,00,00,00,00,00,00
"WbemAdapFileTime"=hex:00,20,2b,88,67,40,bf,01
"WbemAdapFileSize"=dword:00003d10
"WbemAdapStatus"=dword:00000000
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W
3SVC\Security]
"Security"=hex:01,00,14,80,a0,00,00,00,ac,00,00,00,14,00,00,00,30,0
0,00,00,02,\
00,1c,00,01,00,00,00,02,80,14,00,ff,01,0f,00,01,01,00,00,00,00,01,
00,00,\
00,00,02,00,70,00,04,00,00,00,00,00,18,00,fd,01,02,00,01,01,00,00,
0,00,00,\
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,01,02,00,01,01,0
0,00,00,\
00,00,05,0b,00,00,00,20,02,00,00,00,00,1c,00,fd,01,02,00,01,02,00,0
0,00,00,\
00,05,20,00,00,00,23,02,00,00,72,00,73,00,01,01,00,00,00,00,05,1
2,00,00,\
00,01,01,00,00,00,00,00,05,12,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W
3SVC\Enum]
"0"="Root\\LEGACY_W3SVC\\0000"
"Count"=dword:00000001
"NextInstance"=dword:00000001
```

## System Information Report for Clients

### Netfinity 5100

This system configuration inventory was obtained from 1 of 8 clients (Netfinity 5100).

All 8 clients are identically configured.  
 System Information report written at: 03/19/2001 02:09:05 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 Build 2195
OS Manufacturer	Microsoft Corporation
System Name	RTCLIENT64
System Manufacturer	IBM
System Model	Netfinity 5100 -[86582RY]-
System Type	X86-based PC
Processor	x86 Family 6 Model 8 Stepping 3 GenuineIntel ~731 Mhz
BIOS Version	IBM BIOS Ver 3.0
Windows Directory	C:\WINNT
System Directory	C:\WINNT\System32
Boot Device	\Device\Harddisk0\Partition1
Locale	United States
User Name	RTCLIENT64\tpcc
Time Zone	Eastern Standard Time
Total Physical Memory	523,796 KB
Available Physical Memory	429,364 KB
Total Virtual Memory	1,801,936 KB
Available Virtual Memory	1,635,996 KB
Page File Space	1,278,140 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
2	Standard floppy disk controller	OK
4	Direct memory access controller	OK

[Forced Hardware]

Device PNP Device ID  
 No Forced Hardware

[I/O]

Address Range	Device	Status
0x0000-0x0CF7	PCI bus	OK
0x0000-0x0CF7	Direct memory access controller	OK
0x0D00-0x4AEB	PCI bus	OK
0x82E8-0xFFFF	PCI bus	OK
0x03B0-0x03BB	S3 Inc. Savage4	OK
0x03C0-0x03DF	S3 Inc. Savage4	OK
0x2000-0x201F	IBM Netfinity Fault Tolerance PCI Adapter	OK
0x0A79-0x0A79	ISAPNP Read Data Port	OK
0x0279-0x0279	ISAPNP Read Data Port	OK
0x02F4-0x02F7	ISAPNP Read Data Port	OK
0x0060-0x0060	Standard 101/102-Key or Microsoft Natural	
PS/2 Keyboard	OK	

0x0064-0x0064	Standard 101/102-Key or Microsoft Natural	
PS/2 Keyboard	OK	
0x03F0-0x03F5	Standard floppy disk controller	OK
0x03F7-0x03F7	Standard floppy disk controller	OK
0x0378-0x037F	Printer Port (LPT1)	OK
0x03F8-0x03FF	Communications Port (COM1)	OK
0x02F8-0x02FF	Communications Port (COM2)	OK
0x0020-0x0021	Advanced programmable interrupt controller	OK
0x00A0-0x00A1	Advanced programmable interrupt controller	OK
0x04D0-0x04D1	Advanced programmable interrupt controller	OK
0x0080-0x008F	Direct memory access controller	OK
0x00C0-0x00DF	Direct memory access controller	OK
0x0040-0x0043	System timer	OK
0x0070-0x0073	System CMOS/real time clock	OK
0x0061-0x0061	System speaker	OK
0x00F0-0x00FF	Numeric data processor	OK
0x0F50-0x0F58	Motherboard resources	OK
0xFD00-0xFD3F	Motherboard resources	OK
0xFE80-0xFEBF	Motherboard resources	OK
0xFEC0-0xFEDF	Motherboard resources	OK
0xEE9B-0xEE9B	Motherboard resources	OK
0x00E8-0x00E9	Not Available	OK
0x0840-0x084F	Standard Dual Channel PCI IDE Controller	OK
0x01F0-0x01F7	Primary IDE Channel	OK
0x03F6-0x03F6	Primary IDE Channel	OK
0x0170-0x0177	Secondary IDE Channel	OK
0x0376-0x0376	Secondary IDE Channel	OK
0x4AEC-0x82E7	PCI bus	OK
0x4B00-0x4BFF	Adaptec AIC-7899 Ultra160/m PCI SCSI Card	OK
0x4C00-0x4CFF	Adaptec AIC-7899 Ultra160/m PCI SCSI Card	OK
0x5000-0x5FFF	DEC 21152 PCI to PCI bridge	OK
0x5100-0x511F	Intel(R) PRO/100+ Dual Port Server Adapter #3	OK
0x5120-0x513F	Intel(R) PRO/100+ Dual Port Server Adapter #4	OK

[IRQs]

IRQ Number	Device
30	Microsoft ACPI-Compliant System
31	S3 Inc. Savage4
27	IBM Netfinity Fault Tolerance PCI Adapter
1	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard
12	PS/2 Compatible Mouse
6	Standard floppy disk controller
4	Communications Port (COM1)
3	Communications Port (COM2)
8	System CMOS/real time clock
13	Numeric data processor
5	Not Available
14	Primary IDE Channel
11	Standard OpenHCD USB Host Controller
28	Adaptec AIC-7899 Ultra160/m PCI SCSI Card
29	Adaptec AIC-7899 Ultra160/m PCI SCSI Card
22	Intel(R) PRO/100+ Dual Port Server Adapter #3
23	Intel(R) PRO/100+ Dual Port Server Adapter #4

[Memory]

Range	Device	Status
0xC8000-0xCFFFF	PCI bus	OK
0xC8000-0xCFFFF	System board	OK
0xF0000000-0xFEBFFFFF	PCI bus	OK

0xF0000000-0xFEBFFFFFFF S3 Inc. Savage4 OK  
 0xFED00000-0xFEDFFFFFFF PCI bus OK  
 0xFE000000-0xFFDFFFFFFF PCI bus OK  
 0xFEB80000-0xFEBFFFFFFF S3 Inc. Savage4 OK  
 0xA0000-0xBFFFFFFF S3 Inc. Savage4 OK  
 0xFEB7FC00-0xFEB7FC1F IBM Netfinity Fault Tolerance PCI Adapter OK  
 0xFEC00000-0xFECFFFFFFF Advanced programmable interrupt controller OK  
 0xFEE00000-0xFEEFFFFFFF Advanced programmable interrupt controller OK  
 0xFF700000-0xFF700FFF Standard OpenHCD USB Host Controller OK  
 0xD0000-0xDFFFF PCI bus OK  
 0x20000000-0xEFFFFFFF PCI bus OK  
 0xEFFF0000-0xEFFFFFFF Adaptec AIC-7899 Ultra160/m PCI SCSI Card OK  
 0xEFFE0000-0xEFFEFFFF Adaptec AIC-7899 Ultra160/m PCI SCSI Card OK  
 0xEE000000-0xEFAFFFFFFF DEC 21152 PCI to PCI bridge OK  
 0xED800000-0xEDFFFFFFF DEC 21152 PCI to PCI bridge OK  
 0xEFA00000-0xEFA00FFF Intel(R) PRO/100+ Dual Port Server Adapter #3 OK  
 0xEF800000-0xEF8FFFFFFF Intel(R) PRO/100+ Dual Port Server Adapter #3 OK  
 0xEFA01000-0xEFA01FFF Intel(R) PRO/100+ Dual Port Server Adapter #4 OK  
 0xEF900000-0xEF9FFFFF Intel(R) PRO/100+ Dual Port Server Adapter #4 OK  
 0x0000-0x9FFFF System board OK  
 0x100000-0xFFFFFFF System board OK  
 0x1000000-0x1FFFFFFF System board OK  
 0xE0000-0xFFFFF System board OK  
 0xFFFE0000-0xFFFFFFF System board OK  
 0xCC000-0xCFFFF System board OK

[Components]

[ Following are sub-categories of this main category ]

[Multimedia]

[ Following are sub-categories of this main category ]

[Audio Codecs]

Codec Version	Manufacturer Size	Description Creation Date	Status	File
c:\winnt\system32\iac25_32.ax	Intel Corporation	Indeo® audio software	OK	C:\WINNT\System32\IAC25_32.AX
2.05.53	195.00 KB (199,680 bytes)			12/7/1999 7:00:00 AM
c:\winnt\system32\msg723.acm	Microsoft Corporation		OK	C:\WINNT\System32\MSG723.ACM
106.77 KB (109,328 bytes)				11/13/2000 10:26:20 AM
c:\winnt\system32\lhacm.acm	Microsoft Corporation		OK	C:\WINNT\System32\LHACM.ACM
33.27 KB (34,064 bytes)				11/13/2000 10:26:21 AM
c:\winnt\system32\tssoft32.acm	DSP GROUP, INC.		OK	C:\WINNT\System32\TSSOFT32.ACM
10.27 KB (10,512 bytes)				12/7/1999 7:00:00 AM
c:\winnt\system32\msgsm32.acm	Microsoft Corporation		OK	C:\WINNT\System32\MSGSM32.ACM
22.27 KB (22,800 bytes)				12/7/1999 7:00:00 AM
c:\winnt\system32\msg711.acm	Microsoft Corporation		OK	C:\WINNT\System32\MSG711.ACM
10.27 KB (10,512 bytes)				12/7/1999 7:00:00 AM
c:\winnt\system32\imaadp32.acm	Microsoft Corporation		OK	C:\WINNT\System32\IMAADP32.ACM

5.00.2134.1 16.27 KB (16,656 bytes) 12/7/1999 7:00:00 AM  
 c:\winnt\system32\msadp32.acm Microsoft Corporation  
 OK C:\WINNT\System32\MSADP32.ACM 5.00.2134.1  
 14.77 KB (15,120 bytes) 12/7/1999 7:00:00 AM

[Video Codecs]

Codec Version	Manufacturer Size	Description Creation Date	Status	File
c:\winnt\system32\ir50_32.dll	Intel Corporation	Indeo® video		
5.10	OK			C:\WINNT\System32\IR50_32.DLL
R.5.10.15.2.55				737.50 KB (755,200 bytes)
				12/7/1999 7:00:00 AM
c:\winnt\system32\msh263.drv	Microsoft Corporation		OK	C:\WINNT\System32\MSH263.DRV
252.27 KB (258,320 bytes)				11/13/2000 10:25:48 AM
c:\winnt\system32\msh261.drv	Microsoft Corporation		OK	C:\WINNT\System32\MSH261.DRV
163.77 KB (167,696 bytes)				11/13/2000 10:26:21 AM
c:\winnt\system32\msvidc32.dll	Microsoft Corporation		OK	C:\WINNT\System32\MSVIDC32.DLL
27.27 KB (27,920 bytes)				12/7/1999 7:00:00 AM
c:\winnt\system32\ir32_32.dll	Intel(R) Corporation		OK	C:\WINNT\System32\IR32_32.DLL
194.50 KB (199,168 bytes)				12/7/1999 7:00:00 AM
c:\winnt\system32\iccvld.dll	Radius Inc.		OK	C:\WINNT\System32\ICCVLD.DLL
(110,592 bytes)				1.10.0.6 108.00 KB
				12/7/1999 7:00:00 AM
c:\winnt\system32\msrle32.dll	Microsoft Corporation		OK	C:\WINNT\System32\MSRLE32.DLL
10.77 KB (11,024 bytes)				5.00.2134.1
				12/7/1999 7:00:00 AM

[CD-ROM]

Item	Value
Drive	D:
Description	CD-ROM Drive
Media Loaded	False
Media Type	CD-ROM
Name	LITEON CD-ROM LTN403
Manufacturer	(Standard CD-ROM drives)
Status	OK
Transfer Rate	Not Available
SCSI Target ID	0
PNP Device ID	IDE\CDROMLITEON_CD-ROM_LTN403_____D
	U26____\5&326853DD&0&0.0.0

[Sound Device]

Item	Value
No sound devices	

[Display]

Item	Value
Name	S3 Inc. Savage4
PNP Device ID	PCI\VEN_5333&DEV_8A22&SUBSYS_01C51014&REV_04\3&267A616A&0&08
Adapter Type	S3 Savage4, S3 compatible
Adapter Description	S3 Inc. Savage4
Adapter RAM	8.00 MB (8,388,608 bytes)
Installed Drivers	s3sav4.dll
Driver Version	5.01.840.0001
INF File	s3sav4.inf (S3Inc section)

Color Planes 1  
Color Table Entries 65536  
Resolution 800 x 600 x 60 hertz  
Bits/Pixel 16

[Infrared]

Item Value  
No infrared devices

[Input]

[ Following are sub-categories of this main category ]

[Keyboard]

Item Value  
Description Standard 101/102-Key or Microsoft Natural  
PS/2 Keyboard  
Name Enhanced (101- or 102-key)  
Layout 00000409  
PNP Device ID ACPI\PNP0303\4\F0B8F99&0  
NumberOfFunctionKeys 12

[Pointing Device]

Item Value  
Hardware Type PS/2 Compatible Mouse  
Number of Buttons 3  
Status OK  
PNP Device ID ACPI\PNP0F13\4\F0B8F99&0  
Power Management Supported False  
Double Click Threshold 6  
Handedness Right Handed Operation

[Modem]

Item Value  
No modems

[Network]

[ Following are sub-categories of this main category ]

[Adapter]

Item Value  
Name [00000000] IBM Netfinity Fault Tolerance PCI Adapter  
Adapter Type Ethernet 802.3  
Product Name IBM Netfinity Fault Tolerance PCI Adapter  
Installed True  
PNP Device ID  
PCI\VEN\_1022&DEV\_2000&SUBSYS\_20001014&REV\_44\3&267  
A616A&0&10  
Last Reset 3/19/2001 9:06:28 AM  
Index 0  
Service Name PCNet5  
IP Address 192.168.127.80  
IP Subnet 255.255.255.0  
Default IP Gateway Not Available  
DHCP Enabled False  
DHCP Server Not Available  
DHCP Lease Expires Not Available  
DHCP Lease Obtained Not Available  
MAC Address 00:06:29:D5:81:D3

Service Name PCNet  
IRQ Number 27  
I/O Port 0x2000-0x201F  
Driver c:\winnt\system32\drivers\pcentn5m.sys (33811, 4.23.00)

Name [00000003] Intel(R) PRO/100+ Dual Port Server Adapter  
Adapter Type Ethernet 802.3  
Product Name Intel(R) PRO/100+ Dual Port Server Adapter  
Installed True  
PNP Device ID  
PCI\VEN\_8086&DEV\_1229&SUBSYS\_10F08086&REV\_05\4&328  
6C9C3&0&2030

Last Reset 3/19/2001 9:06:28 AM

Index 3

Service Name E100B  
IP Address 192.168.80.99  
IP Subnet 255.255.255.0  
Default IP Gateway Not Available  
DHCP Enabled False  
DHCP Server Not Available  
DHCP Lease Expires Not Available  
DHCP Lease Obtained Not Available  
MAC Address 00:90:27:FC:3A:84

Service Name E100B

IRQ Number 22

I/O Port 0x5100-0x511F

Driver c:\winnt\system32\drivers\e100bnt5.sys (80144,  
4.01.67.0000)

Name [00000004] Intel(R) PRO/100+ Dual Port Server Adapter  
Adapter Type Ethernet 802.3  
Product Name Intel(R) PRO/100+ Dual Port Server Adapter  
Installed True  
PNP Device ID

PCI\VEN\_8086&DEV\_1229&SUBSYS\_10F08086&REV\_05\4&328  
6C9C3&0&2830

Last Reset 3/19/2001 9:06:28 AM

Index 4

Service Name E100B  
IP Address 192.168.81.99  
IP Subnet 255.255.255.0  
Default IP Gateway Not Available  
DHCP Enabled False  
DHCP Server Not Available  
DHCP Lease Expires Not Available  
DHCP Lease Obtained Not Available  
MAC Address 00:90:27:FC:3A:85

Service Name E100B

IRQ Number 23

I/O Port 0x5120-0x513F

Driver c:\winnt\system32\drivers\e100bnt5.sys (80144,  
4.01.67.0000)

Name [00000007] RAS Async Adapter  
Adapter Type Not Available  
Product Name RAS Async Adapter  
Installed True

PNP Device ID Not Available

Last Reset 3/19/2001 9:06:28 AM

Index 7

Service Name AsyncMac  
IP Address Not Available  
IP Subnet Not Available  
Default IP Gateway Not Available  
DHCP Enabled False  
DHCP Server Not Available  
DHCP Lease Expires Not Available  
DHCP Lease Obtained Not Available

MAC Address Not Available  
Service Name Not Available

Name [00000008] WAN Miniport (L2TP)  
Adapter Type Not Available  
Product Name WAN Miniport (L2TP)  
Installed True  
PNP Device ID ROOT\MS\_L2TPMINIPOINT\0000  
Last Reset 3/19/2001 9:06:28 AM  
Index 8  
Service Name Rasl2tp  
IP Address Not Available  
IP Subnet Not Available  
Default IP Gateway Not Available  
DHCP Enabled False  
DHCP Server Not Available  
DHCP Lease Expires Not Available  
DHCP Lease Obtained Not Available  
MAC Address Not Available  
Service Name Rasl2tp  
Driver c:\winnt\system32\drivers\rasl2tp.sys (50800, 5.00.2179.1)

Name [00000009] WAN Miniport (PPTP)  
Adapter Type Wide Area Network (WAN)  
Product Name WAN Miniport (PPTP)  
Installed True  
PNP Device ID ROOT\MS\_PPTPMINIPOINT\0000  
Last Reset 3/19/2001 9:06:28 AM  
Index 9  
Service Name PptpMiniport  
IP Address Not Available  
IP Subnet Not Available  
Default IP Gateway Not Available  
DHCP Enabled False  
DHCP Server Not Available  
DHCP Lease Expires Not Available  
DHCP Lease Obtained Not Available  
MAC Address 50:50:54:50:30:30  
Service Name PptpMiniport  
Driver c:\winnt\system32\drivers\raspptp.sys (47856, 5.00.2160.1)

Name [00000010] Direct Parallel  
Adapter Type Not Available  
Product Name Direct Parallel  
Installed True  
PNP Device ID ROOT\MS\_PTIMINIPOINT\0000  
Last Reset 3/19/2001 9:06:28 AM  
Index 10  
Service Name Raspti  
IP Address Not Available  
IP Subnet Not Available  
Default IP Gateway Not Available  
DHCP Enabled False  
DHCP Server Not Available  
DHCP Lease Expires Not Available  
DHCP Lease Obtained Not Available  
MAC Address Not Available  
Service Name Raspti  
Driver c:\winnt\system32\drivers\raspti.sys (16880, 5.00.2146.1)

Name [00000011] WAN Miniport (IP)  
Adapter Type Not Available  
Product Name WAN Miniport (IP)  
Installed True  
PNP Device ID ROOT\MS\_NDISWANIP\0000  
Last Reset 3/19/2001 9:06:28 AM  
Index 11  
Service Name NdisWan

IP Address Not Available  
IP Subnet Not Available  
Default IP Gateway Not Available  
DHCP Enabled False  
DHCP Server Not Available  
DHCP Lease Expires Not Available  
DHCP Lease Obtained Not Available  
MAC Address Not Available  
Service Name NdisWan  
Driver c:\winnt\system32\drivers\ndiswan.sys (90768, 5.00.2184.1)

Name [00000012] IBM 10/100 EtherJet PCI Adapter  
Adapter Type Not Available  
Product Name IBM 10/100 EtherJet PCI Adapter  
Installed True  
PNP Device ID Not Available  
Last Reset 3/19/2001 9:06:28 AM  
Index 12  
Service Name IBMFE  
IP Address Not Available  
IP Subnet Not Available  
Default IP Gateway Not Available  
DHCP Enabled False  
DHCP Server Not Available  
DHCP Lease Expires Not Available  
DHCP Lease Obtained Not Available  
MAC Address Not Available  
Service Name Not Available

[Protocol]

Item	Value
Name	MSAFD Tcpip [TCP/IP]
ConnectionlessService	False
GuaranteesDelivery	True
GuaranteesSequencing	True
MaximumAddressSize	16 bytes
MaximumMessageSize	0 bytes
MessageOriented	False
MinimumAddressSize	16 bytes
PseudoStreamOriented	False
SupportsBroadcasting	False
SupportsConnectData	False
SupportsDisconnectData	False
SupportsEncryption	False
SupportsExpeditedData	True
SupportsGracefulClosing	True
SupportsGuaranteedBandwidth	False
SupportsMulticasting	False
Name	MSAFD Tcpip [UDP/IP]
ConnectionlessService	True
GuaranteesDelivery	False
GuaranteesSequencing	False
MaximumAddressSize	16 bytes
MaximumMessageSize	65467 bytes
MessageOriented	True
MinimumAddressSize	16 bytes
PseudoStreamOriented	False
SupportsBroadcasting	True
SupportsConnectData	False
SupportsDisconnectData	False
SupportsEncryption	False
SupportsExpeditedData	False
SupportsGracefulClosing	False
SupportsGuaranteedBandwidth	False



SupportsMulticasting True

Name RSVP UDP Service Provider  
 ConnectionlessService True  
 GuaranteesDelivery False  
 GuaranteesSequencing False  
 MaximumAddressSize 16 bytes  
 MaximumMessageSize 65467 bytes  
 MessageOriented True  
 MinimumAddressSize 16 bytes  
 PseudoStreamOriented False  
 SupportsBroadcasting True  
 SupportsConnectData False  
 SupportsDisconnectData False  
 SupportsEncryption True  
 SupportsExpeditedData False  
 SupportsGracefulClosing False  
 SupportsGuaranteedBandwidth False  
 SupportsMulticasting True

Name RSVP TCP Service Provider  
 ConnectionlessService False  
 GuaranteesDelivery True  
 GuaranteesSequencing True  
 MaximumAddressSize 16 bytes  
 MaximumMessageSize 0 bytes  
 MessageOriented False  
 MinimumAddressSize 16 bytes  
 PseudoStreamOriented False  
 SupportsBroadcasting False  
 SupportsConnectData False  
 SupportsDisconnectData False  
 SupportsEncryption True  
 SupportsExpeditedData True  
 SupportsGracefulClosing True  
 SupportsGuaranteedBandwidth False  
 SupportsMulticasting False

Name MSAFD NetBIOS  
 [\Device\NetBT\_Tcpip\_{0D94F72B-D8C4-4330-8E93-24430A29C603}] SEQPACKET 9  
 ConnectionlessService False  
 GuaranteesDelivery True  
 GuaranteesSequencing True  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize 20 bytes  
 PseudoStreamOriented False  
 SupportsBroadcasting False  
 SupportsConnectData False  
 SupportsDisconnectData False  
 SupportsEncryption False  
 SupportsExpeditedData False  
 SupportsGracefulClosing False  
 SupportsGuaranteedBandwidth False  
 SupportsMulticasting False

Name MSAFD NetBIOS  
 [\Device\NetBT\_Tcpip\_{0D94F72B-D8C4-4330-8E93-24430A29C603}] DATAGRAM 9  
 ConnectionlessService True  
 GuaranteesDelivery False  
 GuaranteesSequencing False  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes

MessageOriented True  
 MinimumAddressSize 20 bytes  
 PseudoStreamOriented False  
 SupportsBroadcasting True  
 SupportsConnectData False  
 SupportsDisconnectData False  
 SupportsEncryption False  
 SupportsExpeditedData False  
 SupportsGracefulClosing False  
 SupportsGuaranteedBandwidth False  
 SupportsMulticasting False

Name MSAFD NetBIOS  
 [\Device\NetBT\_Tcpip\_{30EDC51D-8BEE-442A-854F-EB2430CB698C}] SEQPACKET 4  
 ConnectionlessService False  
 GuaranteesDelivery True  
 GuaranteesSequencing True  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize 20 bytes  
 PseudoStreamOriented False  
 SupportsBroadcasting False  
 SupportsConnectData False  
 SupportsDisconnectData False  
 SupportsEncryption False  
 SupportsExpeditedData False  
 SupportsGracefulClosing False  
 SupportsGuaranteedBandwidth False  
 SupportsMulticasting False

Name MSAFD NetBIOS  
 [\Device\NetBT\_Tcpip\_{30EDC51D-8BEE-442A-854F-EB2430CB698C}] DATAGRAM 4  
 ConnectionlessService True  
 GuaranteesDelivery False  
 GuaranteesSequencing False  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize 20 bytes  
 PseudoStreamOriented False  
 SupportsBroadcasting True  
 SupportsConnectData False  
 SupportsDisconnectData False  
 SupportsEncryption False  
 SupportsExpeditedData False  
 SupportsGracefulClosing False  
 SupportsGuaranteedBandwidth False  
 SupportsMulticasting False

Name MSAFD NetBIOS  
 [\Device\NetBT\_Tcpip\_{ABE02F8E-D543-4481-9933-28B4E994C913}] SEQPACKET 3  
 ConnectionlessService False  
 GuaranteesDelivery True  
 GuaranteesSequencing True  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize 20 bytes  
 PseudoStreamOriented False  
 SupportsBroadcasting False  
 SupportsConnectData False

SupportsDisconnectData False  
 SupportsEncryption False  
 SupportsExpeditedData False  
 SupportsGracefulClosing False  
 SupportsGuaranteedBandwidth False  
 SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT\_Tcpip\_{ABE02F8E-D543-4481-9933-28B4E994C913}] DATAGRAM 3

ConnectionlessService True  
 GuaranteesDelivery False  
 GuaranteesSequencing False  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize 20 bytes  
 PseudoStreamOriented False  
 SupportsBroadcasting True  
 SupportsConnectData False  
 SupportsDisconnectData False  
 SupportsEncryption False  
 SupportsExpeditedData False  
 SupportsGracefulClosing False  
 SupportsGuaranteedBandwidth False  
 SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT\_Tcpip\_{7163CAD1-68F2-44F1-8D40-FE069C0B852D}] SEQPACKET 0

ConnectionlessService False  
 GuaranteesDelivery True  
 GuaranteesSequencing True  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize 20 bytes  
 PseudoStreamOriented False  
 SupportsBroadcasting False  
 SupportsConnectData False  
 SupportsDisconnectData False  
 SupportsEncryption False  
 SupportsExpeditedData False  
 SupportsGracefulClosing False  
 SupportsGuaranteedBandwidth False  
 SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT\_Tcpip\_{7163CAD1-68F2-44F1-8D40-FE069C0B852D}] DATAGRAM 0

ConnectionlessService True  
 GuaranteesDelivery False  
 GuaranteesSequencing False  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize 20 bytes  
 PseudoStreamOriented False  
 SupportsBroadcasting True  
 SupportsConnectData False  
 SupportsDisconnectData False  
 SupportsEncryption False  
 SupportsExpeditedData False  
 SupportsGracefulClosing False  
 SupportsGuaranteedBandwidth False  
 SupportsMulticasting False

SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT\_Tcpip\_{BBE4B191-4DBC-47A4-8B78-1F7C7BC318DC}] SEQPACKET 7

ConnectionlessService False  
 GuaranteesDelivery True  
 GuaranteesSequencing True  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize 20 bytes  
 PseudoStreamOriented False  
 SupportsBroadcasting False  
 SupportsConnectData False  
 SupportsDisconnectData False  
 SupportsEncryption False  
 SupportsExpeditedData False  
 SupportsGracefulClosing False  
 SupportsGuaranteedBandwidth False  
 SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT\_Tcpip\_{BBE4B191-4DBC-47A4-8B78-1F7C7BC318DC}] DATAGRAM 7

ConnectionlessService True  
 GuaranteesDelivery False  
 GuaranteesSequencing False  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize 20 bytes  
 PseudoStreamOriented False  
 SupportsBroadcasting True  
 SupportsConnectData False  
 SupportsDisconnectData False  
 SupportsEncryption False  
 SupportsExpeditedData False  
 SupportsGracefulClosing False  
 SupportsGuaranteedBandwidth False  
 SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT\_Tcpip\_{90DA37FB-3BB6-4C69-B2FF-A5AD5CB2AC0A}] SEQPACKET 8

ConnectionlessService False  
 GuaranteesDelivery True  
 GuaranteesSequencing True  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize 20 bytes  
 PseudoStreamOriented False  
 SupportsBroadcasting False  
 SupportsConnectData False  
 SupportsDisconnectData False  
 SupportsEncryption False  
 SupportsExpeditedData False  
 SupportsGracefulClosing False  
 SupportsGuaranteedBandwidth False  
 SupportsMulticasting False

Name MSAFD NetBIOS

```

[Device\NetBT_Tcpip_{90DA37FB-3BB6-4C69-B2FF-A5AD5CB2-AC0A}] DATAGRAM8
ConnectionlessService      True
GuaranteesDelivery        False
GuaranteesSequencing      False
MaximumAddressSize        20 bytes
MaximumMessageSize        64000 bytes
MessageOriented           True
MinimumAddressSize        20 bytes
PseudoStreamOriented      False
SupportsBroadcasting      True
SupportsConnectData       False
SupportsDisconnectData    False
SupportsEncryption        False
SupportsExpeditedData     False
SupportsGracefulClosing   False
SupportsGuaranteedBandwidth False
SupportsMulticasting      False

```

[WinSock]

```

Item      Value
File      c:\winnt\system32\winsock.dll
Version   3.10
Size      2.80 KB (2,864 bytes)

File      c:\winnt\system32\wsock32.dll
Version   5.00.2152.1
Size      21.27 KB (21,776 bytes)

```

[Ports]

[ Following are sub-categories of this main category ]

[Serial]

```

Item      Value
Name      COM1
Status    OK
PNP Device ID      ACPI\PNP0501\1
Maximum Input Buffer Size      0
Maximum Output Buffer Size     False
Settable Baud Rate      True
Settable Data Bits      True
Settable Flow Control    True
Settable Parity         True
Settable Parity Check   True
Settable Stop Bits     True
Settable RLS           True
Supports RLS           True
Supports 16 Bit Mode    False
Supports Special Characters  False
Baud Rate 9600
Bits/Byte 8
Stop Bits 1
Parity None
Busy 0
Abort Read/Write on Error      0
Binary Mode Enabled-1         0
Continue Xmit on XOff         0
CTS Outflow Control 0
Discard NULL Bytes 0
DSR Outflow Control 0
DSR Sensitivity 0
DTR Flow Control Type      Enable
EOF Character 0

```

```

Error Replace Character      0
Error Replacement Enabled    0
Event Character 0
Parity Check Enabled 0
RTS Flow Control Type      Enable
XOff Character 19
XOffXmit Threshold 512
XOn Character 17
XOnXmit Threshold 2048
XOnXOff InFlow Control      0
XOnXOff OutFlow Control     0
IRQ Number 4
I/O Port 0x03F8-0x03FF
Driver c:\winnt\system32\drivers\serial.sys (62448, 5.00.2134.1)

```

```

Name      COM2
Status    OK
PNP Device ID      ACPI\PNP0501\2
Maximum Input Buffer Size      0
Maximum Output Buffer Size     False
Settable Baud Rate      True
Settable Data Bits      True
Settable Flow Control    True
Settable Parity         True
Settable Parity Check   True
Settable Stop Bits     True
Settable RLS           True
Supports RLS           True
Supports 16 Bit Mode    False
Supports Special Characters  False
Baud Rate 9600
Bits/Byte 8
Stop Bits 1
Parity None
Busy 0

```

```

Abort Read/Write on Error      0
Binary Mode Enabled-1         0
Continue Xmit on XOff         0
CTS Outflow Control 0
Discard NULL Bytes 0
DSR Outflow Control 0
DSR Sensitivity 0
DTR Flow Control Type      Enable
EOF Character 0
Error Replace Character      0
Error Replacement Enabled    0
Event Character 0
Parity Check Enabled 0
RTS Flow Control Type      Enable
XOff Character 19
XOffXmit Threshold 512
XOn Character 17
XOnXmit Threshold 2048
XOnXOff InFlow Control      0
XOnXOff OutFlow Control     0
IRQ Number 3
I/O Port 0x02F8-0x02FF
Driver c:\winnt\system32\drivers\serial.sys (62448, 5.00.2134.1)

```

[Parallel]

```

Item      Value
Name      LPT1
PNP Device ID      ACPI\PNP0400\1

```

[Storage]

[ Following are sub-categories of this main category ]

[Drives]

Item Value  
 Drive A:  
 Description 3 1/2 Inch Floppy Drive

Drive C:  
 Description Local Fixed Disk  
 Compressed False  
 File System NTFS  
 Size 8.47 GB (9,097,125,888 bytes)  
 Free Space 6.89 GB (7,398,551,552 bytes)  
 Volume Name  
 Volume Serial Number 10FFC016  
 Partition Disk #0, Partition #0  
 Partition Size 8.47 GB (9,097,127,424 bytes)  
 Starting Offset 32256 bytes  
 Drive Description Disk drive  
 Drive Manufacturer (Standard disk drives)  
 Drive Model IBM-PSG DMVS09D !# SCSI Disk Device  
 Drive BytesPerSector 512  
 Drive MediaLoaded True  
 Drive MediaType Fixed hard disk media  
 Drive Partitions 1  
 Drive SCSI Bus 0  
 Drive SCSI Logical Unit 0  
 Drive SCSI Port 2  
 Drive SCSI Target ID 0  
 Drive SectorsPerTrack 63  
 Drive Size 9097159680 bytes  
 Drive TotalCylinders 1106  
 Drive TotalSectors 17767890  
 Drive TotalTracks 282030  
 Drive TracksPerCylinder 255

[SCSI]

Item Value  
 Name Adaptec AIC-7899 Ultra160/m PCI SCSI Card  
 Caption Adaptec AIC-7899 Ultra160/m PCI SCSI Card  
 Driver adpu160m  
 Status OK  
 PNP Device ID  
 PCI\VEN\_9005&DEV\_00CF&SUBSYS\_019D1014&REV\_01\3&13  
 C0B0C5&0&18  
 Device ID  
 PCI\VEN\_9005&DEV\_00CF&SUBSYS\_019D1014&REV\_01\3&13  
 C0B0C5&0&18  
 Device Map Not Available  
 Index Not Available  
 Max Number Controlled Not Available  
 IRQ Number 28  
 I/O Port 0x4B00-0x4BFF  
 Driver c:\winnt\system32\drivers\adpu160m.sys (64432, v3.10a)

Name Adaptec AIC-7899 Ultra160/m PCI SCSI Card  
 Caption Adaptec AIC-7899 Ultra160/m PCI SCSI Card  
 Driver adpu160m  
 Status OK  
 PNP Device ID  
 PCI\VEN\_9005&DEV\_00CF&SUBSYS\_019D1014&REV\_01\3&13  
 C0B0C5&0&19

Device ID  
 PCI\VEN\_9005&DEV\_00CF&SUBSYS\_019D1014&REV\_01\3&13  
 C0B0C5&0&19  
 Device Map Not Available  
 Index Not Available  
 Max Number Controlled Not Available  
 IRQ Number 29  
 I/O Port 0x4C00-0x4CFF  
 Driver c:\winnt\system32\drivers\adpu160m.sys (64432, v3.10a)

[Printing]

Name Port Name Server Name  
 No printing information

[Problem Devices]

Device	PNP Device ID	Error Code
Not Available	ACPI\IBM37C0\4&F0B8F99&0 28	

[USB]

Device	PNP Device ID
Standard OpenHCD USB Host Controller	PCI\VEN_1166&DEV_0220&SUBSYS_02201166&REV_04\3&267 A616A&0&7A
USB Root Hub	USB\ROOT_HUB\4&372644EA&0

[Software Environment]

[ Following are sub-categories of this main category ]

[Drivers]

Name	Description	File	Type	Started
Start Mode	State	Status	Error Control	
Accept Pause	Accept Stop			
abiosdsk	Abiosdsk	Not Available	Kernel Driver	
False	Disabled	Stopped	OK	Ignore
False				False
abp480n5	abp480n5	Not Available	Kernel Driver	
False	Disabled	Stopped	OK	Normal
False				False
acpi	Microsoft ACPI Driver			
c:\winnt\system32\drivers\acpi.sys			Kernel Driver	
True	Boot	Running	OK	Normal
True				False
acpiec	ACPIEC	c:\winnt\system32\drivers\acpiec.sys	Kernel Driver	
False	False	Disabled	Stopped	OK
Normal	False	False		
adpu160m	adpu160m	c:\winnt\system32\drivers\adpu160m.sys	Kernel Driver	
True	Boot	Running	OK	Normal
False	True			
afd	AFD Networking Support Environment			
c:\winnt\system32\drivers\afd.sys	Kernel Driver		Kernel Driver	True
Auto	Running	OK	Normal	False
True				True
aha154x	Aha154x	Not Available	Kernel Driver	
False	Disabled	Stopped	OK	Normal
False				False
aic116x	aic116x	Not Available	Kernel Driver	
False	Disabled	Stopped	OK	Normal
False				False
aic78u2	aic78u2	Not Available	Kernel Driver	
False	Disabled	Stopped	OK	Normal
False				False

aic78xx	aic78xx	Not Available	Kernel Driver	cpqfw2e	cpqfw2e	Not Available	Kernel Driver
False	Disabled	Stopped	Normal	False	Disabled	Stopped	Normal
False		OK	False	False			False
ami0nt	ami0nt	Not Available	Kernel Driver	dac960nt	dac960nt	Not Available	Kernel Driver
False	Disabled	Stopped	Normal	False	Disabled	Stopped	Normal
False		OK	False	False			False
amsint	amsint	Not Available	Kernel Driver	deckzpsx	deckzpsx	Not Available	Kernel Driver
False	Disabled	Stopped	Normal	False	Disabled	Stopped	Normal
False		OK	False	False			False
asc	asc	Not Available	Kernel Driver	dfsdriver	DfsDriver	c:\winnt\system32\drivers\dfs.sys	File System
False	Disabled	Stopped	Normal	Driver	True	Boot	Running
False		OK	False	False	True		OK
asc3350p	asc3350p	Not Available	Kernel Driver	disk	Disk Driver	c:\winnt\system32\drivers\disk.sys	Kernel Driver
False	Disabled	Stopped	Normal	Kernel Driver	True	Boot	Running
False		OK	False	Normal	False	True	OK
asc3550	asc3550	Not Available	Kernel Driver	diskperf	Diskperf	c:\winnt\system32\drivers\diskperf.sys	Kernel Driver
False	Disabled	Stopped	Normal	Kernel Driver	True	Boot	Running
False		OK	False	Normal	False	True	OK
asynctac	RAS Asynchronous Media Driver			dmboot	dmboot	c:\winnt\system32\drivers\dmboot.sys	Kernel Driver
c:\winnt\system32\drivers\asynctac.sys			Kernel Driver	Kernel Driver	False	Disabled	Stopped
False	Manual	Stopped	Normal	Normal	False	False	OK
False			False	dmio	Logical Disk Manager Driver		
atapi	Standard IDE/ESDI Hard Disk Controller			c:\winnt\system32\drivers\dmio.sys			Kernel Driver
c:\winnt\system32\drivers\atapi.sys			Kernel Driver	True	Boot	Running	OK
True	Boot	Running	Normal	True			False
True			False	dmload	dmload	c:\winnt\system32\drivers\dmload.sys	Kernel Driver
atdisk	Atdisk	Not Available	Kernel Driver	Kernel Driver	True	Boot	Running
False	Disabled	Stopped	Ignore	Normal	False	True	OK
False			False	e100b	Intel PRO Adapter Driver		
atmarpc	ATM ARP Client Protocol			c:\winnt\system32\drivers\e100bnt5.sys			Kernel Driver
c:\winnt\system32\drivers\atmarpc.sys			Kernel Driver	True	Manual	Running	OK
False	Manual	Stopped	Normal	True			False
False			False	efs	EFS	c:\winnt\system32\drivers\efs.sys	File System
audstub	Audio Stub Driver			Driver	True	Disabled	Running
c:\winnt\system32\drivers\audstub.sys			Kernel Driver	False	True		OK
True	Manual	Running	Normal	fastfat	Fastfat	c:\winnt\system32\drivers\fastfat.sys	File
True			False	System Driver	True	Disabled	Running
beep	Beep	c:\winnt\system32\drivers\beep.sys		Normal	False	True	OK
Kernel Driver	True	System	Running	fd16_700	Fd16_700	Not Available	Kernel Driver
Normal	False	True	OK	False	Disabled	Stopped	Normal
buslogic	BusLogic	Not Available	Kernel Driver	False			False
False	Disabled	Stopped	Normal	fdc	Floppy Disk Controller Driver		
False			False	c:\winnt\system32\drivers\fdc.sys			Kernel Driver
cd20xrnt	cd20xrnt	Not Available	Kernel Driver	Manual	Running	OK	Normal
False	Disabled	Stopped	Normal	fireport	fireport	Not Available	Kernel Driver
False		OK	False	False	Disabled	Stopped	Normal
cdaudio	Cdaudio	c:\winnt\system32\drivers\cdaudio.sys		False			False
Kernel Driver	False	System	Stopped	flashpnt	flashpnt	Not Available	Kernel Driver
Ignore	False	False	OK	False	Disabled	Stopped	Normal
cdfs	Cdfs	c:\winnt\system32\drivers\cdfs.sys	File	False			False
System Driver	True	Disabled	Running	flpydisk	Floppy Disk Driver		
Normal	False	True	OK	c:\winnt\system32\drivers\flpydisk.sys			Kernel Driver
cdrom	CD-ROM Driver	c:\winnt\system32\drivers\cdrom.sys		True	Manual	Running	OK
Kernel Driver	True	System	Running	True			False
Normal	False	True	OK	ftdisk	Volume Manager Driver		
changer	Changer	Not Available	Kernel Driver	c:\winnt\system32\drivers\ftdisk.sys			Kernel Driver
False	System	Stopped	Ignore	True	Boot	Running	OK
False			False	True			False
cpqarray	Cpqarray	Not Available	Kernel Driver	gpc	Generic Packet Classifier		
False	Disabled	Stopped	Normal	c:\winnt\system32\drivers\msgpc.sys			Kernel Driver
False			False	True	Manual	Running	OK
cpqarray2	cpqarray2	Not Available	Kernel Driver	True			Normal
False	Disabled	Stopped	Normal	i8042prt	i8042 Keyboard and PS/2 Mouse Port Driver		
False			False	c:\winnt\system32\drivers\i8042prt.sys			Kernel Driver
cpqfcalm	cpqfcalm	Not Available	Kernel Driver	True	System	Running	OK
False	Disabled	Stopped	Normal	True			Normal
False			False	ibmf	IBM 10/100 Ethernet PCI Adapter NT Driver		
				c:\winnt\system32\drivers\ibmfent5.sys			Kernel Driver

False	Manual	Stopped	OK	Normal	False	mspclock	Microsoft Streaming Clock Proxy				
False						c:\winnt\system32\drivers\mspclock.sys	Kernel Driver				
ini910u	ini910u	Not Available		Kernel Driver		False	Manual	Stopped	OK	Normal	False
False	Disabled	Stopped	OK	Normal	False	False					
False						mspqm	Microsoft Streaming Quality Manager Proxy				
intelide	IntelIde	Not Available		Kernel Driver		c:\winnt\system32\drivers\mspqm.sys	Kernel Driver				
False	Disabled	Stopped	OK	Normal	False	False	Manual	Stopped	OK	Normal	False
False						False					
ipfilterdriver	IP Traffic Filter Driver					mup	Mup	c:\winnt\system32\drivers\mup.sys			File
c:\winnt\system32\drivers\ipfltdrv.sys				Kernel Driver		System Driver	True	Boot	Running	OK	
False	Manual	Stopped	OK	Normal	False	Normal	False	True			
False						nrcr710	Nrcr710	Not Available		Kernel Driver	
ipinip	IP in IP Tunnel Driver			Kernel Driver		False	Disabled	Stopped	OK	Normal	False
Kernel Driver	False	Manual	Stopped	OK		False					
Normal	False	False				ndis	NDIS System Driver	c:\winnt\system32\drivers\ndis.sys			
ipnat	IP Network Address Translator			Kernel Driver		Kernel Driver	True	Boot	Running	OK	
c:\winnt\system32\drivers\ipnat.sys				Kernel Driver		Normal	False	True			
False	Manual	Stopped	OK	Normal	False	ndistapi	Remote Access NDIS TAPI Driver				
False						c:\winnt\system32\drivers\ndistapi.sys	Kernel Driver				
ipsec	IPSEC driver			Kernel Driver		True	Manual	Running	OK	Normal	False
Kernel Driver	True	Manual	Running	OK		True					
Normal	False	True				ndiswan	Remote Access NDIS WAN Driver				
ipsraidn	ipsraidn	Not Available		Kernel Driver		c:\winnt\system32\drivers\ndiswan.sys	Kernel Driver				
False	Disabled	Stopped	OK	Normal	False	True	Manual	Running	OK	Normal	False
False						True					
isapnp	PnP ISA/EISA Bus Driver			Kernel Driver		ndproxy	NDIS Proxy				
c:\winnt\system32\drivers\isapnp.sys				Kernel Driver		c:\winnt\system32\drivers\ndproxy.sys	Kernel Driver				
True	Boot	Running	OK	Critical	False	True	Manual	Running	OK	Normal	False
True						True					
kbdclass	Keyboard Class Driver			Kernel Driver		netbios	NetBIOS Interface	c:\winnt\system32\drivers\netbios.sys			
c:\winnt\system32\drivers\kbdclass.sys				Kernel Driver		File System Driver	True	System	Running	OK	
True	System	Running	OK	Normal	False	Normal	False	True			
True						netbt	NetBios over Tcpip	c:\winnt\system32\drivers\netbt.sys			
ksecdd	KSecDD	c:\winnt\system32\drivers\ksecdd.sys		Kernel Driver		Kernel Driver	True	System	Running	OK	
Kernel Driver	True	Boot	Running	OK		Normal	False	True			
Normal	False	True				netdetect	NetDetect	c:\winnt\system32\drivers\netdetect.sys			
lbrtfdc	lbrtfdc	Not Available		Kernel Driver		Kernel Driver	False	Manual	Stopped	OK	
False	System	Stopped	OK	Ignore	False	Normal	False	False			
False						npfs	Npfs	c:\winnt\system32\drivers\npfs.sys			File
lp6nds35	lp6nds35	Not Available		Kernel Driver		System Driver	True	System	Running	OK	
False	Disabled	Stopped	OK	Normal	False	Normal	False	True			
False						ntfs	Ntfs	c:\winnt\system32\drivers\ntfs.sys			File
mnmdd	mnmdd	c:\winnt\system32\drivers\mnmdd.sys		Kernel Driver		System Driver	True	Disabled	Running	OK	
Kernel Driver	True	System	Running	OK		Normal	False	True			
Ignore	False	True				null	Null	c:\winnt\system32\drivers\null.sys			
modem	Modem	c:\winnt\system32\drivers\modem.sys		Kernel Driver		Kernel Driver	True	System	Running	OK	
Kernel Driver	False	Manual	Stopped	OK		Normal	False	True			
Ignore	False	False				nwlnkflt	IPX Traffic Filter Driver				
mouclass	Mouse Class Driver			Kernel Driver		c:\winnt\system32\drivers\nwlnkflt.sys	Kernel Driver				
c:\winnt\system32\drivers\mouclass.sys				Kernel Driver		False	Manual	Stopped	OK	Normal	False
True	System	Running	OK	Normal	False	False					
True						nwlnkfld	IPX Traffic Forwarder Driver				
mountmgr	MountMgr	c:\winnt\system32\drivers\mountmgr.sys		Kernel Driver		c:\winnt\system32\drivers\nwlnkfld.sys	Kernel Driver				
Kernel Driver	True	Boot	Running	OK		False	Manual	Stopped	OK	Normal	False
Normal	False	True				False					
mraid35x	mraid35x	Not Available		Kernel Driver		openhci	Microsoft USB Open Host Controller Driver				
False	Disabled	Stopped	OK	Normal	False	c:\winnt\system32\drivers\openhci.sys	Kernel Driver				
False						True	Manual	Running	OK	Normal	False
False						True					
mrx smb	MRXSMB	c:\winnt\system32\drivers\mrx smb.sys		System Driver		File	True				
System Driver	True	System	Running	OK		parallel	Parallel class driver	c:\winnt\system32\drivers\parallel.sys			
Normal	False	True				Kernel Driver	True	Manual	Running	OK	
msfs	Msfs	c:\winnt\system32\drivers\msfs.sys		System Driver		File	Normal	False	True		
System Driver	True	System	Running	OK		parport	Parallel port driver	c:\winnt\system32\drivers\parport.sys			
Normal	False	True				Kernel Driver	True	System	Running	OK	
mskssrv	Microsoft Streaming Service Proxy			Kernel Driver		Ignore	False	True			
c:\winnt\system32\drivers\mskssrv.sys				Kernel Driver		partmgr	PartMgr	c:\winnt\system32\drivers\partmgr.sys			
False	Manual	Stopped	OK	Normal	False	Kernel Driver	True	Boot	Running	OK	
False						Normal	False	True			

parvdm	ParVdm	c:\winnt\system32\drivers\parvdm.sys							
Kernel Driver	True	Auto	Running	OK					
Ignore	False	True							
pci	PCI Bus Driver	c:\winnt\system32\drivers\pci.sys							
Kernel Driver	True	Boot	Running	OK					
Critical	False	True							
pcidump	PCIDump	Not Available		Kernel Driver					
False	System	Stopped	OK	Ignore	False				
pciide	PCIIde	c:\winnt\system32\drivers\pciide.sys							
Kernel Driver	True	Boot	Running	OK					
Normal	False	True							
pcmcia	Pcmcia	c:\winnt\system32\drivers\pcmcia.sys							
Kernel Driver	False	Disabled	Stopped	OK					
Normal	False	False							
pcnet	AMD PCNET Compatible Adapter Driver								
c:\winnt\system32\drivers\pcntn5m.sys				Kernel Driver					
True	Manual	Running	OK	Normal	False				
True									
pdcomp	PDCOMP	Not Available		Kernel Driver					
False	Manual	Stopped	OK	Ignore	False				
False									
pdframe	PDFRAME	Not Available		Kernel Driver					
False	Manual	Stopped	OK	Ignore	False				
False									
pdreli	PDRELI	Not Available		Kernel Driver					
False	Manual	Stopped	OK	Ignore	False				
False									
pdrframe	PDRFRAME	Not Available		Kernel Driver					
False	Manual	Stopped	OK	Ignore	False				
False									
pptpminiport	WAN Miniport (PPTP)								
c:\winnt\system32\drivers\raspptp.sys				Kernel Driver					
True	Manual	Running	OK	Normal	False				
True									
ptilink	Direct Parallel Link Driver								
c:\winnt\system32\drivers\ptilink.sys				Kernel Driver					
True	Manual	Running	OK	Normal	False				
True									
ql1080	ql1080	Not Available		Kernel Driver					
False	Disabled	Stopped	OK	Normal	False				
False									
ql10wnt	Ql10wnt	Not Available		Kernel Driver					
False	Disabled	Stopped	OK	Normal	False				
False									
ql1240	ql1240	Not Available		Kernel Driver					
False	Disabled	Stopped	OK	Normal	False				
False									
ql2100	ql2100	Not Available		Kernel Driver					
False	Disabled	Stopped	OK	Normal	False				
False									
rasacd	Remote Access Auto Connection Driver								
c:\winnt\system32\drivers\rasacd.sys				Kernel Driver					
True	System	Running	OK	Normal	False				
True									
rasl2tp	WAN Miniport (L2TP)								
c:\winnt\system32\drivers\rasl2tp.sys				Kernel Driver					
True	Manual	Running	OK	Normal	False				
True									
raspti	Direct Parallel	c:\winnt\system32\drivers\raspti.sys							
Kernel Driver	True	Manual	Running	OK					
Normal	False	True							
rca	Microsoft Streaming Network Raw Channel Access								
c:\winnt\system32\drivers\rca.sys				Kernel Driver	False				
Manual	Stopped	OK	Normal	False	False				
rdbss	Rdbss	c:\winnt\system32\drivers\rdbss.sys							File
System Driver	True	System	Running	OK					
Normal	False	True							
rdpdr	Terminal Server Device Redirector Driver								
c:\winnt\system32\drivers\rdpdr.sys				Kernel Driver					
True	Manual	Running	OK	Normal	False				
True									
rdpwd	RDPWD	c:\winnt\system32\drivers\rdpwd.sys							
Kernel Driver	True	Manual	Running	OK					
Ignore	False	True							
redbook	Digital CD Audio Playback Filter Driver								
c:\winnt\system32\drivers\redbook.sys				Kernel Driver					
False	System	Stopped	OK	Normal	False				
False									
s3inc	S3Inc	c:\winnt\system32\drivers\s3sav4m.sys							
Kernel Driver	True	Manual	Running	OK					
Ignore	False	True							
serenum	Serenum Filter Driver								
c:\winnt\system32\drivers\serenum.sys				Kernel Driver					
True	Manual	Running	OK	Normal	False				
True									
serial	Serial port driver	c:\winnt\system32\drivers\serial.sys							
Kernel Driver	True	System	Running	OK					
Ignore	False	True							
sfloppy	Sfloppy	c:\winnt\system32\drivers\sfloppy.sys							
Kernel Driver	False	System	Stopped	OK					
Ignore	False	False							
sglfb	sglfb	Not Available		Kernel Driver					
False	System	Stopped	OK	Normal	False				
False									
simbad	Simbad	Not Available		Kernel Driver					
False	Disabled	Stopped	OK	Normal	False				
False									
sparrow	Sparrow	Not Available		Kernel Driver					
False	Disabled	Stopped	OK	Normal	False				
False									
spud	Special Purpose Utility Driver								
c:\winnt\system32\drivers\spud.sys				Kernel Driver					
True	Manual	Running	OK	Normal	False				
True									
srv	Srv	c:\winnt\system32\drivers\srv.sys		File System					
Driver	True	Manual	Running	OK	Normal				
False	True								
swenum	Software Bus Driver								
c:\winnt\system32\drivers\swenum.sys				Kernel Driver					
True	Manual	Running	OK	Normal	False				
True									
symc810	symc810	Not Available		Kernel Driver					
False	Disabled	Stopped	OK	Normal	False				
False									
symc8xx	symc8xx	Not Available		Kernel Driver					
False	Disabled	Stopped	OK	Normal	False				
False									
sym_hi	sym_hi	Not Available		Kernel Driver					
False	Disabled	Stopped	OK	Normal	False				
False									
tcpip	TCP/IP Protocol Driver								
c:\winnt\system32\drivers\tcpip.sys				Kernel Driver					
True	System	Running	OK	Normal	False				
True									
tdasync	TDASYNC	c:\winnt\system32\drivers\tdasync.sys							
Kernel Driver	False	Manual	Stopped	OK					
Ignore	False	False							
tdipx	TDIPX	c:\winnt\system32\drivers\tdipx.sys							
Kernel Driver	False	Manual	Stopped	OK					
Ignore	False	False							
tdnetb	TDNETB	c:\winnt\system32\drivers\tdnetb.sys							
Kernel Driver	False	Manual	Stopped	OK					
Ignore	False	False							

```

tdpipe  TDIPIPE  c:\winnt\system32\drivers\tdpipe.sys
Kernel Driver  False   Manual  Stopped  OK
Ignore  False  False
tdspix  TDSPIX  c:\winnt\system32\drivers\tdspix.sys
Kernel Driver  False   Manual  Stopped  OK
Ignore  False  False
tdtcp   TDTTCP  c:\winnt\system32\drivers\tdtcp.sys
Kernel Driver  True    Manual  Running  OK
Ignore  False  True
termdd  Terminal Device Driver
c:\winnt\system32\drivers\termdd.sys  Kernel Driver
True  Auto  Running  OK  Normal  False
True
tga     tga     Not Available  Kernel Driver
False  System  Stopped  OK  Ignore  False
False
udfs    Udfs    c:\winnt\system32\drivers\udfs.sys
System Driver  False   Disabled  Stopped  OK
Normal  False  False
ultra66  ultra66  Not Available  Kernel Driver
False  Disabled  Stopped  OK  Normal  False
False
update  Microcode Update Driver
c:\winnt\system32\drivers\update.sys  Kernel Driver
True  Manual  Running  OK  Normal  False
True
usbhub  Microsoft USB Standard Hub Driver
c:\winnt\system32\drivers\usbhub.sys  Kernel Driver
True  Manual  Running  OK  Normal  False
True
vgasave  VgaSave  c:\winnt\system32\drivers\vga.sys
Kernel Driver  True    System  Running  OK
Ignore  False  True
wanarp   Remote Access IP ARP Driver
c:\winnt\system32\drivers\wanarp.sys  Kernel Driver
True  Manual  Running  OK  Normal  False
True
wdica   WDICA  Not Available  Kernel Driver
False  Manual  Stopped  OK  Ignore  False
False

```

[Environment Variables]

```

Variable  Value  User Name
ComSpec  %SystemRoot%\system32\cmd.exe  <SYSTEM>
NUMBER_OF_PROCESSORS  2  <SYSTEM>
OS  Windows_NT  <SYSTEM>
Os2LibPath  %SystemRoot%\system32\os2\dll;
<SYSTEM>
Path
%SystemRoot%\system32;%SystemRoot%;%SystemRoot%\System32\
Wbem;c:\nttools\C:\Program Files\Microsoft SQL
Server\80\Tools\BINN  <SYSTEM>
PATHEXT
.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH
<SYSTEM>
PROCESSOR_ARCHITECTURE  x86  <SYSTEM>
PROCESSOR_IDENTIFIER  x86 Family 6 Model 8 Stepping 3,
GenuineIntel  <SYSTEM>
PROCESSOR_LEVEL  6  <SYSTEM>
PROCESSOR_REVISION  0803  <SYSTEM>
TEMP  %SystemRoot%\TEMP  <SYSTEM>
TMP  %SystemRoot%\TEMP  <SYSTEM>
windir  %SystemRoot%  <SYSTEM>
TEMP  %USERPROFILE%\Local Settings\Temp
RTCLIENT64\tpcc
TMP  %USERPROFILE%\Local Settings\Temp
RTCLIENT64\tpcc

```

```

TEMP  %USERPROFILE%\Local Settings\Temp
RTCLIENT64\Administrator
TMP  %USERPROFILE%\Local Settings\Temp
RTCLIENT64\Administrator

```

[Jobs]

[ Following are sub-categories of this main category ]

[Print]

Document Size	Owner	Notify	Status	Time
Submitted	Start Time	Until Time	Elapsed Time	Pages Printed
Job ID	Priority	Parameters	Driver Name	Print Processor
Host Print Queue	Data Type	Name		
No print jobs				

File

[Network Connections]

Local Name	Remote Name	Type	Status
User Name			
No network connections information			

[Running Tasks]

Name	Path	Process ID	Priority	Min Working Set	Max Working Set	Start Time	Version	Size	File Date
system	idle process	Not Available		0	0				Not
Available	Not Available			Not Available					Unknown
Unknown	Unknown								
system	Not Available	8	8	0					
1413120	Not Available	Unknown	Unknown	Unknown	Unknown				
smss.exe	c:\winnt\system32\smss.exe	180	11						
204800	1413120	3/19/2001 2:06:47 PM		5.00.2170.1					
44.27 KB (45,328 bytes)		12/7/1999 7:00:00 AM							
csrss.exe	Not Available	204	13	Not Available					
Not Available		3/19/2001 2:06:50 PM		Unknown					
Unknown	Unknown								
winlogon.exe	c:\winnt\system32\winlogon.exe	200	13						
204800	1413120	3/19/2001 2:06:51 PM		5.00.2182.1					
173.27 KB (177,424 bytes)		12/7/1999 7:00:00 AM							
services.exe	c:\winnt\system32\services.exe	256	9						
204800	1413120	3/19/2001 2:06:52 PM		5.00.2134.1					
86.77 KB (88,848 bytes)		12/7/1999 7:00:00 AM							
lsass.exe	c:\winnt\system32\lsass.exe	268	13						
204800	1413120	3/19/2001 2:06:52 PM		5.00.2184.1					
32.77 KB (33,552 bytes)		12/7/1999 7:00:00 AM							
svchost.exe	c:\winnt\system32\svchost.exe	440	8						
204800	1413120	3/19/2001 2:06:55 PM		5.00.2134.1					
7.77 KB (7,952 bytes)		12/7/1999 7:00:00 AM							
spoolsv.exe	c:\winnt\system32\spoolsv.exe	456	8						
204800	1413120	3/19/2001 2:06:55 PM		5.00.2161.1					
43.77 KB (44,816 bytes)		11/13/2000 6:01:25 AM							
msdtc.exe	c:\winnt\system32\msdtc.exe	484	8						
204800	1413120	3/19/2001 2:06:55 PM		1999.9.3421.3					
6.77 KB (6,928 bytes)		11/13/2000 6:23:37 AM							
svchost.exe	c:\winnt\system32\svchost.exe	608	8						
204800	1413120	3/19/2001 2:06:57 PM		5.00.2134.1					
7.77 KB (7,952 bytes)		12/7/1999 7:00:00 AM							
llssrv.exe	c:\winnt\system32\llssrv.exe	632	9						
204800	1413120	3/19/2001 2:06:57 PM		5.00.2167.1					
114.27 KB (117,008 bytes)		12/7/1999 7:00:00 AM							
regsvc.exe	c:\winnt\system32\regsvc.exe	696	8						
204800	1413120	3/19/2001 2:06:57 PM		5.00.2155.1					
65.27 KB (66,832 bytes)		12/7/1999 7:00:00 AM							



```

rsys.exe c:\benchcraft\rsys.exe 712 8 204800
1413120 3/19/2001 2:06:57 PM Not Available
32.00 KB (32,768 bytes) 9/8/2000 3:47:26 PM
rshsvc.exe c:\winnt\system32\rshsvc.exe 740 8
204800 1413120 3/19/2001 2:06:58 PM Not Available
53.50 KB (54,784 bytes) 11/21/2000 5:07:53 PM
mstask.exec c:\winnt\system32\mstask.exe 756 8
204800 1413120 3/19/2001 2:06:58 PM 4.71.2137.1
115.27 KB (118,032 bytes) 11/13/2000 10:26:08 AM
explorer.exe c:\winnt\explorer.exe 680 8
204800 1413120 3/19/2001 2:07:06 PM
5.00.2920.0000 232.77 KB (238,352 bytes) 12/7/1999
7:00:00 AM
tcpsvcs.exe c:\winnt\system32\tcpsvcs.exe 976 8
204800 1413120 3/19/2001 2:07:07 PM 5.00.2134.1
24.77 KB (25,360 bytes) 12/7/1999 7:00:00 AM
termsrv.exe c:\winnt\system32\termsrv.exe 1004 10
204800 1413120 3/19/2001 2:07:07 PM 5.00.2182.1
136.77 KB (140,048 bytes) 11/13/2000 6:23:41 AM
winmgmt.exe c:\winnt\system32\wbem\winmgmt.exe
1036 8 204800 1413120 3/19/2001 2:07:08 PM
1.50.1085.0001 188.05 KB (192,567 bytes) 12/7/1999
7:00:00 AM
inetinfo.exe c:\winnt\system32\inetrv\inetinfo.exe
1064 8 204800 1413120 3/19/2001 2:07:08 PM
5.00.0984 14.27 KB (14,608 bytes) 11/27/2000 1:51:25 PM
dfssvc.exe c:\winnt\system32\dfssvc.exe 1188 8
204800 1413120 3/19/2001 2:07:09 PM 5.00.2191.1
85.27 KB (87,312 bytes) 12/7/1999 7:00:00 AM
mmc.exe c:\winnt\system32\mmc.exe 1268 8
204800 1413120 3/19/2001 2:07:19 PM 5.00.2153.1
589.27 KB (603,408 bytes) 12/7/1999 7:00:00 AM
svchost.exe c:\winnt\system32\svchost.exe 1484 8
204800 1413120 3/19/2001 2:07:43 PM 5.00.2134.1
7.77 KB (7,952 bytes) 12/7/1999 7:00:00 AM
rsvp.exe c:\winnt\system32\rsvp.exe 1324 8
204800 1413120 3/19/2001 2:08:44 PM 5.00.2167.1
172.77 KB (176,912 bytes) 12/7/1999 7:00:00 AM

```

[Loaded Modules]

```

Name Version Size File Date Manufacturer Path
traffic.dll 5.00.2139.1 30.77 KB (31,504 bytes)
12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\traffic.dll
rsvp.exe 5.00.2167.1 172.77 KB (176,912 bytes)
12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\rsvp.exe
tapisrv.dll 5.00.2186.1 168.77 KB (172,816 bytes)
12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\tapisrv.dll
wbemprox.dll 1.50.1085.0001 40.05 KB (41,016 bytes)
12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\wbem\wbemprox.dll
olepro32.dll 5.0.4512 160.27 KB (164,112 bytes)
12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\olepro32.dll
dmocx.dll 5.00.2134.1 23.27 KB (23,824 bytes)
12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\dmocx.dll
mlang.dll 5.00.2920.0000 510.77 KB (523,024 bytes)
12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\mlang.dll
rassapi.dll 5.00.2188.1 14.27 KB (14,608 bytes)
12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\rassapi.dll

```

```

adsnt.dll 5.00.2191.1 194.27 KB (198,928 bytes)
12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\adsnt.dll
dbghelp.dll 5.00.2195.1 159.27 KB (163,088
bytes) 12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\dbghelp.dll
localsec.dll 5.00.2134.1 227.27 KB (232,720
bytes) 12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\localsec.dll
devmgr.dll 5.00.2166.1 215.77 KB (220,944 bytes)
12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\devmgr.dll
filemgmt.dll 5.00.2134.1 287.27 KB (294,160
bytes) 12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\filemgmt.dll
pdh.dll 5.00.2174.1 143.27 KB (146,704 bytes)
12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\pdh.dll
smlogcfg.dll 5.00.2163.1 273.27 KB (279,824
bytes) 12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\smlogcfg.dll
cabinet.dll 5.00.2147.1 54.77 KB (56,080 bytes)
12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\cabinet.dll
msinfo32.dll 5.00.2177.1 312.27 KB (319,760
bytes) 11/13/2000 10:26:16 AM Microsoft Corporation
c:\program files\common files\microsoft shared\msinfo\msinfo32.dll
riched20.dll 5.30.23.1200 421.27 KB (431,376
bytes) 12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\riched20.dll
riched32.dll 5.00.2134.1 3.77 KB (3,856 bytes)
12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\riched32.dll
els.dll 5.00.2175.1 151.27 KB (154,896 bytes)
12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\els.dll
ntmsmgr.dll 1.0,0,1 427.77 KB (438,032 bytes)
12/7/1999 7:00:00 AM Microsoft Corporation and
HighGround Systems, Inc. c:\winnt\system32\ntmsmgr.dll
mmfutil.dll 1.50.1085.0000 32.06 KB (32,829 bytes)
12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\mmfutil.dll
logdrive.dll 1.50.1085.0000 200.06 KB (204,863
bytes) 12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\logdrive.dll
dfrgres.dll 5.00.2150.1 27.50 KB (28,160 bytes)
12/7/1999 7:00:00 AM Executive Software International, Inc.
c:\winnt\system32\dfrgres.dll
dfrgsnap.dll 5.00.2150.1 41.77 KB (42,768 bytes)
12/7/1999 7:00:00 AM Executive Software International, Inc.
c:\winnt\system32\dfrgsnap.dll
dmdskres.dll 2191.1.296.2 119.00 KB (121,856
bytes) 12/7/1999 7:00:00 AM Microsoft Corp.,
VERITAS Software c:\winnt\system32\dmdskres.dll
dmutil.dll 2191.1.296.2 41.77 KB (42,768 bytes)
12/7/1999 7:00:00 AM VERITAS Software Corp.
c:\winnt\system32\dmutil.dll
ntmsapi.dll 5.00.1948.1 50.27 KB (51,472 bytes)
12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\ntmsapi.dll
dmdskmgr.dll 2191.1.296.2 158.77 KB (162,576
bytes) 12/7/1999 7:00:00 AM Microsoft Corp.,
VERITAS Software c:\winnt\system32\dmdskmgr.dll
mycomput.dll 5.00.2134.1 107.77 KB (110,352
bytes) 12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\mycomput.dll

```

mmcndmgr.dll	5.00.2178.1	815.27 KB (834,832 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	fcachdll.dll	5.00.0984	43.77 KB (44,816 bytes)	11/27/2000 1:52:55 PM	Microsoft Corporation
c:\winnt\system32\mmcndmgr.dll					c:\winnt\system32\fcachdll.dll				
mmc.exe	5.00.2153.1	589.27 KB (603,408 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	iisfecnv.dll	5.00.0984	7.27 KB (7,440 bytes)	11/27/2000 1:51:25 PM	Microsoft Corporation
c:\winnt\system32\mmc.exe					c:\winnt\system32\iisfecnv.dll				
dfssvc.exe	5.00.2191.1	85.27 KB (87,312 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	isatq.dll	5.00.0984	61.27 KB (62,736 bytes)	11/27/2000 1:51:27 PM	Microsoft Corporation
c:\winnt\system32\dfssvc.exe					c:\winnt\system32\isatq.dll				
mailmsg.dll	5.00.0984	64.27 KB (65,808 bytes)	11/27/2000 1:52:55 PM	Microsoft Corporation	infocomm.dll	5.00.0984	234.27 KB (239,888 bytes)	11/27/2000 1:51:25 PM	Microsoft Corporation
c:\winnt\system32\inetsrv\mailmsg.dll					c:\winnt\system32\inetsrv\infocomm.dll				
ntfsdrv.dll	5.00.0984	36.77 KB (37,648 bytes)	11/27/2000 1:52:56 PM	Microsoft Corporation	smtpsvc.dll	5.00.0984	406.27 KB (416,016 bytes)	11/27/2000 1:52:57 PM	Microsoft Corporation
c:\winnt\system32\inetsrv\ntfsdrv.dll					c:\winnt\system32\inetsrv\smtpsvc.dll				
iislog.dll	5.00.0984	76.27 KB (78,096 bytes)	11/27/2000 1:51:25 PM	Microsoft Corporation	security.dll	5.00.2154.1	5.77 KB (5,904 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\inetsrv\iislog.dll					c:\winnt\system32\security.dll				
httpext.dll	0.9.3939.9	418.27 KB (428,304 bytes)	11/27/2000 1:51:24 PM	Microsoft Corporation	svcxext.dll	5.00.0984	39.77 KB (40,720 bytes)	11/27/2000 1:51:26 PM	Microsoft Corporation
c:\winnt\system32\inetsrv\httpext.dll					c:\winnt\system32\inetsrv\svcxext.dll				
rpcproxy.dll	5.00.2176.1	16.27 KB (16,656 bytes)	11/27/2000 1:51:23 PM	Microsoft Corporation	admexs.dll	5.00.0984	27.77 KB (28,432 bytes)	11/27/2000 1:51:24 PM	Microsoft Corporation
c:\winnt\system32\rpcproxy\rpcproxy.dll					c:\winnt\system32\inetsrv\admexs.dll				
fpexedll.dll	4.0.2.3406	20.06 KB (20,541 bytes)	11/13/2000 6:26:42 AM	Microsoft Corporation	wamreg.dll	5.00.0984	46.27 KB (47,376 bytes)	11/27/2000 1:51:32 PM	Microsoft Corporation
c:\program files\common files\microsoft shared\web server extensions\40\bin\fpexedll.dll					c:\winnt\system32\inetsrv\wamreg.dll				
md5filt.dll	5.00.0984	32.77 KB (33,552 bytes)	11/27/2000 1:51:31 PM	Microsoft Corporation	metadata.dll	5.00.0984	70.77 KB (72,464 bytes)	11/27/2000 1:51:25 PM	Microsoft Corporation
c:\winnt\system32\inetsrv\md5filt.dll					c:\winnt\system32\inetsrv\metadata.dll				
aqueue.dll	5.00.0984	260.27 KB (266,512 bytes)	11/13/2000 6:26:35 AM	Microsoft Corporation	iismap.dll	5.00.0984	56.27 KB (57,616 bytes)	11/27/2000 1:51:26 PM	Microsoft Corporation
c:\winnt\system32\inetsrv\aqueue.dll					c:\winnt\system32\iismap.dll				
gzip.dll	5.00.0984	30.27 KB (30,992 bytes)	11/27/2000 1:51:30 PM	Microsoft Corporation	nsepm.dll	5.00.0984	43.27 KB (44,304 bytes)	11/27/2000 1:51:25 PM	Microsoft Corporation
c:\winnt\system32\inetsrv\gzip.dll					c:\winnt\system32\inetsrv\nsepm.dll				
compfilt.dll	5.00.0984	22.27 KB (22,800 bytes)	11/27/2000 1:51:30 PM	Microsoft Corporation	coadmin.dll	5.00.0984	39.77 KB (40,720 bytes)	11/27/2000 1:51:27 PM	Microsoft Corporation
c:\winnt\system32\inetsrv\compfilt.dll					c:\winnt\system32\inetsrv\coadmin.dll				
seo.dll	5.00.0984	230.77 KB (236,304 bytes)	11/13/2000 6:26:36 AM	Microsoft Corporation	iisadmin.dll	5.00.0984	14.77 KB (15,120 bytes)	11/27/2000 1:51:24 PM	Microsoft Corporation
c:\winnt\system32\inetsrv\seo.dll					c:\winnt\system32\inetsrv\iisadmin.dll				
sspifilt.dll	5.00.0984	43.27 KB (44,304 bytes)	11/27/2000 1:51:32 PM	Microsoft Corporation	rpcref.dll	5.00.0984	4.27 KB (4,368 bytes)	11/27/2000 1:51:25 PM	Microsoft Corporation
c:\winnt\system32\inetsrv\sspifilt.dll					c:\winnt\system32\inetsrv\rpcref.dll				
iscomlog.dll	5.00.0984	24.77 KB (25,360 bytes)	11/27/2000 1:51:25 PM	Microsoft Corporation	iisrtl.dll	5.00.0984	120.77 KB (123,664 bytes)	11/27/2000 1:51:26 PM	Microsoft Corporation
c:\winnt\system32\inetsrv\iscomlog.dll					c:\winnt\system32\iisrtl.dll				
lonsint.dll	5.00.0984	11.77 KB (12,048 bytes)	11/27/2000 1:51:25 PM	Microsoft Corporation	inetinfo.exe	5.00.0984	14.27 KB (14,608 bytes)	11/27/2000 1:51:25 PM	Microsoft Corporation
c:\winnt\system32\inetsrv\lonsint.dll					c:\winnt\system32\inetsrv\inetinfo.exe				
inetsloc.dll	5.00.0984	20.27 KB (20,752 bytes)	11/27/2000 1:51:27 PM	Microsoft Corporation	wshnetbs.dll	5.00.2134.1	7.77 KB (7,952 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\inetsloc.dll					c:\winnt\system32\wshnetbs.dll				
w3svc.dll	5.00.0984	347.27 KB (355,600 bytes)	11/27/2000 1:51:32 PM	Microsoft Corporation	rapilib.dll	5.00.2167.1	25.27 KB (25,872 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\inetsrv\w3svc.dll					c:\winnt\system32\rapilib.dll				
staxmem.dll	5.00.0984	8.27 KB (8,464 bytes)	11/27/2000 1:51:24 PM	Microsoft Corporation	rsvsp.dll	5.00.2167.1	74.77 KB (76,560 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\staxmem.dll					c:\winnt\system32\rsvsp.dll				
extrace.dll	5.00.0984	13.77 KB (14,096 bytes)	11/27/2000 1:51:26 PM	Microsoft Corporation	ntmarta.dll	5.00.2158.1	98.77 KB (101,136 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\extrace.dll					c:\winnt\system32\ntmarta.dll				
rwnh.dll	5.00.0984	10.77 KB (11,024 bytes)	11/13/2000 6:26:36 AM	Microsoft Corporation	provthrd.dll	1.50.1085.0000	68.07 KB (69,708 bytes)	11/13/2000 10:26:07 AM	Microsoft Corporation
c:\winnt\system32\rwnh.dll					c:\winnt\system32\wbem\provthrd.dll				
					ntevt.dll	1.50.1085.0000	192.06 KB (196,669 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
					c:\winnt\system32\wbem\ntevt.dll				

perfos.dll 5.00.2155.1 21.27 KB (21,776 bytes)  
12/7/1999 7:00:00 AM Microsoft Corporation  
c:\winnt\system32\perfos.dll  
psapi.dll 5.00.2134.1 28.27 KB (28,944 bytes)  
12/7/1999 7:00:00 AM Microsoft Corporation  
c:\winnt\system32\psapi.dll  
framedyn.dll 1.50.1085.0000 164.05 KB (167,992 bytes)  
12/7/1999 7:00:00 AM Microsoft Corporation  
c:\winnt\system32\wbem\framedyn.dll  
cimwin32.dll 1.50.1085.0000 1.03 MB (1,077,306 bytes)  
12/7/1999 7:00:00 AM Microsoft Corporation  
c:\winnt\system32\wbem\cimwin32.dll  
wbemsvc.dll 1.50.1085.0000 140.07 KB (143,430 bytes)  
12/7/1999 7:00:00 AM Microsoft Corporation  
c:\winnt\system32\wbem\wbemsvc.dll  
wbemess.dll 1.50.1085.0001 352.05 KB (360,503 bytes)  
12/7/1999 7:00:00 AM Microsoft Corporation  
c:\winnt\system32\wbem\wbemess.dll  
fastprox.dll 1.50.1085.0001 144.08 KB (147,534 bytes)  
12/7/1999 7:00:00 AM Microsoft Corporation  
c:\winnt\system32\wbem\fastprox.dll  
wbemcore.dll 1.50.1085.0001 632.05 KB (647,224 bytes)  
12/7/1999 7:00:00 AM Microsoft Corporation  
c:\winnt\system32\wbem\wbemcore.dll  
wbemcomn.dll 1.50.1085.0001 684.05 KB (700,472 bytes)  
12/7/1999 7:00:00 AM Microsoft Corporation  
c:\winnt\system32\wbem\wbemcomn.dll  
winmgmt.exe 1.50.1085.0001 188.05 KB (192,567 bytes)  
12/7/1999 7:00:00 AM Microsoft Corporation  
c:\winnt\system32\wbem\winmgmt.exe  
rdpwsx.dll 5.00.2180.1 94.40 KB (96,664 bytes)  
11/13/2000 6:23:40 AM Microsoft Corporation  
c:\winnt\system32\rdpwsx.dll  
ntlsapi.dll 5.00.2134.1 6.77 KB (6,928 bytes)12/7/1999  
7:00:00 AM Microsoft Corporation  
c:\winnt\system32\ntlsapi.dll  
mstlsapi.dll 5.00.2181.1 24.77 KB (25,360 bytes)  
12/7/1999 7:00:00 AM Microsoft Corporation  
c:\winnt\system32\mstlsapi.dll  
icaapi.dll 5.00.2134.1 118.77 KB (121,616 bytes)  
11/13/2000 6:23:39 AM Microsoft Corporation  
c:\winnt\system32\icaapi.dll  
regapi.dll 5.00.2155.1 35.27 KB (36,112 bytes)  
12/7/1999 7:00:00 AM Microsoft Corporation  
c:\winnt\system32\regapi.dll  
termsrv.exe 5.00.2182.1 136.77 KB (140,048 bytes)  
11/13/2000 6:23:41 AM Microsoft Corporation  
c:\winnt\system32\termsrv.exe  
simptcp.dll 5.00.2134.1 19.27 KB (19,728 bytes)  
11/13/2000 6:23:31 AM Microsoft Corporation  
c:\winnt\system32\simptcp.dll  
tcpsvcs.exe 5.00.2134.1 24.77 KB (25,360 bytes)  
12/7/1999 7:00:00 AM Microsoft Corporation  
c:\winnt\system32\tcpsvcs.exe  
wininet.dll 5.00.2920.0000 456.77 KB (467,728 bytes)  
12/7/1999 7:00:00 AM Microsoft Corporation  
c:\winnt\system32\wininet.dll  
urlmon.dll 5.00.2920.0000 426.77 KB (437,008 bytes)  
12/7/1999 7:00:00 AM Microsoft Corporation  
c:\winnt\system32\urlmon.dll  
faxshell.dll 5.00.2134.1 8.27 KB (8,464 bytes)12/7/1999  
7:00:00 AM Microsoft Corporation  
c:\winnt\system32\faxshell.dll  
msacm32.dll 5.00.2134.1 65.27 KB (66,832 bytes)  
12/7/1999 7:00:00 AM Microsoft Corporation  
c:\winnt\system32\msacm32.dll

avifil32.dll 5.00.2134.1 76.27 KB (78,096 bytes)  
12/7/1999 7:00:00 AM Microsoft Corporation  
c:\winnt\system32\avifil32.dll  
msvfw32.dll 5.00.2134.1 113.77 KB (116,496 bytes)  
12/7/1999 7:00:00 AM Microsoft Corporation  
c:\winnt\system32\msvfw32.dll  
docprop2.dll 5.00.2178.1 297.77 KB (304,912 bytes)  
12/7/1999 7:00:00 AM Microsoft Corporation  
c:\winnt\system32\docprop2.dll  
linkinfo.dll 5.00.2134.1 15.77 KB (16,144 bytes)  
12/7/1999 7:00:00 AM Microsoft Corporation  
c:\winnt\system32\linkinfo.dll  
msi.dll 1.10.1029.0 1.71 MB (1,794,320 bytes)  
12/7/1999 7:00:00 AM Microsoft Corporation  
c:\winnt\system32\msi.dll  
powrprof.dll 5.00.2920.0000 13.27 KB (13,584 bytes)  
12/7/1999 7:00:00 AM Microsoft Corporation  
c:\winnt\system32\powrprof.dll  
batmeter.dll 5.00.2920.0000 20.27 KB (20,752 bytes)  
12/7/1999 7:00:00 AM Microsoft Corporation  
c:\winnt\system32\batmeter.dll  
stobject.dll 5.00.2144.1 81.77 KB (83,728 bytes)  
12/7/1999 7:00:00 AM Microsoft Corporation  
c:\winnt\system32\stobject.dll  
webcheck.dll 5.00.2920.0000 251.77 KB (257,808 bytes)  
12/7/1999 7:00:00 AM Microsoft Corporation  
c:\winnt\system32\webcheck.dll  
browsec.dll 5.00.2920.0000 34.50 KB (35,328 bytes)  
12/7/1999 7:00:00 AM Microsoft Corporation  
c:\winnt\system32\browsec.dll  
netui1.dll 5.00.2134.1 210.27 KB (215,312 bytes)  
12/7/1999 7:00:00 AM Microsoft Corporation  
c:\winnt\system32\netui1.dll  
netui0.dll 5.00.2134.1 70.27 KB (71,952 bytes)  
12/7/1999 7:00:00 AM Microsoft Corporation  
c:\winnt\system32\netui0.dll  
ntlanman.dll 5.00.2157.1 35.27 KB (36,112 bytes)  
12/7/1999 7:00:00 AM Microsoft Corporation  
c:\winnt\system32\ntlanman.dll  
ntshrui.dll 5.00.2134.1 46.77 KB (47,888 bytes)  
12/7/1999 7:00:00 AM Microsoft Corporation  
c:\winnt\system32\ntshrui.dll  
mydocs.dll 5.00.2920.0000 55.77 KB (57,104 bytes)  
12/7/1999 7:00:00 AM Microsoft Corporation  
c:\winnt\system32\mydocs.dll  
hhsetup.dll 4.74.8702 66.27 KB (67,856 bytes) 12/7/1999  
7:00:00 AM Microsoft Corporation  
c:\winnt\system32\hhsetup.dll  
mmshext.dll 5.00.2153.1 24.27 KB (24,848 bytes)  
12/7/1999 7:00:00 AM Microsoft Corporation  
c:\winnt\system32\mmshext.dll  
browseui.dll 5.00.2920.0000 793.27 KB (812,304 bytes)  
12/7/1999 7:00:00 AM Microsoft Corporation  
c:\winnt\system32\browseui.dll  
shdocvw.dll 5.00.2920.0000 1.05 MB (1,104,144 bytes)  
12/7/1999 7:00:00 AM Microsoft Corporation  
c:\winnt\system32\shdocvw.dll  
explorer.exe 5.00.2920.0000 232.77 KB (238,352 bytes)  
12/7/1999 7:00:00 AM Microsoft Corporation  
c:\winnt\explorer.exe  
msidle.dll 5.00.2920.0000 6.27 KB (6,416 bytes)12/7/1999  
7:00:00 AM Microsoft Corporation  
c:\winnt\system32\msidle.dll  
mstask.exe 4.71.2137.1 115.27 KB (118,032 bytes)  
11/13/2000 10:26:08 AM Microsoft Corporation  
c:\winnt\system32\mstask.exe

rshsvc.dll	Not Available	18.00 KB (18,432 bytes)		msdart.dll	2.60.6526.0	144.27 KB (147,728 bytes)	
11/21/2000 5:07:53 PM	Not Available			1/3/2001 4:23:48 PM	Microsoft Corporation		
c:\winnt\system32\rshsvc.dll				c:\winnt\system32\msdart.dll			
rshsvc.exe	Not Available	53.50 KB (54,784 bytes)		oledb32.dll	2.60.6526.0	448.27 KB (459,024 bytes)	
11/21/2000 5:07:53 PM	Not Available			1/3/2001 4:23:49 PM	Microsoft Corporation		
c:\winnt\system32\rshsvc.exe				c:\program files\common files\system\ole db\oledb32.dll			
rsys.exe	Not Available	32.00 KB (32,768 bytes)		msjint40.dll	4.00.2927.2	148.27 KB (151,824 bytes)	
9/8/2000 3:47:26 PM	Not Available		c:\benchcraft\rsys.exe	12/7/1999 7:00:00 AM	Microsoft Corporation		
regsvcs.exe	5.00.2155.1	65.27 KB (66,832 bytes)		c:\winnt\system32\msjint40.dll			
12/7/1999 7:00:00 AM			Microsoft Corporation	msjter40.dll	4.00.2927.2	52.27 KB (53,520 bytes)	
c:\winnt\system32\regsvcs.exe				12/7/1999 7:00:00 AM	Microsoft Corporation		
llsrpc.dll	5.00.2149.1	45.77 KB (46,864 bytes)		c:\winnt\system32\msjter40.dll			
12/7/1999 7:00:00 AM			Microsoft Corporation	mswstr10.dll	4.00.2927.10	600.27 KB (614,672 bytes)	
c:\winnt\system32\llsrpc.dll				12/7/1999 7:00:00 AM	Microsoft Corporation		
llssrv.exe	5.00.2167.1	114.27 KB (117,008 bytes)		c:\winnt\system32\mswstr10.dll			
12/7/1999 7:00:00 AM			Microsoft Corporation	msjet40.dll	4.00.2927.4	1.43 MB (1,495,312 bytes)	
c:\winnt\system32\llssrv.exe				12/7/1999 7:00:00 AM	Microsoft Corporation		
ntmsdba.dll	5.00.2187.1	167.77 KB (171,792 bytes)		c:\winnt\system32\msjet40.dll			
12/7/1999 7:00:00 AM			Microsoft Corporation	msjetoledb40.dll	4.00.2927.2	340.27 KB (348,432 bytes)	
c:\winnt\system32\ntmsdba.dll				12/7/1999 7:00:00 AM	Microsoft Corporation		
rasdlg.dll	5.00.2194.1	514.27 KB (526,608 bytes)		c:\winnt\system32\msjetoledb40.dll			
12/7/1999 7:00:00 AM			Microsoft Corporation	iasrad.dll	5.00.2139.1	94.27 KB (96,528 bytes)	
c:\winnt\system32\rasdlg.dll				12/7/1999 7:00:00 AM	Microsoft Corporation		
netcfgx.dll	5.00.2175.1	533.77 KB (546,576 bytes)		c:\winnt\system32\iasrad.dll			
12/7/1999 7:00:00 AM			Microsoft Corporation	iasam.dll	5.00.2160.1	96.27 KB (98,576 bytes)	
c:\winnt\system32\netcfgx.dll				12/7/1999 7:00:00 AM	Microsoft Corporation		
rasmans.dll	5.00.2188.1	146.77 KB (150,288 bytes)		c:\winnt\system32\iasam.dll			
12/7/1999 7:00:00 AM			Microsoft Corporation	iasads.dll	5.00.2134.1	73.77 KB (75,536 bytes)	
c:\winnt\system32\rasmans.dll				12/7/1999 7:00:00 AM	Microsoft Corporation		
netshell.dll	5.00.2176.1	456.77 KB (467,728 bytes)		c:\winnt\system32\iasads.dll			
12/7/1999 7:00:00 AM			Microsoft Corporation	iaspolicy.dll	5.00.2134.1	25.27 KB (25,872 bytes)	
c:\winnt\system32\netshell.dll				12/7/1999 7:00:00 AM	Microsoft Corporation		
netman.dll	5.00.2175.1	88.77 KB (90,896 bytes)		c:\winnt\system32\iaspolicy.dll			
12/7/1999 7:00:00 AM			Microsoft Corporation	iasvcs.dll	5.00.2160.1	58.77 KB (60,176 bytes)	
c:\winnt\system32\netman.dll				12/7/1999 7:00:00 AM	Microsoft Corporation		
sens.dll	5.00.2163.1	36.77 KB (37,648 bytes)		c:\winnt\system32\iasvcs.dll			
12/7/1999 7:00:00 AM			Microsoft Corporation	iasdo.dll	5.00.2157.1	262.27 KB (268,560 bytes)	
c:\winnt\system32\sens.dll				12/7/1999 7:00:00 AM	Microsoft Corporation		
iaslpr.dll	5.00.2184.1	33.27 KB (34,064 bytes)		c:\winnt\system32\iasdo.dll			
12/7/1999 7:00:00 AM			Microsoft Corporation	ntmssvc.dll	5.00.2187.1	390.77 KB (400,144 bytes)	
c:\winnt\system32\iaslpr.dll				12/7/1999 7:00:00 AM	Microsoft Corporation		
iasacct.dll	5.00.2134.1	28.27 KB (28,944 bytes)		c:\winnt\system32\ntmssvc.dll			
12/7/1999 7:00:00 AM			Microsoft Corporation	ias.dll	5.00.2134.1	7.27 KB (7,440 bytes)	12/7/1999
c:\winnt\system32\iasacct.dll				7:00:00 AM	Microsoft Corporation		
iasuser.dll	5.00.2134.1	25.77 KB (26,384 bytes)		c:\winnt\system32\ias.dll			
12/7/1999 7:00:00 AM			Microsoft Corporation	es.dll	1999.9.3422.21	231.77 KB (237,328 bytes)	
c:\winnt\system32\iasuser.dll				12/7/1999 7:00:00 AM	Microsoft Corporation		
iasnap.dll	5.00.2134.1	58.77 KB (60,176 bytes)		c:\winnt\system32\es.dll			
12/7/1999 7:00:00 AM			Microsoft Corporation	mtxoci.dll	1999.9.3421.3	109.27 KB (111,888 bytes)	
c:\winnt\system32\iasnap.dll				11/13/2000 6:23:38 AM	Microsoft Corporation		
iaspipe.dll	5.00.2134.1	41.77 KB (42,768 bytes)		c:\winnt\system32\mtxoci.dll			
12/7/1999 7:00:00 AM			Microsoft Corporation	resutils.dll	5.00.2191.1	39.77 KB (40,720 bytes)	
c:\winnt\system32\iaspipe.dll				12/7/1999 7:00:00 AM	Microsoft Corporation		
expsrv.dll	6.0.8540	370.27 KB (379,152 bytes)	12/7/1999	c:\winnt\system32\resutils.dll			
7:00:00 AM			Microsoft Corporation	clusapi.dll	5.00.2179.1	50.27 KB (51,472 bytes)	
c:\winnt\system32\expsrv.dll				12/7/1999 7:00:00 AM	Microsoft Corporation		
vbajet32.dll	6.1.8268	30.27 KB (30,992 bytes)		c:\winnt\system32\clusapi.dll			
12/7/1999 7:00:00 AM			Microsoft Corporation	msvcp50.dll	5.00.7051	552.50 KB (565,760 bytes)	
c:\winnt\system32\vbajet32.dll				12/7/1999 7:00:00 AM	Microsoft Corporation		
msjtes40.dll	4.00.2927.8	232.27 KB (237,840 bytes)		c:\winnt\system32\msvcp50.dll			
12/7/1999 7:00:00 AM			Microsoft Corporation	xolehlp.dll	1999.9.3421.3	17.27 KB (17,680 bytes)	
c:\winnt\system32\msjtes40.dll				11/13/2000 6:23:38 AM	Microsoft Corporation		
oledb32r.dll	2.60.6526.0	68.27 KB (69,904 bytes)		c:\winnt\system32\xolehlp.dll			
1/3/2001 4:23:49 PM	Microsoft Corporation		c:\program	msdtclog.dll	1999.9.3421.3	89.77 KB (91,920 bytes)	
files\common files\system\ole db\oledb32r.dll				11/13/2000 6:23:37 AM	Microsoft Corporation		
comdlg32.dll	5.00.2920.0000	236.77 KB (242,448 bytes)		c:\winnt\system32\msdtclog.dll			
12/7/1999 7:00:00 AM			Microsoft Corporation				
c:\winnt\system32\comdlg32.dll							

mtxclu.dll	1999.9.3421.3	50.27 KB (51,472 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\mtxclu.dll		
msdtcprx.dll	1999.9.3422.10	619.27 KB (634,128 bytes)
11/13/2000 6:23:38 AM		Microsoft Corporation
c:\winnt\system32\msdtcprx.dll		
txfaux.dll	1999.9.3422.24	341.27 KB (349,456 bytes)
11/13/2000 6:23:37 AM		Microsoft Corporation
c:\winnt\system32\txfaux.dll		
msdtctm.dll	1999.9.3422.12	1.02 MB (1,070,864 bytes)
11/13/2000 6:23:38 AM		Microsoft Corporation
c:\winnt\system32\msdtctm.dll		
msdtc.exe	1999.9.3421.3	6.77 KB (6,928 bytes)
11/13/2000 6:23:37 AM		Microsoft Corporation
c:\winnt\system32\msdtc.exe		
wmi.dll	5.00.2191.1	6.27 KB (6,416 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\wmi.dll		
admwprox.dll	5.00.0984	31.77 KB (32,528 bytes)
11/27/2000 1:51:26 PM		Microsoft Corporation
c:\winnt\system32\admwprox.dll		
inetpp.dll	5.00.2161.1	63.27 KB (64,784 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\inetpp.dll		
win32spl.dll	5.00.2162.1	92.27 KB (94,480 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\win32spl.dll		
usbmon.dll	5.00.2165.1	11.27 KB (11,536 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\usbmon.dll		
tcpmon.dll	5.00.2165.1	40.77 KB (41,744 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\tcpmon.dll		
pjlmon.dll	5.00.2165.1	12.77 KB (13,072 bytes)
11/30/1999 6:39:36 PM		Microsoft Corporation
c:\winnt\system32\pjlmon.dll		
cnbjmon.dll	5.00.2134.1	43.77 KB (44,816 bytes)
11/30/1999 6:38:48 PM		Microsoft Corporation
c:\winnt\system32\cnbjmon.dll		
localspl.dll	5.00.2191.1	244.77 KB (250,640 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\localspl.dll		
spoolss.dll	5.00.2161.1	61.77 KB (63,248 bytes)
11/13/2000 6:01:25 AM		Microsoft Corporation
c:\winnt\system32\spoolss.dll		
spoolsv.exe	5.00.2161.1	43.77 KB (44,816 bytes)
11/13/2000 6:01:25 AM		Microsoft Corporation
c:\winnt\system32\spoolsv.exe		
rpsess.dll	5.00.2181.1	229.27 KB (234,768 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\rpsess.dll		
svchost.exe	5.00.2134.1	7.77 KB (7,952 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\svchost.exe		
dssbase.dll	5.00.2150.1	140.77 KB (144,144 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\dssbase.dll		
oakley.dll	5.00.2174.1	420.27 KB (430,352 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\oakley.dll		
mfc42u.dll	6.00.8665.0	972.05 KB (995,384 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\mfc42u.dll		
polagent.dll	5.00.2183.1	108.27 KB (110,864 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\polagent.dll		

scecli.dll	5.00.2191.1	105.27 KB (107,792 bytes)
12/7/1999 7:00:00 AM		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		Microsoft Corporation
c:\winnt\system32\atl.dll		
certcli.dll	5.00.2175.1	132.27 KB (135,440 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\certcli.dll		
esent.dll	6.0.3939.6	1.07 MB (1,120,016 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\esent.dll		
ntdsatq.dll	5.00.2181.1	31.27 KB (32,016 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\ntdsatq.dll		
ntdsad.dll	5.00.2195.1	993.27 KB (1,017,104 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\ntdsad.dll		
kdcsvc.dll	5.00.2181.1	133.77 KB (136,976 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\kdcsvc.dll		
sfmapi.dll	5.00.2134.1	38.77 KB (39,696 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\sfmapi.dll		
rassfm.dll	5.00.2168.1	21.27 KB (21,776 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\rassfm.dll		
mpr.dll	5.00.2146.1	53.27 KB (54,544 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\mpr.dll		
schannel.dll	5.00.2170.1	139.77 KB (143,120 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\schannel.dll		
netlogon.dll	5.00.2182.1	347.77 KB (356,112 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\netlogon.dll		
msv1_0.dll	5.00.2164.1	94.77 KB (97,040 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\msv1_0.dll		
kerberos.dll	5.00.2181.1	196.77 KB (201,488 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\kerberos.dll		
msprivs.dll	5.00.2154.1	41.50 KB (42,496 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\msprivs.dll		
samsrv.dll	5.00.2192.1	357.77 KB (366,352 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\samsrv.dll		
lsasrv.dll	5.00.2184.1	487.77 KB (499,472 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\lsasrv.dll		
lsass.exe	5.00.2184.1	32.77 KB (33,552 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\lsass.exe		
wmicore.dll	5.00.2178.1	70.77 KB (72,464 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\wmicore.dll		
rasadhlp.dll	5.00.2168.1	7.27 KB (7,440 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\rasadhlp.dll		
winmr.dll	5.00.2160.1	18.77 KB (19,216 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\winmr.dll		
rn20.dll	5.00.2152.1	35.77 KB (36,624 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\rn20.dll		

msock.dll	5.00.2152.1	62.27 KB (63,760 bytes)	adslrpc.dll	5.00.2172.1	127.77 KB (130,832 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation	12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\msock.dll			c:\winnt\system32\adslrpc.dll		
msgsvcs.dll	5.00.2181.1	33.77 KB (34,576 bytes)	activeds.dll	5.00.2172.1	172.77 KB (176,912 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation	12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\msgsvcs.dll			c:\winnt\system32\activeds.dll		
xactsrv.dll	5.00.2134.1	90.27 KB (92,432 bytes)	mprapi.dll	5.00.2181.1	79.27 KB (81,168 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation	12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\xactsrv.dll			c:\winnt\system32\mprapi.dll		
browser.dll	5.00.2142.1	48.27 KB (49,424 bytes)	iphlpapi.dll	5.00.2173.2	67.77 KB (69,392 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation	12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\browser.dll			c:\winnt\system32\iphlpapi.dll		
alrsvcs.dll	5.00.2134.1	17.77 KB (18,192 bytes)	icmp.dll	5.00.2134.1	7.27 KB (7,440 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation	12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\alrsvcs.dll			c:\winnt\system32\icmp.dll		
trkwks.dll	5.00.2166.1	88.77 KB (90,896 bytes)	dhcpcsvc.dll	5.00.2153.1	88.77 KB (90,896 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation	12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\trkwks.dll			c:\winnt\system32\dhcpcsvc.dll		
seclogon.dll	5.00.2135.1	15.77 KB (16,144 bytes)	eventlog.dll	5.00.2178.1	43.77 KB (44,816 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation	12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\seclogon.dll			c:\winnt\system32\eventlog.dll		
wshtcpip.dll	5.00.2134.1	17.27 KB (17,680 bytes)	ntdsapi.dll	5.00.2160.1	56.27 KB (57,616 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation	12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\wshtcpip.dll			c:\winnt\system32\ntdsapi.dll		
msafd.dll	5.00.2153.1	54.27 KB (55,568 bytes)	scesrv.dll	5.00.2188.1	225.77 KB (231,184 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation	12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\msafd.dll			c:\winnt\system32\scesrv.dll		
psbase.dll	5.00.2146.1	111.77 KB (114,448 bytes)	umpnrmgr.dll	5.00.2182.1	86.27 KB (88,336 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation	12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\psbase.dll			c:\winnt\system32\umpnrmgr.dll		
cryptsvcs.dll	5.00.2181.1	61.77 KB (63,248 bytes)	services.exe	5.00.2134.1	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\cryptsvcs.dll			c:\winnt\system32\services.exe		
cryptdll.dll	5.00.2135.1	41.27 KB (42,256 bytes)	clbcatq.dll	1999.9.3422.14	479.27 KB (490,768 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation	11/13/2000 6:23:31 AM		Microsoft Corporation
c:\winnt\system32\cryptdll.dll			c:\winnt\system32\clbcatq.dll		
wkssvc.dll	5.00.2181.1	95.27 KB (97,552 bytes)	oleaut32.dll	2.40.4512	600.27 KB (614,672 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation	12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\wkssvc.dll			c:\winnt\system32\oleaut32.dll		
srvsvc.dll	5.00.2178.1	79.27 KB (81,168 bytes)	csui.dll	5.00.2172.1	227.27 KB (232,720 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation	12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\srvsvc.dll			c:\winnt\system32\csui.dll		
cfgmgr32.dll	5.00.2134.1	16.77 KB (17,168 bytes)	winspool.drv	5.00.2167.1	109.77 KB (112,400 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation	12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\cfgmgr32.dll			c:\winnt\system32\winspool.drv		
dmserver.dll	2191.1.296.2	11.77 KB (12,048 bytes)	winscard.dll	5.00.2134.1	77.27 KB (79,120 bytes)
12/7/1999 7:00:00 AM		VERITAS Software Corp.	12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\dmserver.dll			c:\winnt\system32\winscard.dll		
lmhsvc.dll	5.00.2134.1	9.27 KB (9,488 bytes)	wlnotify.dll	5.00.2164.1	53.27 KB (54,544 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation	12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\lmhsvc.dll			c:\winnt\system32\wlnotify.dll		
dnssrvr.dll	5.00.2181.1	88.27 KB (90,384 bytes)	csddl.dll	5.00.2189.1	98.27 KB (100,624 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation	12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\dnssrvr.dll			c:\winnt\system32\csddl.dll		
tapi32.dll	5.00.2182.1	123.27 KB (126,224 bytes)	lz32.dll	5.00.2134.1	9.77 KB (10,000 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation	12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\tapi32.dll			c:\winnt\system32\lz32.dll		
rasman.dll	5.00.2188.1	54.77 KB (56,080 bytes)	version.dll	5.00.2134.1	15.77 KB (16,144 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation	12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\rasman.dll			c:\winnt\system32\version.dll		
rasapi32.dll	5.00.2188.1	189.77 KB (194,320 bytes)	rsabase.dll	5.00.2150.1	128.77 KB (131,856 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation	12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\rasapi32.dll			c:\winnt\system32\rsabase.dll		
rtutils.dll	5.00.2168.1	43.77 KB (44,816 bytes)	mscat32.dll	5.131.2134.1	7.77 KB (7,952 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation	12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\rtutils.dll			c:\winnt\system32\mscat32.dll		

ole32.dll	5.00.2181.1	966.27 KB (989,456 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\ole32.dll		
imagehlp.dll	5.00.2195.1	125.27 KB (128,272 bytes)
12/7/1999 7:00:00 AM		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\msasn1.dll		
crypt32.dll	5.131.2173.1	465.77 KB (476,944 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\crypt32.dll		
wintrust.dll	5.131.2143.1	162.27 KB (166,160 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\wintrust.dll		
setupapi.dll	5.00.2183.1	554.27 KB (567,568 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\setupapi.dll		
winmm.dll	5.00.2161.1	184.77 KB (189,200 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\winmm.dll		
comctl32.dll	5.81	540.27 KB (553,232 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\comctl32.dll		
shlwapi.dll	5.00.2920.0000	282.77 KB (289,552 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\shlwapi.dll		
shell32.dll	5.00.2920.0000	2.24 MB (2,352,400 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\shell32.dll		
msgina.dll	5.00.2191.1	309.77 KB (317,200 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\msgina.dll		
winsta.dll	5.00.2134.1	36.27 KB (37,136 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\winsta.dll		
wsock32.dll	5.00.2152.1	21.27 KB (21,776 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\wsock32.dll		
dnsapi.dll	5.00.2181.1	129.77 KB (132,880 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\dnsapi.dll		
wldap32.dll	5.00.2168.1	155.77 KB (159,504 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\wldap32.dll		
ws2help.dll	5.00.2134.1	17.77 KB (18,192 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\ws2help.dll		
ws2_32.dll	5.00.2134.1	69.77 KB (71,440 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\ws2_32.dll		
samlib.dll	5.00.2160.1	46.27 KB (47,376 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\samlib.dll		
netrap.dll	5.00.2134.1	11.27 KB (11,536 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\netrap.dll		
netapi32.dll	5.00.2194.1	302.77 KB (310,032 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
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.dll	5.00.2154.1	46.77 KB (47,888 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\secur32.dll		

sfc.dll	5.00.2164.1	84.27 KB (86,288 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\sfc.dll		
nddeapi.dll	5.00.2137.1	15.27 KB (15,632 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\nddeapi.dll		
userenv.dll	5.00.2185.1	361.27 KB (369,936 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\userenv.dll		
user32.dll	5.00.2180.1	393.27 KB (402,704 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\user32.dll		
gdi32.dll	5.00.2180.1	228.77 KB (234,256 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\gdi32.dll		
rpcrt4.dll	5.00.2193.1	434.27 KB (444,688 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\rpcrt4.dll		
advapi32.dll	5.00.2191.1	349.27 KB (357,648 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\advapi32.dll		
kernel32.dll	5.00.2191.1	715.27 KB (732,432 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\kernel32.dll		
msvcrt.dll	6.10.8637.0	288.09 KB (295,000 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\msvcrt.dll		
winlogon.exe	5.00.2182.1	173.27 KB (177,424 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\winlogon.exe		
sfcfiles.dll	5.00.2195.1	973.27 KB (996,624 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\sfcfiles.dll		
ntdll.dll	5.00.2163.1	469.77 KB (481,040 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\ntdll.dll		
smss.exe	5.00.2170.1	44.27 KB (45,328 bytes)
12/7/1999 7:00:00 AM		Microsoft Corporation
c:\winnt\system32\smss.exe		

[Services]

Display Name	Name	State	Start Mode	Start Name
Service Type	Path	Error Control		
Tag ID				
Alerter	Alerter	Running	Auto	Share Process
c:\winnt\system32\services.exe		Normal	LocalSystem	0
Application Management		AppMgmt	Stopped	Manual
Share Process	c:\winnt\system32\services.exe	Normal	LocalSystem	0
Computer Browser	Browser	Running	Auto	Share Process
c:\winnt\system32\services.exe		Normal	LocalSystem	0
Indexing Service	cisvc	Stopped	Manual	Share Process
c:\winnt\system32\cisvc.exe		Normal	LocalSystem	0
ClipBook	ClipSrv	Stopped	Manual	Own Process
c:\winnt\system32\clipsrv.exe		Normal	LocalSystem	0
Distributed File System		Dfs	Running	Auto
Own Process	c:\winnt\system32\dfssvc.exe	Normal	LocalSystem	0
DHCP Client	Dhcp	Running	Auto	Share Process
c:\winnt\system32\services.exe		Normal	LocalSystem	0
Logical Disk Manager Administrative Service		Stopped	Manual	Share Process
c:\winnt\system32\dmadmin.exe	/com	Normal	LocalSystem	0
Logical Disk Manager		dmsrvr	Running	Auto
Share Process	c:\winnt\system32\services.exe	Normal	LocalSystem	0

DNS Client	Dnscache	Running	Auto	Share Process		Remote Access Auto Connection Manager	RasAuto	Stopped		
c:\winnt\system32\services.exe		Normal	LocalSystem	0		Manual	Share Process	c:\winnt\system32\svchost.exe -k		
Event Log	Eventlog	Running	Auto	Share Process		netsvcs	Normal	LocalSystem	0	
c:\winnt\system32\services.exe		Normal	LocalSystem	0		Remote Access Connection Manager	RasMan	Stopped		
COM+ Event System	EventSystem		Running	Manual		Manual	Share Process	c:\winnt\system32\svchost.exe -k		
Share Process	c:\winnt\system32\svchost.exe -k	netsvcs				netsvcs	Normal	LocalSystem	0	
Normal	LocalSystem		0			Routing and Remote Access	RemoteAccess	Stopped		
Fax Service	Fax	Stopped	Manual	Own Process		Disabled	Share Process	c:\winnt\system32\svchost.exe -k		
c:\winnt\system32\faxsvc.exe		Normal	LocalSystem	0		netsvcs	Normal	LocalSystem	0	
Internet Authentication Service	IAS	Running	Auto			Remote Registry Service	RemoteRegistry	Running		
Share Process	c:\winnt\system32\svchost.exe -k	netsvcs				Auto	Own Process	c:\winnt\system32\regsvc.exe		
Normal	LocalSystem		0			Normal	LocalSystem	0		
IIS Admin Service	IISADMIN	Running	Auto	Share Process		Remote Command Service	RMSYS	Running	Auto	
c:\winnt\system32\inetrv\inetinfo.exe		Normal	LocalSystem	0		Own Process	c:\benchcraft\rmsys.exe	Normal	LocalSystem	0
Intersite Messaging	IsmServ	Stopped	Disabled	Own Process		Remote Procedure Call (RPC) Locator	RpcLocator			
c:\winnt\system32\ismsserv.exe		Normal	LocalSystem	0		Stopped	Manual	Own Process		
Kerberos Key Distribution Center			kdc	Stopped		c:\winnt\system32\locator.exe	Normal	LocalSystem	0	
Disabled	Share Process	c:\winnt\system32\lsass.exe				Remote Procedure Call (RPC)	RpcSs	Running	Auto	
Normal	LocalSystem		0			Share Process	c:\winnt\system32\svchost -k	rpccs		
Server	lanmanserver	Running	Auto	Share Process		Normal	LocalSystem	0		
c:\winnt\system32\services.exe		Normal	LocalSystem	0		Remote Shell Service	RshSvc	Running	Auto	Own Process
Workstation	lanmanworkstation	Running	Auto			c:\winnt\system32\rshsvc.exe	Normal	LocalSystem	0	
Share Process	c:\winnt\system32\services.exe	Normal				QoS RSVP	RSVP	Running	Manual	Own Process
LocalSystem		0				c:\winnt\system32\rsvp.exe -s	Normal	LocalSystem	0	
License Logging Service	LicenseService	Running				Security Accounts Manager	SamSs	Running	Auto	
Auto	Own Process	c:\winnt\system32\llssrv.exe				Share Process	c:\winnt\system32\lsass.exe	Normal		
Normal	LocalSystem		0			LocalSystem		0		
TCP/IP NetBIOS Helper Service	LmHosts	Running	Auto			Smart Card Helper	SCardDrv	Stopped	Manual	Share Process
Share Process	c:\winnt\system32\services.exe	Normal				c:\winnt\system32\scardsvr.exe	Ignore	LocalSystem	0	
LocalSystem		0				Smart Card	SCardSvr	Stopped	Manual	Share Process
Messenger Messenger	Running	Auto	Share Process			c:\winnt\system32\scardsvr.exe	Ignore	LocalSystem	0	
c:\winnt\system32\services.exe		Normal	LocalSystem	0		Task Scheduler	Schedule	Running	Auto	Share Process
NetMeeting Remote Desktop Sharing	mnmsrv	Stopped				c:\winnt\system32\mstask.exe	Normal	LocalSystem	0	
Manual	Own Process	c:\winnt\system32\mnmsrv.exe				RunAs Service	seclogon	Running	Auto	Share Process
Normal	LocalSystem		0			c:\winnt\system32\services.exe	Ignore	LocalSystem	0	
Distributed Transaction Coordinator	MSDTC	Running				System Event Notification	SENS	Running	Auto	
Auto	Own Process	c:\winnt\system32\msdtc.exe				Share Process	c:\winnt\system32\svchost.exe -k	netsvcs		
Normal	LocalSystem		0			Normal	LocalSystem	0		
Windows Installer	MSIServer	Stopped	Manual	Share Process		Internet Connection Sharing	SharedAccess	Stopped		
c:\winnt\system32\msiexec.exe /v		Normal	LocalSystem	0		Manual	Share Process	c:\winnt\system32\svchost.exe -k		
0						netsvcs	Normal	LocalSystem	0	
Network DDE	NetDDE	Stopped	Manual	Share Process		Simple TCP/IP Services	SimpTcp	Running	Auto	
c:\winnt\system32\netdde.exe		Normal	LocalSystem	0		Share Process	c:\winnt\system32\tpcsvcs.exe	Normal		
Network DDE DSDM	NetDDEdsdm	Stopped	Manual			LocalSystem		0		
Share Process	c:\winnt\system32\netdde.exe	Normal				Simple Mail Transport Protocol (SMTP)	SMTPSVC			
LocalSystem		0				Running	Auto	Share Process		
Net Logon	Netlogon	Stopped	Manual	Share Process		c:\winnt\system32\inetrv\inetinfo.exe	Normal	LocalSystem	0	
c:\winnt\system32\lsass.exe		Normal	LocalSystem	0		Print Spooler	Spooler	Running	Auto	Own Process
Network Connections	Netman	Running	Manual	Share Process		c:\winnt\system32\spoolsv.exe	Normal	LocalSystem	0	
0						Performance Logs and Alerts	SysmonLog	Stopped		
File Replication	NtFrs	Stopped	Manual	Own Process		Manual	Own Process	c:\winnt\system32\smlogsvc.exe		
c:\winnt\system32\ntfrs.exe		Ignore	LocalSystem	0		Normal	LocalSystem	0		
NT LM Security Support Provider			NtLmSsp	Stopped		Telephony TapiSrv	Running	Manual	Share Process	
Manual	Share Process	c:\winnt\system32\lsass.exe				c:\winnt\system32\svchost.exe -k	tapisrv	Normal	LocalSystem	0
Normal	LocalSystem		0			Terminal Services	TermService	Running	Auto	
Removable Storage	NtmsSvc	Running	Auto	Share Process		Own Process	c:\winnt\system32\termsrv.exe	Normal	LocalSystem	0
c:\winnt\system32\svchost.exe -k	netsvcs	Normal	LocalSystem	0		LocalSystem		0		
Plug and Play	PlugPlay	Running	Auto	Share Process		Telnet	TlntSvr	Stopped	Manual	Own Process
c:\winnt\system32\services.exe		Normal	LocalSystem	0		c:\winnt\system32\tlntsvr.exe	Normal	LocalSystem	0	
IPSEC Policy Agent	PolicyAgent	Running	Auto			Distributed Link Tracking Server	TrkSvr	Stopped	Manual	
Share Process	c:\winnt\system32\lsass.exe	Normal				Share Process	c:\winnt\system32\services.exe	Normal		
LocalSystem		0				LocalSystem		0		
Protected Storage	ProtectedStorage	Running	Auto			Distributed Link Tracking Client	TrkWks	Running	Auto	
Share Process	c:\winnt\system32\services.exe	Normal				Share Process	c:\winnt\system32\services.exe	Normal		
LocalSystem		0				LocalSystem		0		



```

Uninterruptible Power Supply  UPS      Stopped  Manual
Own Process                  c:\winnt\system32\ups.exe  Normal
LocalSystem                  0
Utility Manager              UtilMan  Stopped  Manual  Own Process
c:\winnt\system32\utilman.exe Normal  LocalSystem  0
Windows Time                 W32Time Stopped  Manual  Share Process
c:\winnt\system32\services.exe Normal  LocalSystem  0
World Wide Web Publishing Service W3SVC  Running
Auto Share Process
c:\winnt\system32\inetrv\inetinfo.exe Normal  LocalSystem
0
Windows Management Instrumentation WinMgmt Running
Auto Own Process
c:\winnt\system32\wbem\winmgmt.exe Ignore  LocalSystem
0
Windows Management Instrumentation Driver Extensions
Wmi Running Manual Share Process
c:\winnt\system32\services.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  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\Communications  All
Users:Accessories\Communications  All Users
Accessories\Entertainment  All Users:Accessories\Entertainment
All Users
Accessories\System Tools  All Users:Accessories\System Tools
All Users
Administrative Tools All Users:Administrative Tools  All Users
Microsoft SQL ServerAll Users:Microsoft SQL Server All Users
Startup  All Users:Startup  All Users
Accessories  RTCLIENT64\tpcc:Accessories
RTCLIENT64\tpcc
Accessories\Accessibility
RTCLIENT64\tpcc:Accessories\Accessibility
RTCLIENT64\tpcc
Accessories\Entertainment
RTCLIENT64\tpcc:Accessories\Entertainment
RTCLIENT64\tpcc
Accessories\System Tools
RTCLIENT64\tpcc:Accessories\System Tools
RTCLIENT64\tpcc
Administrative Tools RTCLIENT64\tpcc:Administrative Tools
RTCLIENT64\tpcc
Startup  RTCLIENT64\tpcc:Startup  RTCLIENT64\tpcc
Accessories  RTCLIENT64\Administrator:Accessories
RTCLIENT64\Administrator
Accessories\Accessibility
RTCLIENT64\Administrator:Accessories\Accessibility
RTCLIENT64\Administrator
Accessories\Entertainment
RTCLIENT64\Administrator:Accessories\Entertainment
RTCLIENT64\Administrator
Accessories\System Tools
RTCLIENT64\Administrator:Accessories\SystemTools
RTCLIENT64\Administrator
Startup  RTCLIENT64\Administrator:Startup
RTCLIENT64\Administrator

```

[Startup Programs]

Program	Command	User Name	Location
synctime	synctime.cmd	All Users	Common Startup

[OLE Registration]

Object	Local Server
Sound (OLE2)	sndrec32.exe
Media Clip	mplay32.exe
Video Clip	mplay32.exe /avi
MIDI Sequence	mplay32.exe /mid
Sound	Not Available
Media Clip	Not Available
Image Document	"C:\Program Files\Windows NT\Accessories\ImageVue\KodakImg.exe"
WordPad Document	"%ProgramFiles%\Windows NT\Accessories\WORDPAD.EXE"
Windows Media Services DRM Storage object	Not Available
Bitmap Image	mspaint.exe

[Internet Explorer 5]

[ Following are sub-categories of this main category ]

[Summary]

Item	Value
Version	5.00.2920.0000
Build	52920
Product ID	51876-270-1133067-05431
Application Path	C:\Program Files\Internet Explorer
Language	English (United States)
Active Printer	Not Available
Cipher Strength	56-bit
Content Advisor	Disabled
IEAK Install	No

[File Versions]

File	Version	Size	Date	Path	Company
advapi32.dll	5.0.2191.1	349 KB	12/7/1999	7:00:00 AM	C:\WINNT\system32 Microsoft Corporation
advpack.dll	5.0.2920.0	87 KB	12/7/1999	7:00:00 AM	C:\WINNT\system32 Microsoft Corporation
browseic.dll	5.0.2920.0	35 KB	12/7/1999	7:00:00 AM	C:\WINNT\system32 Microsoft Corporation
browseui.dll	5.0.2920.0	793 KB	12/7/1999	7:00:00 AM	C:\WINNT\system32 Microsoft Corporation
ckcnv.exe	5.0.2189.1	9 KB	12/7/1999	7:00:00 AM	C:\WINNT\system32 Microsoft Corporation
comctl32.dll	5.81.2920.0	540 KB	12/7/1999	7:00:00 AM	C:\WINNT\system32 Microsoft Corporation
crypt32.dll	5.131.2173.1	466 KB	12/7/1999	7:00:00 AM	C:\WINNT\system32 Microsoft Corporation
enhsg.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
iemigrat.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
iesetup.dll	5.0.2920.0	57 KB	12/7/1999	7:00:00 AM	C:\WINNT\system32 Microsoft Corporation
ieexplore.exe	5.0.2920.0	59 KB	12/7/1999	7:00:00 AM	C:\Program Files\Internet Explorer Microsoft Corporation
imagehlp.dll	5.0.2195.1	125 KB	12/7/1999	7:00:00 AM	C:\WINNT\system32 Microsoft Corporation
imghelp.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available

```

inseng.dll 5.0.2920.0 72 KB 12/7/1999 7:00:00 AM
C:\WINNT\system32 Microsoft Corporation
jobexec.dll5.0.0.1 47 KB 12/7/1999 7:00:00 AM
C:\WINNT\system32 Microsoft Corporation
jscript.dll 5.1.0.4615 476 KB 12/7/1999 7:00:00 AM
C:\WINNT\system32 Microsoft Corporation
jsproxy.dll 5.0.2920.0 13 KB 12/7/1999 7:00:00 AM
C:\WINNT\system32 Microsoft Corporation
msaahtml.dll <File Missing> Not Available Not
Available Not Available Not Available
mshhtml.dll 5.0.2920.0 2302 KB 12/7/1999 7:00:00 AM
C:\WINNT\system32 Microsoft Corporation
msjava.dll 5.0.3234.0 918 KB 12/7/1999 7:00:00 AM
C:\WINNT\system32 Microsoft Corporation
msoss.dll <File Missing> Not Available Not Available
Not Available Not Available
msxml.dll 5.0.2920.0 521 KB 12/7/1999 7:00:00 AM
C:\WINNT\system32 Microsoft Corporation
occache.dll 5.0.2920.0 86 KB 12/7/1999 7:00:00 AM
C:\WINNT\system32 Microsoft Corporation
ole32.dll 5.0.2181.1 966 KB 12/7/1999 7:00:00 AM
C:\WINNT\system32 Microsoft Corporation
oleaut32.dll 2.40.4512.1 600 KB 12/7/1999
7:00:00 AM C:\WINNT\system32 Microsoft Corporation
olepro32.dll 5.0.4512.1 160 KB 12/7/1999 7:00:00 AM
C:\WINNT\system32 Microsoft Corporation
rsabase.dll 5.0.2150.1 129 KB 12/7/1999 7:00:00 AM
C:\WINNT\system32 Microsoft Corporation
rsaenh.dll <File Missing> Not Available Not Available
Not Available Not Available
rsapi32.dll <File Missing> Not Available Not Available
Not Available Not Available
rsasig.dll <File Missing> Not Available Not Available
Not Available Not Available
schannel.dll 5.0.2170.0 140 KB 12/7/1999 7:00:00 AM
C:\WINNT\system32 Microsoft Corporation
shdoc401.dll <File Missing> Not Available Not
Available Not Available Not Available
shdocvw.dll 5.0.2920.0 1078 KB 12/7/1999 7:00:00 AM
C:\WINNT\system32 Microsoft Corporation
shell32.dll 5.0.2920.0 2297 KB 12/7/1999 7:00:00 AM
C:\WINNT\system32 Microsoft Corporation
shlwapi.dll 5.0.2920.0 283 KB 12/7/1999 7:00:00 AM
C:\WINNT\system32 Microsoft Corporation
url.dll 5.0.2920.0 82 KB 12/7/1999 7:00:00 AM
C:\WINNT\system32 Microsoft Corporation
urlmon.dll 5.0.2920.0 427 KB 12/7/1999 7:00:00 AM
C:\WINNT\system32 Microsoft Corporation
vbscript.dll 5.1.0.4615 428 KB 12/7/1999 7:00:00 AM
C:\WINNT\system32 Microsoft Corporation
webcheck.dll 5.0.2920.0 252 KB 12/7/1999 7:00:00 AM
C:\WINNT\system32 Microsoft Corporation
win.com 5.0.2134.1 24 KB 12/7/1999 7:00:00 AM
C:\WINNT\system32 Microsoft Corporation
wininet.dll 5.0.2920.0 457 KB 12/7/1999 7:00:00 AM
C:\WINNT\system32 Microsoft Corporation
winsock.dll 3.10.0.103 3 KB 12/7/1999 7:00:00 AM
C:\WINNT\system32 Microsoft Corporation
wintrust.dll 5.131.2143.1 162 KB 12/7/1999
7:00:00 AM C:\WINNT\system32 Microsoft Corporation
wsock.vxd <File Missing> Not Available Not Available
Not Available Not Available
wsock32.dll 5.0.2152.1 21 KB 12/7/1999 7:00:00 AM
C:\WINNT\system32 Microsoft Corporation
wsock32n.dll <File Missing> 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\tpcc\Local Settings\Temporary Internet Files
Total Disk Space 8675 MB
Available Disk Space 7055 MB
Maximum Cache Size 270 MB
Available Cache Size 271 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
No personal certificate information available

[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

```

**xSeries 220**

This system configuration inventory was obtained from 1 of 4 clients (xSeries 220).

All 4 clients are identically configured.

System Information report written at: 03/19/2001 01:14:56 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 Build 2195
OS Manufacturer	Microsoft Corporation
System Name	CLIENT10
System Manufacturer	IBM
System Model	IBM eserver xSeries 220 -[8645]-
System Type	X86-based PC
Processor	x86 Family 6 Model 8 Stepping 6 GenuineIntel ~933 Mhz
Processor	x86 Family 6 Model 8 Stepping 6 GenuineIntel ~933 Mhz
BIOS Version	IBM BIOS Ver 3.0
Windows Directory	C:\WINNT
System Directory	C:\WINNT\System32
Boot Device	\Device\Harddisk0\Partition1
Locale	United States
User Name	CLIENT10\Administrator
Time Zone	Eastern Standard Time
Total Physical Memory	654,808 KB
Available Physical Memory	558,516 KB
Total Virtual Memory	2,865,768 KB
Available Virtual Memory	2,678,784 KB
Page File Space	2,210,960 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
2	Standard floppy disk controller	OK
4	Direct memory access controller	OK

[Forced Hardware]

Device	PNP Device ID
No Forced Hardware	

[I/O]

Address Range	Device	Status
0x0000-0x203F	PCI bus	OK
0x0000-0x203F	Direct memory access controller	OK
0x03B0-0x03BB	S3 Inc. Savage4	OK
0x03C0-0x03DF	S3 Inc. Savage4	OK
0x2000-0x203F	IBM Netfinity 10/100 Ethernet Adapter	OK
0x0A79-0x0A79	ISAPNP Read Data Port	OK
0x0279-0x0279	ISAPNP Read Data Port	OK

0x02F4-0x02F7	ISAPNP Read Data Port	OK
0x002E-0x002F	Motherboard resources	OK
0x0438-0x0439	Motherboard resources	OK
0x0430-0x0437	Motherboard resources	OK
0x0430-0x0437	Not Available	OK
0x0060-0x0060	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard	OK
0x0064-0x0064	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard	OK
0x03F0-0x03F5	Standard floppy disk controller	OK
0x03F7-0x03F7	Standard floppy disk controller	OK
0x0378-0x037F	Printer Port (LPT1)	OK
0x03F8-0x03FF	Communications Port (COM1)	OK
0x02F8-0x02FF	Communications Port (COM2)	OK
0x0020-0x0021	Advanced programmable interrupt controller	OK
0x00A0-0x00A1	Advanced programmable interrupt controller	OK
0x0080-0x008F	Direct memory access controller	OK
0x00C0-0x00DF	Direct memory access controller	OK
0x0040-0x0043	System timer	OK
0x0070-0x0073	System CMOS/real time clock	OK
0x0061-0x0061	System speaker	OK
0x00F0-0x00FF	Numeric data processor	OK
0x0600-0x0600	Motherboard resources	OK
0x0374-0x0375	Motherboard resources	OK
0x0377-0x0377	Motherboard resources	OK
0x0F50-0x0F58	Motherboard resources	OK
0x0700-0x070F	Standard Dual Channel PCI IDE Controller	OK
0x01F0-0x01F7	Primary IDE Channel	OK
0x03F6-0x03F6	Primary IDE Channel	OK
0x0170-0x0177	Secondary IDE Channel	OK
0x0376-0x0376	Secondary IDE Channel	OK
0x2040-0xFFFF	PCI bus	OK
0x2100-0x21FF	Adaptec AIC-7892 Ultra160/m PCI SCSI Card	OK
0x3000-0x3FFF	DEC 21152 PCI to PCI bridge	OK
0x3000-0x3FFF	Intel(R) PRO/100+ Dual Port Server Adapter	OK
0x3020-0x303F	Intel(R) PRO/100+ Dual Port Server Adapter #2	OK

[IRQs]

IRQ Number	Device
30	Microsoft ACPI-Compliant System
31	S3 Inc. Savage4
27	IBM Netfinity 10/100 Ethernet Adapter
1	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard
12	PS/2 Compatible Mouse
6	Standard floppy disk controller
4	Communications Port (COM1)
3	Communications Port (COM2)
8	System CMOS/real time clock
13	Numeric data processor
14	Primary IDE Channel
10	Standard OpenHCD USB Host Controller
28	Adaptec AIC-7892 Ultra160/m PCI SCSI Card
24	Intel(R) PRO/100+ Dual Port Server Adapter
25	Intel(R) PRO/100+ Dual Port Server Adapter #2

[Memory]

Range	Device	Status
0xA0000-0xBFFFF	PCI bus	OK
0xA0000-0xBFFFF	S3 Inc. Savage4	OK
0xF0000000-0xFCFFFFFF	PCI bus	OK
0xF0000000-0xFCFFFFFF	S3 Inc. Savage4	OK

0xFD000000-0xFFFFFFFF PCI bus OK  
 0xFEB80000-0xFEBFFFFFF S3 Inc. Savage4 OK  
 0xFEB7F000-0xFEB7FFFF IBM Netfinity 10/100 Ethernet Adapter OK  
 0xFEAA0000-0xFEAFFFFF IBM Netfinity 10/100 Ethernet Adapter OK  
 0xFEB7E000-0xFEB7EFFF Standard OpenHCD USB Host Controller OK  
 0x28000000-0xECFFFFFF PCI bus OK  
 0xED000000-0xEFFFFFFF PCI bus OK  
 0xEFFF0000-0xEFFFFFFF Adaptec AIC-7892 Ultra160/m PCI SCSI Card OK  
 0xEE000000-0xEF1FFFFF DEC 21152 PCI to PCI bridge OK  
 0xECF00000-0xECFFFFFF DEC 21152 PCI to PCI bridge OK  
 0xECF00000-0xECFFFFFF Intel(R) PRO/100+ Dual Port Server Adapter OK  
 0xEF000000-0xEF0FFFFF Intel(R) PRO/100+ Dual Port Server Adapter OK  
 0xECF01000-0xECF01FFF Intel(R) PRO/100+ Dual Port Server Adapter #2 OK  
 0xEF100000-0xEF1FFFFF Intel(R) PRO/100+ Dual Port Server Adapter #2 OK

[Components]

[ Following are sub-categories of this main category ]

[Multimedia]

[ Following are sub-categories of this main category ]

[Audio Codecs]

Codec Version	Manufacturer Size	Description Creation Date	Status	File
c:\winnt\system32\iac25_32.ax	2.05.53 195.00 KB (199,680 bytes)	Intel Corporation	Indeo® audio software OK	C:\WINNT\System32\IAC25_32.AX 12/7/1999 7:00:00 AM
c:\winnt\system32\msg723.acm	106.77 KB (109,328 bytes)	Microsoft Corporation	OK	C:\WINNT\System32\MSG723.ACM 2/5/2001 3:49:25 PM
c:\winnt\system32\lhacm.acm	33.27 KB (34,064 bytes)	Microsoft Corporation	OK	C:\WINNT\System32\LHACM.ACM 2/5/2001 3:49:25 PM
c:\winnt\system32\msgsm32.acm	22.27 KB (22,800 bytes)	Microsoft Corporation	OK	C:\WINNT\System32\MSGSM32.ACM 12/7/1999 7:00:00 AM
c:\winnt\system32\msg711.acm	10.27 KB (10,512 bytes)	Microsoft Corporation	OK	C:\WINNT\System32\MSG711.ACM 12/7/1999 7:00:00 AM
c:\winnt\system32\tsoft32.acm	9.27 KB (9,488 bytes)	DSP GROUP, INC.	OK	C:\WINNT\System32\TSSOFT32.ACM 1.01 12/7/1999 7:00:00 AM
c:\winnt\system32\msadp32.acm	14.77 KB (15,120 bytes)	Microsoft Corporation	OK	C:\WINNT\System32\MSADP32.ACM 5.00.2134.1 12/7/1999 7:00:00 AM
c:\winnt\system32\imaadp32.acm	5.00.2134.1 16.27 KB (16,656 bytes)	Microsoft Corporation	OK	C:\WINNT\System32\IMAADP32.ACM 12/7/1999 7:00:00 AM

[Video Codecs]

Codec Version	Manufacturer	Description	Status	File
c:\winnt\system32\ir50_32.dll	5.10 OK	Intel Corporation	Indeo® video	C:\WINNT\System32\IR50_32.DLL

File Name	Size	Manufacturer	Status	Date
c:\winnt\system32\iccvid.dll	737.50 KB (755,200 bytes)	Radius Inc.	OK	12/7/1999 7:00:00 AM
C:\WINNT\System32\ICCVID.DLL	108.00 KB (110,592 bytes)	1.10.0.6	OK	12/7/1999 7:00:00 AM
c:\winnt\system32\ir32_32.dll	194.50 KB (199,168 bytes)	Intel(R) Corporation	OK	12/7/1999 7:00:00 AM
C:\WINNT\System32\IR32_32.DLL	Not Available	Not Available		12/7/1999 7:00:00 AM
c:\winnt\system32\msvidc32.dll	27.27 KB (27,920 bytes)	Microsoft Corporation	OK	12/7/1999 7:00:00 AM
C:\WINNT\System32\MSVIDC32.DLL	5.00.2134.1			12/7/1999 7:00:00 AM
c:\winnt\system32\msh263.drv	252.27 KB (258,320 bytes)	Microsoft Corporation	OK	2/5/2001 3:49:00 PM
C:\WINNT\System32\MSH263.DRV	4.4.3385			2/5/2001 3:49:00 PM
c:\winnt\system32\msh261.drv	163.77 KB (167,696 bytes)	Microsoft Corporation	OK	2/5/2001 3:49:25 PM
C:\WINNT\System32\MSH261.DRV	4.4.3385			2/5/2001 3:49:25 PM
c:\winnt\system32\msrle32.dll	10.77 KB (11,024 bytes)	Microsoft Corporation	OK	12/7/1999 7:00:00 AM
C:\WINNT\System32\MSRLE32.DLL	5.00.2134.1			12/7/1999 7:00:00 AM

[CD-ROM]

Item	Value
Drive	D:
Description	CD-ROM Drive
Media Loaded	False
Media Type	CD-ROM
Name	LITEON CD-ROM LTN485S
Manufacturer	(Standard CD-ROM drives)
Status	OK
Transfer Rate	Not Available
SCSI Target ID	0
PNP Device ID	IDE\CDROMLITEON_CD-ROM_LTN485S_____J
U04_____	\5&326853DD&0&0.0.0

[Sound Device]

Item	Value
No sound devices	

[Display]

Item	Value
Name	S3 Inc. Savage4
PNP Device ID	PCI\VEN_5333&DEV_8A22&SUBSYS_01C51014&REV_04\3&267A616A&0&08
Adapter Type	S3 Savage4, S3 compatible
Adapter Description	S3 Inc. Savage4
Adapter RAM	8.00 MB (8,388,608 bytes)
Installed Drivers	s3sav4.dll
Driver Version	5.01.840.0001
INF File	s3sav4.inf (S3Inc section)
Color Planes	1
Color Table Entries	65536
Resolution	800 x 600 x 60 hertz
Bits/Pixel	16

[Infrared]

Item	Value
No infrared devices	

[Input]

[ Following are sub-categories of this main category ]

[Keyboard]

Item Value  
Description Standard 101/102-Key or Microsoft Natural  
PS/2 Keyboard  
Name Enhanced (101- or 102-key)  
Layout 00000409  
PNP Device ID ACPI\PNP0303\4&23FD4C84&0  
NumberOfFunctionKeys 12

[Pointing Device]

Item Value  
Hardware Type PS/2 Compatible Mouse  
Number of Buttons 3  
Status OK  
PNP Device ID ACPI\PNP0F13\4&23FD4C84&0  
Power Management Supported False  
Double Click Threshold 6  
Handedness Right Handed Operation

[Modem]

Item Value  
No modems

[Network]

[ Following are sub-categories of this main category ]

[Adapter]

Item Value  
Name [00000000] IBM Netfinity 10/100 Ethernet Adapter  
Adapter Type Ethernet 802.3  
Product Name IBM Netfinity 10/100 Ethernet Adapter  
Installed True  
PNP Device ID  
PCI\VEN\_8086&DEV\_1229&SUBSYS\_105C1014&REV\_08\3&267  
A616A&0&10  
Last Reset 3/19/2001 8:05:54 AM  
Index 0  
Service Name E100B  
IP Address 192.168.125.11  
IP Subnet 255.255.255.0  
Default IP Gateway Not Available  
DHCP Enabled False  
DHCP Server Not Available  
DHCP Lease Expires Not Available  
DHCP Lease Obtained Not Available  
MAC Address 00:02:55:D4:77:D1  
Service Name E100B  
IRQ Number 27  
I/O Port 0x2000-0x203F  
Driver c:\winnt\system32\drivers\e100bnt5.sys (104720,  
5.00.67.0000)

Name [00000001] Intel(R) PRO/100+ Dual Port Server Adapter  
Adapter Type Ethernet 802.3  
Product Name Intel(R) PRO/100+ Dual Port Server Adapter  
Installed True

PNP Device ID  
PCI\VEN\_8086&DEV\_1229&SUBSYS\_10F08086&REV\_05\4&27A  
BF15E&0&2038  
Last Reset 3/19/2001 8:05:54 AM  
Index 1  
Service Name E100B  
IP Address 192.168.15.99  
IP Subnet 255.255.255.0  
Default IP Gateway Not Available  
DHCP Enabled False  
DHCP Server Not Available  
DHCP Lease Expires Not Available  
DHCP Lease Obtained Not Available  
MAC Address 00:03:47:0D:AD:B8  
Service Name E100B  
IRQ Number 24  
I/O Port 0x3000-0x3FFF  
Driver c:\winnt\system32\drivers\e100bnt5.sys (104720,  
5.00.67.0000)

Name [00000002] Intel(R) PRO/100+ Dual Port Server Adapter  
Adapter Type Ethernet 802.3  
Product Name Intel(R) PRO/100+ Dual Port Server Adapter  
Installed True  
PNP Device ID  
PCI\VEN\_8086&DEV\_1229&SUBSYS\_10F08086&REV\_05\4&27A  
BF15E&0&2838  
Last Reset 3/19/2001 8:05:54 AM  
Index 2  
Service Name E100B  
IP Address 192.168.15.199  
IP Subnet 255.255.255.0  
Default IP Gateway Not Available  
DHCP Enabled False  
DHCP Server Not Available  
DHCP Lease Expires Not Available  
DHCP Lease Obtained Not Available  
MAC Address 00:03:47:0D:AD:B9  
Service Name E100B  
IRQ Number 25  
I/O Port 0x3020-0x303F  
Driver c:\winnt\system32\drivers\e100bnt5.sys (104720,  
5.00.67.0000)

Name [00000003] RAS Async Adapter  
Adapter Type Not Available  
Product Name RAS Async Adapter  
Installed True  
PNP Device ID Not Available  
Last Reset 3/19/2001 8:05:54 AM  
Index 3  
Service Name AsyncMac  
IP Address Not Available  
IP Subnet Not Available  
Default IP Gateway Not Available  
DHCP Enabled False  
DHCP Server Not Available  
DHCP Lease Expires Not Available  
DHCP Lease Obtained Not Available  
MAC Address Not Available  
Service Name Not Available

Name [00000004] WAN Miniport (L2TP)  
Adapter Type Not Available  
Product Name WAN Miniport (L2TP)  
Installed True  
PNP Device ID ROOT\MS\_L2TPMINIPORT\0000  
Last Reset 3/19/2001 8:05:54 AM

Index 4  
 Service Name Rasl2tp  
 IP Address Not Available  
 IP Subnet Not Available  
 Default IP Gateway Not Available  
 DHCP Enabled False  
 DHCP Server Not Available  
 DHCP Lease Expires Not Available  
 DHCP Lease Obtained Not Available  
 MAC Address Not Available  
 Service Name Rasl2tp  
 Driver c:\winnt\system32\drivers\rasl2tp.sys (50800, 5.00.2179.1)

Name [00000005] WAN Miniport (PPTP)  
 Adapter Type Wide Area Network (WAN)  
 Product Name WAN Miniport (PPTP)  
 Installed True  
 PNP Device ID ROOT\MS\_PPTPMINIPOINT\0000  
 Last Reset 3/19/2001 8:05:54 AM

Index 5  
 Service Name PptpMiniport  
 IP Address Not Available  
 IP Subnet Not Available  
 Default IP Gateway Not Available  
 DHCP Enabled False  
 DHCP Server Not Available  
 DHCP Lease Expires Not Available  
 DHCP Lease Obtained Not Available  
 MAC Address 50:50:54:50:30:30  
 Service Name PptpMiniport  
 Driver c:\winnt\system32\drivers\rasptp.sys (47856, 5.00.2160.1)

Name [00000006] Direct Parallel  
 Adapter Type Not Available  
 Product Name Direct Parallel  
 Installed True  
 PNP Device ID ROOT\MS\_PTIMINIPOINT\0000  
 Last Reset 3/19/2001 8:05:54 AM

Index 6  
 Service Name Raspti  
 IP Address Not Available  
 IP Subnet Not Available  
 Default IP Gateway Not Available  
 DHCP Enabled False  
 DHCP Server Not Available  
 DHCP Lease Expires Not Available  
 DHCP Lease Obtained Not Available  
 MAC Address Not Available  
 Service Name Raspti  
 Driver c:\winnt\system32\drivers\raspti.sys (16880, 5.00.2146.1)

Name [00000007] WAN Miniport (IP)  
 Adapter Type Not Available  
 Product Name WAN Miniport (IP)  
 Installed True  
 PNP Device ID ROOT\MS\_NDISWANIP\0000  
 Last Reset 3/19/2001 8:05:54 AM

Index 7  
 Service Name NdisWan  
 IP Address Not Available  
 IP Subnet Not Available  
 Default IP Gateway Not Available  
 DHCP Enabled False  
 DHCP Server Not Available  
 DHCP Lease Expires Not Available  
 DHCP Lease Obtained Not Available  
 MAC Address Not Available  
 Service Name NdisWan

Driver c:\winnt\system32\drivers\ndiswan.sys (90768, 5.00.2184.1)

Name [00000008] Intel(R) PRO/100+ Dual Port Server Adapter  
 Adapter Type Not Available  
 Product Name Intel(R) PRO/100+ Dual Port Server Adapter  
 Installed True  
 PNP Device ID Not Available  
 Last Reset 3/19/2001 8:05:54 AM

Index 8  
 Service Name E100B  
 IP Address 192.168.15.99  
 IP Subnet 255.255.255.0  
 Default IP Gateway Not Available  
 DHCP Enabled False  
 DHCP Server Not Available  
 DHCP Lease Expires Not Available  
 DHCP Lease Obtained Not Available  
 MAC Address 00:03:47:0D:AD:B8  
 Service Name Not Available

Name [00000009] Intel(R) PRO/100+ Dual Port Server Adapter  
 Adapter Type Not Available  
 Product Name Intel(R) PRO/100+ Dual Port Server Adapter  
 Installed True  
 PNP Device ID Not Available  
 Last Reset 3/19/2001 8:05:54 AM

Index 9  
 Service Name E100B  
 IP Address Not Available  
 IP Subnet Not Available  
 Default IP Gateway Not Available  
 DHCP Enabled True  
 DHCP Server Not Available  
 DHCP Lease Expires Not Available  
 DHCP Lease Obtained Not Available  
 MAC Address Not Available  
 Service Name Not Available

Name [00000010] Intel(R) PRO/100+ Dual Port Server Adapter  
 Adapter Type Not Available  
 Product Name Intel(R) PRO/100+ Dual Port Server Adapter  
 Installed True  
 PNP Device ID Not Available  
 Last Reset 3/19/2001 8:05:54 AM

Index 10  
 Service Name E100B  
 IP Address 192.168.125.11  
 IP Subnet 255.255.255.0  
 Default IP Gateway Not Available  
 DHCP Enabled False  
 DHCP Server Not Available  
 DHCP Lease Expires Not Available  
 DHCP Lease Obtained Not Available  
 MAC Address 00:02:55:D4:77:D1  
 Service Name Not Available

Name [00000011] Intel(R) PRO/100+ Dual Port Server Adapter  
 Adapter Type Not Available  
 Product Name Intel(R) PRO/100+ Dual Port Server Adapter  
 Installed True  
 PNP Device ID Not Available  
 Last Reset 3/19/2001 8:05:54 AM

Index 11  
 Service Name E100B  
 IP Address Not Available  
 IP Subnet Not Available  
 Default IP Gateway Not Available

DHCP Enabled True  
 DHCP Server Not Available  
 DHCP Lease Expires Not Available  
 DHCP Lease Obtained Not Available  
 MAC Address Not Available  
 Service Name Not Available

[Protocol]

Item Value  
 Name MSAFD Tcpi [TCP/IP]  
 ConnectionlessService False  
 GuaranteesDelivery True  
 GuaranteesSequencing True  
 MaximumAddressSize 16 bytes  
 MaximumMessageSize 0 bytes  
 MessageOriented False  
 MinimumAddressSize 16 bytes  
 PseudoStreamOriented False  
 SupportsBroadcasting False  
 SupportsConnectData False  
 SupportsDisconnectData False  
 SupportsEncryption False  
 SupportsExpeditedData True  
 SupportsGracefulClosing True  
 SupportsGuaranteedBandwidth False  
 SupportsMulticasting False

Name MSAFD Tcpi [UDP/IP]  
 ConnectionlessService True  
 GuaranteesDelivery False  
 GuaranteesSequencing False  
 MaximumAddressSize 16 bytes  
 MaximumMessageSize 65467 bytes  
 MessageOriented True  
 MinimumAddressSize 16 bytes  
 PseudoStreamOriented False  
 SupportsBroadcasting True  
 SupportsConnectData False  
 SupportsDisconnectData False  
 SupportsEncryption False  
 SupportsExpeditedData False  
 SupportsGracefulClosing False  
 SupportsGuaranteedBandwidth False  
 SupportsMulticasting True

Name RSVP UDP Service Provider  
 ConnectionlessService True  
 GuaranteesDelivery False  
 GuaranteesSequencing False  
 MaximumAddressSize 16 bytes  
 MaximumMessageSize 65467 bytes  
 MessageOriented True  
 MinimumAddressSize 16 bytes  
 PseudoStreamOriented False  
 SupportsBroadcasting True  
 SupportsConnectData False  
 SupportsDisconnectData False  
 SupportsEncryption True  
 SupportsExpeditedData False  
 SupportsGracefulClosing False  
 SupportsGuaranteedBandwidth False  
 SupportsMulticasting True

Name RSVP TCP Service Provider  
 ConnectionlessService False  
 GuaranteesDelivery True

GuaranteesSequencing True  
 MaximumAddressSize 16 bytes  
 MaximumMessageSize 0 bytes  
 MessageOriented False  
 MinimumAddressSize 16 bytes  
 PseudoStreamOriented False  
 SupportsBroadcasting False  
 SupportsConnectData False  
 SupportsDisconnectData False  
 SupportsEncryption True  
 SupportsExpeditedData True  
 SupportsGracefulClosing True  
 SupportsGuaranteedBandwidth False  
 SupportsMulticasting False

Name MSAFD NetBIOS

[[Device\NetBT\_Tcpi\_{979D4457-D388-4601-80B5-A16899D791E1}]] SEQUENCE 8  
 ConnectionlessService False  
 GuaranteesDelivery True  
 GuaranteesSequencing True  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize 20 bytes  
 PseudoStreamOriented False  
 SupportsBroadcasting False  
 SupportsConnectData False  
 SupportsDisconnectData False  
 SupportsEncryption False  
 SupportsExpeditedData False  
 SupportsGracefulClosing False  
 SupportsGuaranteedBandwidth False  
 SupportsMulticasting False

Name MSAFD NetBIOS

[[Device\NetBT\_Tcpi\_{979D4457-D388-4601-80B5-A16899D791E1}]] DATAGRAM 8  
 ConnectionlessService True  
 GuaranteesDelivery False  
 GuaranteesSequencing False  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize 20 bytes  
 PseudoStreamOriented False  
 SupportsBroadcasting True  
 SupportsConnectData False  
 SupportsDisconnectData False  
 SupportsEncryption False  
 SupportsExpeditedData False  
 SupportsGracefulClosing False  
 SupportsGuaranteedBandwidth False  
 SupportsMulticasting False

Name MSAFD NetBIOS

[[Device\NetBT\_Tcpi\_{949B5C72-3DD4-4846-A2D4-4E6333C49D13}]] SEQUENCE 7  
 ConnectionlessService False  
 GuaranteesDelivery True  
 GuaranteesSequencing True  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize 20 bytes

PseudoStreamOriented False  
 SupportsBroadcasting False  
 SupportsConnectData False  
 SupportsDisconnectData False  
 SupportsEncryption False  
 SupportsExpeditedData False  
 SupportsGracefulClosing False  
 SupportsGuaranteedBandwidth False  
 SupportsMulticasting False

Name MSAFD NetBIOS

[\\Device\\NetBT\_Tcpip\_{949B5C72-3DD4-4846-A2D4-4E6333C49D13}] DATAGRAM 7  
 ConnectionlessService True  
 GuaranteesDelivery False  
 GuaranteesSequencing False  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize 20 bytes  
 PseudoStreamOriented False  
 SupportsBroadcasting True  
 SupportsConnectData False  
 SupportsDisconnectData False  
 SupportsEncryption False  
 SupportsExpeditedData False  
 SupportsGracefulClosing False  
 SupportsGuaranteedBandwidth False  
 SupportsMulticasting False

Name MSAFD NetBIOS

[\\Device\\NetBT\_Tcpip\_{C91C0CB5-B4E1-4EA6-B7CB-3907BC6D72D2}] SEQPACKET 6  
 ConnectionlessService False  
 GuaranteesDelivery True  
 GuaranteesSequencing True  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize 20 bytes  
 PseudoStreamOriented False  
 SupportsBroadcasting False  
 SupportsConnectData False  
 SupportsDisconnectData False  
 SupportsEncryption False  
 SupportsExpeditedData False  
 SupportsGracefulClosing False  
 SupportsGuaranteedBandwidth False  
 SupportsMulticasting False

Name MSAFD NetBIOS

[\\Device\\NetBT\_Tcpip\_{C91C0CB5-B4E1-4EA6-B7CB-3907BC6D72D2}] DATAGRAM 6  
 ConnectionlessService True  
 GuaranteesDelivery False  
 GuaranteesSequencing False  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize 20 bytes  
 PseudoStreamOriented False  
 SupportsBroadcasting True  
 SupportsConnectData False  
 SupportsDisconnectData False  
 SupportsEncryption False

SupportsExpeditedData False  
 SupportsGracefulClosing False  
 SupportsGuaranteedBandwidth False  
 SupportsMulticasting False

Name MSAFD NetBIOS

[\\Device\\NetBT\_Tcpip\_{4EE22FC7-83B6-45AD-AE3C-869D662F6E7D}] SEQPACKET 5  
 ConnectionlessService False  
 GuaranteesDelivery True  
 GuaranteesSequencing True  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize 20 bytes  
 PseudoStreamOriented False  
 SupportsBroadcasting False  
 SupportsConnectData False  
 SupportsDisconnectData False  
 SupportsEncryption False  
 SupportsExpeditedData False  
 SupportsGracefulClosing False  
 SupportsGuaranteedBandwidth False  
 SupportsMulticasting False

Name MSAFD NetBIOS

[\\Device\\NetBT\_Tcpip\_{4EE22FC7-83B6-45AD-AE3C-869D662F6E7D}] DATAGRAM 5  
 ConnectionlessService True  
 GuaranteesDelivery False  
 GuaranteesSequencing False  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize 20 bytes  
 PseudoStreamOriented False  
 SupportsBroadcasting True  
 SupportsConnectData False  
 SupportsDisconnectData False  
 SupportsEncryption False  
 SupportsExpeditedData False  
 SupportsGracefulClosing False  
 SupportsGuaranteedBandwidth False  
 SupportsMulticasting False

Name MSAFD NetBIOS

[\\Device\\NetBT\_Tcpip\_{F7D98B12-E5E5-4D36-AF7B-4F430C2D56C2}] SEQPACKET 0  
 ConnectionlessService False  
 GuaranteesDelivery True  
 GuaranteesSequencing True  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize 20 bytes  
 PseudoStreamOriented False  
 SupportsBroadcasting False  
 SupportsConnectData False  
 SupportsDisconnectData False  
 SupportsEncryption False  
 SupportsExpeditedData False  
 SupportsGracefulClosing False  
 SupportsGuaranteedBandwidth False  
 SupportsMulticasting False



Name MSAFD NetBIOS

[\Device\NetBT\_Tcpip\_{F7D98B12-E5E5-4D36-AF7B-4F430C2D56C2}] DATAGRAM0

ConnectionlessService True

GuaranteesDelivery False

GuaranteesSequencing False

MaximumAddressSize 20 bytes

MaximumMessageSize 64000 bytes

MessageOriented True

MinimumAddressSize 20 bytes

PseudoStreamOriented False

SupportsBroadcasting True

SupportsConnectData False

SupportsDisconnectData False

SupportsEncryption False

SupportsExpeditedData False

SupportsGracefulClosing False

SupportsGuaranteedBandwidth False

SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT\_Tcpip\_{57687CD0-D67F-47D8-BEC5-EFD99AB56155}] SEQPACKET1

ConnectionlessService False

GuaranteesDelivery True

GuaranteesSequencing True

MaximumAddressSize 20 bytes

MaximumMessageSize 64000 bytes

MessageOriented True

MinimumAddressSize 20 bytes

PseudoStreamOriented False

SupportsBroadcasting False

SupportsConnectData False

SupportsDisconnectData False

SupportsEncryption False

SupportsExpeditedData False

SupportsGracefulClosing False

SupportsGuaranteedBandwidth False

SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT\_Tcpip\_{57687CD0-D67F-47D8-BEC5-EFD99AB56155}] DATAGRAM1

ConnectionlessService True

GuaranteesDelivery False

GuaranteesSequencing False

MaximumAddressSize 20 bytes

MaximumMessageSize 64000 bytes

MessageOriented True

MinimumAddressSize 20 bytes

PseudoStreamOriented False

SupportsBroadcasting True

SupportsConnectData False

SupportsDisconnectData False

SupportsEncryption False

SupportsExpeditedData False

SupportsGracefulClosing False

SupportsGuaranteedBandwidth False

SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT\_Tcpip\_{EF97AFD3-EF48-4232-A0DE-483DD8621609}] SEQPACKET2

ConnectionlessService False

GuaranteesDelivery True

GuaranteesSequencing True

MaximumAddressSize 20 bytes

MaximumMessageSize 64000 bytes

MessageOriented True

MinimumAddressSize 20 bytes

PseudoStreamOriented False

SupportsBroadcasting False

SupportsConnectData False

SupportsDisconnectData False

SupportsEncryption False

SupportsExpeditedData False

SupportsGracefulClosing False

SupportsGuaranteedBandwidth False

SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT\_Tcpip\_{EF97AFD3-EF48-4232-A0DE-483DD8621609}] DATAGRAM2

ConnectionlessService True

GuaranteesDelivery False

GuaranteesSequencing False

MaximumAddressSize 20 bytes

MaximumMessageSize 64000 bytes

MessageOriented True

MinimumAddressSize 20 bytes

PseudoStreamOriented False

SupportsBroadcasting True

SupportsConnectData False

SupportsDisconnectData False

SupportsEncryption False

SupportsExpeditedData False

SupportsGracefulClosing False

SupportsGuaranteedBandwidth False

SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT\_Tcpip\_{9134BD2A-03E6-42E3-8007-99206B0A4909}] SEQPACKET3

ConnectionlessService False

GuaranteesDelivery True

GuaranteesSequencing True

MaximumAddressSize 20 bytes

MaximumMessageSize 64000 bytes

MessageOriented True

MinimumAddressSize 20 bytes

PseudoStreamOriented False

SupportsBroadcasting False

SupportsConnectData False

SupportsDisconnectData False

SupportsEncryption False

SupportsExpeditedData False

SupportsGracefulClosing False

SupportsGuaranteedBandwidth False

SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT\_Tcpip\_{9134BD2A-03E6-42E3-8007-99206B0A4909}] DATAGRAM3

ConnectionlessService True

GuaranteesDelivery False

GuaranteesSequencing False

MaximumAddressSize 20 bytes

MaximumMessageSize 64000 bytes

MessageOriented True

MinimumAddressSize 20 bytes  
PseudoStreamOriented False  
SupportsBroadcasting True  
SupportsConnectData False  
SupportsDisconnectData False  
SupportsEncryption False  
SupportsExpeditedData False  
SupportsGracefulClosing False  
SupportsGuaranteedBandwidth False  
SupportsMulticasting False

Name MSAFD NetBIOS

[\\Device\\NetBT\_Tcpip\_{A91C1FCE-AC2F-4F7F-8795-A7830CB6AE37}] SEQPACKET 4

ConnectionlessService False  
GuaranteesDelivery True  
GuaranteesSequencing True  
MaximumAddressSize 20 bytes  
MaximumMessageSize 64000 bytes  
MessageOriented True  
MinimumAddressSize 20 bytes  
PseudoStreamOriented False  
SupportsBroadcasting False  
SupportsConnectData False  
SupportsDisconnectData False  
SupportsEncryption False  
SupportsExpeditedData False  
SupportsGracefulClosing False  
SupportsGuaranteedBandwidth False  
SupportsMulticasting False

Name MSAFD NetBIOS

[\\Device\\NetBT\_Tcpip\_{A91C1FCE-AC2F-4F7F-8795-A7830CB6AE37}] DATAGRAM 4

ConnectionlessService True  
GuaranteesDelivery False  
GuaranteesSequencing False  
MaximumAddressSize 20 bytes  
MaximumMessageSize 64000 bytes  
MessageOriented True  
MinimumAddressSize 20 bytes  
PseudoStreamOriented False  
SupportsBroadcasting True  
SupportsConnectData False  
SupportsDisconnectData False  
SupportsEncryption False  
SupportsExpeditedData False  
SupportsGracefulClosing False  
SupportsGuaranteedBandwidth False  
SupportsMulticasting False

[WinSock]

Item	Value
File	c:\winnt\system32\winsock.dll
Version	3.10
Size	2.80 KB (2,864 bytes)
File	c:\winnt\system32\wsock32.dll
Version	5.00.2152.1
Size	21.27 KB (21,776 bytes)

[Ports]

[ Following are sub-categories of this main category ]

[Serial]

Item	Value
Name	COM1
Status	OK
PNP Device ID	ACPI\PNP0501\1
Maximum Input Buffer Size	0
Maximum Output Buffer Size	False
Settable Baud Rate	True
Settable Data Bits	True
Settable Flow Control	True
Settable Parity	True
Settable Parity Check	True
Settable Stop Bits	True
Settable RLSD	True
Supports RLSD	True
Supports 16 Bit Mode	False
Supports Special Characters	False
Baud Rate	9600
Bits/Byte	8
Stop Bits	1
Parity	None
Busy	0
Abort Read/Write on Error	0
Binary Mode Enabled-1	0
Continue XMit on XOff	0
CTS Outflow Control	0
Discard NULL Bytes	0
DSR Outflow Control	0
DSR Sensitivity	0
DTR Flow Control Type	Enable
EOF Character	0
Error Replace Character	0
Error Replacement Enabled	0
Event Character	0
Parity Check Enabled	0
RTS Flow Control Type	Enable
XOff Character	19
XOffXMit Threshold	512
XOn Character	17
XOnXMit Threshold	2048
XOnXOff InFlow Control	0
XOnXOff OutFlow Control	0
IRQ Number	4
I/O Port	0x03F8-0x03FF
Driver	c:\winnt\system32\drivers\serial.sys (62448, 5.00.2134.1)

Item	Value
Name	COM2
Status	OK
PNP Device ID	ACPI\PNP0501\2
Maximum Input Buffer Size	0
Maximum Output Buffer Size	False
Settable Baud Rate	True
Settable Data Bits	True
Settable Flow Control	True
Settable Parity	True
Settable Parity Check	True
Settable Stop Bits	True
Settable RLSD	True
Supports RLSD	True
Supports 16 Bit Mode	False
Supports Special Characters	False
Baud Rate	9600
Bits/Byte	8
Stop Bits	1
Parity	None

Busy 0  
 Abort Read/Write on Error 0  
 Binary Mode Enabled-1  
 Continue XMit on XOff 0  
 CTS Outflow Control 0  
 Discard NULL Bytes 0  
 DSR Outflow Control 0  
 DSR Sensitivity 0  
 DTR Flow Control Type Enable  
 EOF Character 0  
 Error Replace Character 0  
 Error Replacement Enabled 0  
 Event Character 0  
 Parity Check Enabled 0  
 RTS Flow Control Type Enable  
 XOff Character 19  
 XOffXMit Threshold 512  
 XOn Character 17  
 XOnXMit Threshold 2048  
 XOnXOff InFlow Control 0  
 XOnXOff OutFlow Control 0  
 IRQ Number 3  
 I/O Port 0x02F8-0x02FF  
 Driver c:\winnt\system32\drivers\serial.sys (62448, 5.00.2134.1)

[Parallel]

Item	Value
Name	LPT1
PNP Device ID	ACPI\PNP0400\1

[Storage]

[ Following are sub-categories of this main category ]

[Drives]

Item	Value
Drive A:	
Description	3 1/2 Inch Floppy Drive
Drive C:	
Description	Local Fixed Disk
Compressed	False
File System	NTFS
Size	16.94 GB (18,186,059,776 bytes)
Free Space	14.45 GB (15,512,948,736 bytes)
Volume Name	
Volume Serial Number	A8E078EC
Partition Disk #0, Partition #0	
Partition Size	16.94 GB (18,186,061,824 bytes)
Starting Offset	32256 bytes
Drive Description	Disk drive
Drive Manufacturer	(Standard disk drives)
Drive Model	IBM-PSG ST318436LW !# SCSI Disk Device
Drive BytesPerSector	512
Drive MediaLoaded	True
Drive MediaType	Fixed hard disk media
Drive Partitions	1
Drive SCSI Bus	0
Drive SCSI Logical Unit	0
Drive SCSI Port	2
Drive SCSI Target Id	0
Drive SectorsPerTrack	63
Drive Size	18194319360 bytes
Drive TotalCylinders	2212

Drive TotalSectors 35535780  
 Drive TotalTracks 564060  
 Drive TracksPerCylinder 255

[SCSI]

Item	Value
Name	Adaptec AIC-7892 Ultra160/m PCI SCSI Card
Caption	Adaptec AIC-7892 Ultra160/m PCI SCSI Card
Driver	adpu160m
Status	OK
PNP Device ID	
PCI\VEN_9005&DEV_008F&SUBSYS_008F9005&REV_02\3&13C0B0C5&0&18	
Device ID	
PCI\VEN_9005&DEV_008F&SUBSYS_008F9005&REV_02\3&13C0B0C5&0&18	
Device Map	Not Available
Index	Not Available
Max Number Controlled	Not Available
IRQ Number	28
I/O Port	0x2100-0x21FF
Driver	c:\winnt\system32\drivers\adpu160m.sys (64432, v3.10a)

[Printing]

Name	Port Name	Server Name
No printing information		

[Problem Devices]

Device	PNP Device ID	Error Code
Not Available	ACPI\IBM37D1\4&23FD4C84&0	28

[USB]

Device	PNP Device ID
Standard OpenHCD USB Host Controller	
PCI\VEN_1166&DEV_0220&SUBSYS_02201166&REV_04\3&267A616A&0&7A	
USB Root Hub	USB\ROOT_HUB\4&372644EA&0

[Software Environment]

[ Following are sub-categories of this main category ]

[Drivers]

Name	Description	File	Type	Started
Start Mode	State	Status	Error Control	
Accept Pause	Accept Stop			
abiosdsk	Abiosdsk	Not Available	Kernel Driver	
False	Disabled	Stopped	OK	Ignore
False				False
abp480n5	abp480n5	Not Available	Kernel Driver	
False	Disabled	Stopped	OK	Normal
False				False
acpi	Microsoft ACPI Driver			
c:\winnt\system32\drivers\acpi.sys			Kernel Driver	
True	Boot	Running	OK	Normal
True				False
acpiec	ACPIEC	c:\winnt\system32\drivers\acpiec.sys	Kernel Driver	
Kernel Driver	False	Disabled	Stopped	OK
Normal	False	False		

adpu160m	adpu160m	c:\winnt\system32\drivers\adpu160m.sys	Kernel Driver	True	Boot	Running	OK	
Normal	False	True						
afd	AFD Networking Support Environment		Kernel Driver	True				
c:\winnt\system32\drivers\afd.sys			Kernel Driver	True				
Auto	Running	OK	Normal	False	True			
aha154x	Aha154x	Not Available	Kernel Driver					
False	Disabled	Stopped	OK	Normal	False			
False								
aic116x	aic116x	Not Available	Kernel Driver					
False	Disabled	Stopped	OK	Normal	False			
False								
aic78u2	aic78u2	Not Available	Kernel Driver					
False	Disabled	Stopped	OK	Normal	False			
False								
aic78xx	aic78xx	Not Available	Kernel Driver					
False	Disabled	Stopped	OK	Normal	False			
False								
ami0nt	ami0nt	Not Available	Kernel Driver					
False	Disabled	Stopped	OK	Normal	False			
False								
amsint	amsint	Not Available	Kernel Driver					
False	Disabled	Stopped	OK	Normal	False			
False								
asc	asc	Not Available	Kernel Driver					
False	Disabled	Stopped	OK	Normal	False			
False								
asc3350p	asc3350p	Not Available	Kernel Driver					
False	Disabled	Stopped	OK	Normal	False			
False								
asc3550	asc3550	Not Available	Kernel Driver					
False	Disabled	Stopped	OK	Normal	False			
False								
asynmac	RAS Asynchronous Media Driver		Kernel Driver					
c:\winnt\system32\drivers\asynmac.sys			Kernel Driver					
False	Manual	Stopped	OK	Normal	False			
False								
atapi	Standard IDE/ESDI Hard Disk Controller		Kernel Driver					
c:\winnt\system32\drivers\atapi.sys			Kernel Driver					
True	Boot	Running	OK	Normal	False			
True								
atdisk	Atdisk	Not Available	Kernel Driver					
False	Disabled	Stopped	OK	Ignore	False			
False								
atmarpc	ATM ARP Client Protocol		Kernel Driver					
c:\winnt\system32\drivers\atmarpc.sys			Kernel Driver					
False	Manual	Stopped	OK	Normal	False			
False								
audstub	Audio Stub Driver		Kernel Driver					
c:\winnt\system32\drivers\audstub.sys			Kernel Driver					
True	Manual	Running	OK	Normal	False			
True								
beep	Beep	c:\winnt\system32\drivers\beep.sys	Kernel Driver	True	System	Running	OK	
Normal	False	True						
buslogic	BusLogic	Not Available	Kernel Driver					
False	Disabled	Stopped	OK	Normal	False			
False								
cd20xrnt	cd20xrnt	Not Available	Kernel Driver					
False	Disabled	Stopped	OK	Normal	False			
False								
cdaudio	Cdaudio	c:\winnt\system32\drivers\cdaudio.sys	Kernel Driver	False	System	Stopped	OK	
Ignore	False	False						
cdfs	Cdfs	c:\winnt\system32\drivers\cdfs.sys	System Driver	True	Disabled	Running	OK	File
Normal	False	True						
cdrom	CD-ROM Driver	c:\winnt\system32\drivers\cdrom.sys	Kernel Driver	True	System	Running	OK	
Normal	False	True						
changer	Changer	Not Available	Kernel Driver					
False	System	Stopped	OK	Ignore	False			
False								
cpqarray	Cpqarray	Not Available	Kernel Driver					
False	Disabled	Stopped	OK	Normal	False			
False								
cpqarry2	cpqarry2	Not Available	Kernel Driver					
False	Disabled	Stopped	OK	Normal	False			
False								
cpqcalm	cpqcalm	Not Available	Kernel Driver					
False	Disabled	Stopped	OK	Normal	False			
False								
cpqfws2e	cpqfws2e	Not Available	Kernel Driver					
False	Disabled	Stopped	OK	Normal	False			
False								
dac960nt	dac960nt	Not Available	Kernel Driver					
False	Disabled	Stopped	OK	Normal	False			
False								
deckzpsx	deckzpsx	Not Available	Kernel Driver					
False	Disabled	Stopped	OK	Normal	False			
False								
dfsdriver	DfsDriver	c:\winnt\system32\drivers\dfs.sys	File System Driver	True	Boot	Running	OK	Normal
False	True							
disk	Disk Driver	c:\winnt\system32\drivers\disk.sys	Kernel Driver	True	Boot	Running	OK	
Normal	False	True						
diskperf	Diskperf	c:\winnt\system32\drivers\diskperf.sys	Kernel Driver	True	Boot	Running	OK	
Normal	False	True						
dmboot	dmboot	c:\winnt\system32\drivers\dmboot.sys	Kernel Driver	False	Disabled	Stopped	OK	
Normal	False	False						
dmio	Logical Disk Manager Driver	c:\winnt\system32\drivers\dmio.sys	Kernel Driver	True	Boot	Running	OK	Normal
True	Boot	Running	OK	Normal	False			
True								
dmload	dmload	c:\winnt\system32\drivers\dmload.sys	Kernel Driver	True	Boot	Running	OK	
Normal	False	True						
e100b	Intel(R) PRO Adapter Driver	c:\winnt\system32\drivers\e100bnt5.sys	Kernel Driver	True	Manual	Running	OK	Normal
True	Manual	Running	OK	Normal	False			
efs	EFS	c:\winnt\system32\drivers\efs.sys	File System Driver	True	Disabled	Running	OK	Normal
False	True							
fastfat	Fastfat	c:\winnt\system32\drivers\fastfat.sys	System Driver	True	Disabled	Running	OK	File
Normal	False	True						
fd16_700	Fd16_700	Not Available	Kernel Driver					
False	Disabled	Stopped	OK	Normal	False			
False								
fdc	Floppy Disk Controller Driver	c:\winnt\system32\drivers\fdc.sys	Kernel Driver	True	Manual	Running	OK	Normal
Manual	Running	OK	Normal	False	True			
fireport	fireport	Not Available	Kernel Driver					
False	Disabled	Stopped	OK	Normal	False			
False								
flashpnt	flashpnt	Not Available	Kernel Driver					
False	Disabled	Stopped	OK	Normal	False			
False								
flpydisk	Floppy Disk Driver	c:\winnt\system32\drivers\flpydisk.sys	Kernel Driver					

True	Manual	Running	OK	Normal	False	mountmgr	MountMgr	c:\winnt\system32\drivers\mountmgr.sys	
True						Kernel Driver	True	Boot	Running OK
ftdisk	Volume Manager Driver					Normal	False	True	
c:\winnt\system32\drivers\ftdisk.sys				Kernel Driver					
True	Boot	Running	OK	Normal	False	mraid35x	mraid35x	Not Available	Kernel Driver
True						False	Disabled	Stopped	OK
True						Normal	False		
gpc	Generic Packet Classifier					mrx smb	MRXSMB	c:\winnt\system32\drivers\mrx smb.sys	File
c:\winnt\system32\drivers\msgpc.sys				Kernel Driver		System Driver	True	System	Running OK
True	Manual	Running	OK	Normal	False	Normal	False	True	
True						msfs	Msfs	c:\winnt\system32\drivers\msfs.sys	File
i8042prt	i8042 Keyboard and PS/2 Mouse Port Driver					System Driver	True	System	Running OK
c:\winnt\system32\drivers\i8042prt.sys				Kernel Driver		Normal	False	True	
True	System	Running	OK	Normal	False	mskssrv	Microsoft Streaming Service Proxy		
True						c:\winnt\system32\drivers\mskssrv.sys		Kernel Driver	
ibmf6	IBM 10/100 Ethernet PCI Adapter NT Driver					False	Manual	Stopped	OK
c:\winnt\system32\drivers\ibmfent5.sys				Kernel Driver		False			
False	Manual	Stopped	OK	Normal	False	mspclock	Microsoft Streaming Clock Proxy		
False						c:\winnt\system32\drivers\mspclock.sys		Kernel Driver	
ini910u	ini910u	Not Available		Kernel Driver		False	Manual	Stopped	OK
False	Disabled	Stopped	OK	Normal	False	False			
False						mspqm	Microsoft Streaming Quality Manager Proxy		
intelide	IntelIde	Not Available		Kernel Driver		c:\winnt\system32\drivers\mspqm.sys		Kernel Driver	
False	Disabled	Stopped	OK	Normal	False	False	Manual	Stopped	OK
False						False			
ipfilterdriver	IP Traffic Filter Driver					mup	Mup	c:\winnt\system32\drivers\mup.sys	File
c:\winnt\system32\drivers\ipfltdrv.sys				Kernel Driver		System Driver	True	Boot	Running OK
False	Manual	Stopped	OK	Normal	False	Normal	False	True	
False						nrc710	Nrc710	Not Available	Kernel Driver
ipinip	IP in IP Tunnel Driver			c:\winnt\system32\drivers\ipinip.sys		False	Disabled	Stopped	OK
Kernel Driver	False	Manual	Stopped	OK		False			
Normal	False	False				ndis	NDIS System Driver		
ipnat	IP Network Address Translator					Kernel Driver	True	Boot	Running OK
c:\winnt\system32\drivers\ipnat.sys				Kernel Driver		Normal	False	True	
False	Manual	Stopped	OK	Normal	False	ndistapi	Remote Access NDIS TAPI Driver		
False						c:\winnt\system32\drivers\ndistapi.sys		Kernel Driver	
ipsecc	IPSEC driver			c:\winnt\system32\drivers\ipsecc.sys		True	Manual	Running	OK
Kernel Driver	True	Manual	Running	OK		True			
Normal	False	True				ndiswan	Remote Access NDIS WAN Driver		
ipsraidn	ipsraidn	Not Available		Kernel Driver		c:\winnt\system32\drivers\ndiswan.sys		Kernel Driver	
False	Disabled	Stopped	OK	Normal	False	True	Manual	Running	OK
False						True			
isapnp	PnP ISA/EISA Bus Driver					ndproxy	NDIS Proxy		
c:\winnt\system32\drivers\isapnp.sys				Kernel Driver		c:\winnt\system32\drivers\ndproxy.sys		Kernel Driver	
True	Boot	Running	OK	Critical	False	True	Manual	Running	OK
True						True			
kbdclass	Keyboard Class Driver					netbios	NetBIOS Interface		
c:\winnt\system32\drivers\kbdclass.sys				Kernel Driver		File System Driver	True	System	Running OK
True	System	Running	OK	Normal	False	Normal	False	True	
True						netbt	NetBios over Tcpip		
ksecdd	KSecDD	c:\winnt\system32\drivers\ksecdd.sys		Kernel Driver		Kernel Driver	True	System	Running OK
Kernel Driver	True	Boot	Running	OK		Normal	False	True	
Normal	False	True				netdetect	NetDetect		
lbrtfdc	lbrtfdc	Not Available		Kernel Driver		Kernel Driver	False	Manual	Stopped
False	System	Stopped	OK	Ignore	False	Normal	False	False	
False						npfs	Npfs		
ip6nds35	Ip6nds35	Not Available		Kernel Driver		System Driver	True	System	Running OK
False	Disabled	Stopped	OK	Normal	False	Normal	False	True	
False						ntfs	Ntfs		
mnmdd	mnmdd	c:\winnt\system32\drivers\mnmdd.sys		Kernel Driver		System Driver	True	Disabled	Running OK
Kernel Driver	True	System	Running	OK		Normal	False	True	
Ignore	False	True				null	Null		
modem	Modem	c:\winnt\system32\drivers\modem.sys		Kernel Driver		Kernel Driver	True	System	Running OK
Kernel Driver	False	Manual	Stopped	OK		Normal	False	True	
Ignore	False	False				nwlnkflt	IPX Traffic Filter Driver		
mouclass	Mouse Class Driver					c:\winnt\system32\drivers\nwlnkflt.sys		Kernel Driver	
c:\winnt\system32\drivers\mouclass.sys				Kernel Driver		False	Manual	Stopped	OK
True	System	Running	OK	Normal	False	False			
True									

nwlnkfdw	IPX Traffic Forwarder Driver								
c:\winnt\system32\drivers\nwlnkfdw.sys		Kernel Driver							
False	Manual	Stopped	OK	Normal	False				
False									
openhci	Microsoft USB Open Host Controller Driver								
c:\winnt\system32\drivers\openhci.sys		Kernel Driver							
True	Manual	Running	OK	Normal	False				
True									
parallel	Parallel class driver	c:\winnt\system32\drivers\parallel.sys							
Kernel Driver	True	Manual	Running	OK					
Normal	False	True							
parport	Parallel port driver	c:\winnt\system32\drivers\parport.sys							
Kernel Driver	True	System	Running	OK					
Ignore	False	True							
partmgr	PartMgr	c:\winnt\system32\drivers\partmgr.sys							
Kernel Driver	True	Boot	Running	OK					
Normal	False	True							
parvdm	ParVdm	c:\winnt\system32\drivers\parvdm.sys							
Kernel Driver	True	Auto	Running	OK					
Ignore	False	True							
pci	PCI Bus Driver	c:\winnt\system32\drivers\pci.sys							
Kernel Driver	True	Boot	Running	OK					
Critical	False	True							
pcidump	PCIDump	Not Available		Kernel Driver					
False	System	Stopped	OK	Ignore	False				
False									
pciide	PCIIde	c:\winnt\system32\drivers\pciide.sys							
Kernel Driver	True	Boot	Running	OK					
Normal	False	True							
pcmcia	Pcmcia	c:\winnt\system32\drivers\pcmcia.sys							
Kernel Driver	False	Disabled	Stopped	OK					
Normal	False	False							
pcnet	PCNET Adapter Driver								
c:\winnt\system32\drivers\pcntn5m.sys		Kernel Driver							
False	Manual	Stopped	OK	Normal	False				
False									
pdcomp	PDCOMP	Not Available		Kernel Driver					
False	Manual	Stopped	OK	Ignore	False				
False									
pdframe	PDFRAME	Not Available		Kernel Driver					
False	Manual	Stopped	OK	Ignore	False				
False									
pdreli	PDRELI	Not Available		Kernel Driver					
False	Manual	Stopped	OK	Ignore	False				
False									
pdframe	PDRFRAME	Not Available		Kernel Driver					
False	Manual	Stopped	OK	Ignore	False				
False									
pptpminiport	WAN Miniport (PPTP)								
c:\winnt\system32\drivers\rasppptp.sys		Kernel Driver							
True	Manual	Running	OK	Normal	False				
True									
ptilink	Direct Parallel Link Driver								
c:\winnt\system32\drivers\ptilink.sys		Kernel Driver							
True	Manual	Running	OK	Normal	False				
True									
ql1080	ql1080	Not Available		Kernel Driver					
False	Disabled	Stopped	OK	Normal	False				
False									
ql10wnt	Ql10wnt	Not Available		Kernel Driver					
False	Disabled	Stopped	OK	Normal	False				
False									
ql1240	ql1240	Not Available		Kernel Driver					
False	Disabled	Stopped	OK	Normal	False				
False									
ql2100	ql2100	Not Available		Kernel Driver					
False	Disabled	Stopped	OK	Normal	False				
False									
rasacd	Remote Access Auto Connection Driver								
c:\winnt\system32\drivers\rasacd.sys		Kernel Driver							
True	System	Running	OK	Normal	False				
True									
rasl2tp	WAN Miniport (L2TP)								
c:\winnt\system32\drivers\rasl2tp.sys		Kernel Driver							
True	Manual	Running	OK	Normal	False				
True									
raspti	Direct Parallel	c:\winnt\system32\drivers\raspti.sys							
Kernel Driver	True	Manual	Running	OK					
Normal	False	True							
rca	Microsoft Streaming Network Raw Channel Access								
c:\winnt\system32\drivers\rca.sys	Kernel Driver	False							
Manual	Stopped	OK	Normal	False	False				
rdbss	Rdbss	c:\winnt\system32\drivers\rdbss.sys							File
System Driver	True	System	Running	OK					
Normal	False	True							
rdpwd	RDPWD	c:\winnt\system32\drivers\rdpwd.sys							
Kernel Driver	False	Manual	Stopped	OK					
Ignore	False	False							
redbook	Digital CD Audio Playback Filter Driver								
c:\winnt\system32\drivers\redbook.sys		Kernel Driver							
False	System	Stopped	OK	Normal	False				
False									
s3inc	S3Inc	c:\winnt\system32\drivers\s3sav4m.sys							
Kernel Driver	True	Manual	Running	OK					
Ignore	False	True							
serenum	Serenum Filter Driver								
c:\winnt\system32\drivers\serenum.sys		Kernel Driver							
True	Manual	Running	OK	Normal	False				
True									
serial	Serial port driver	c:\winnt\system32\drivers\serial.sys							
Kernel Driver	True	System	Running	OK					
Ignore	False	True							
sfloppy	Sfloppy	c:\winnt\system32\drivers\sfloppy.sys							
Kernel Driver	False	System	Stopped	OK					
Ignore	False	False							
sglfb	sglfb	Not Available		Kernel Driver					
False	System	Stopped	OK	Normal	False				
False									
simbad	Simbad	Not Available		Kernel Driver					
False	Disabled	Stopped	OK	Normal	False				
False									
sparrow	Sparrow	Not Available		Kernel Driver					
False	Disabled	Stopped	OK	Normal	False				
False									
spud	Special Purpose Utility Driver								
c:\winnt\system32\drivers\spud.sys		Kernel Driver							
True	Manual	Running	OK	Normal	False				
True									
srv	Srv	c:\winnt\system32\drivers\srv.sys							File System
Driver	True	Manual	Running	OK	Normal				
False	True								
swenum	Software Bus Driver								
c:\winnt\system32\drivers\swenum.sys		Kernel Driver							
True	Manual	Running	OK	Normal	False				
True									
symc810	symc810	Not Available		Kernel Driver					
False	Disabled	Stopped	OK	Normal	False				
False									
symc8xx	symc8xx	Not Available		Kernel Driver					
False	Disabled	Stopped	OK	Normal	False				
False									
sym_hi	sym_hi	Not Available		Kernel Driver					
False	Disabled	Stopped	OK	Normal	False				
False									
tcpip	TCP/IP Protocol Driver								
c:\winnt\system32\drivers\tcpip.sys		Kernel Driver							

```

True      System      Running OK      Normal False
True
tdasync  TDASYNC      c:\winnt\system32\drivers\tdasync.sys
Kernel Driver      False      Manual Stopped OK
Ignore      False      False
tdipx    TDIPX      c:\winnt\system32\drivers\tdipx.sys
Kernel Driver      False      Manual Stopped OK
Ignore      False      False
tdnetb  TDNETB      c:\winnt\system32\drivers\tdnetb.sys
Kernel Driver      False      Manual Stopped OK
Ignore      False      False
tdpipe  TDPIPE      c:\winnt\system32\drivers\tdpipe.sys
Kernel Driver      False      Manual Stopped OK
Ignore      False      False
tdspix  TDSPIX      c:\winnt\system32\drivers\tdspix.sys
Kernel Driver      False      Manual Stopped OK
Ignore      False      False
tdtcp   TDTCP      c:\winnt\system32\drivers\tdtcp.sys
Kernel Driver      False      Manual Stopped OK
Ignore      False      False
termdd  Terminal Device Driver
c:\winnt\system32\drivers\termdd.sys      Kernel Driver
False      Disabled Stopped OK      Normal False
False
tga     tga      Not Available      Kernel Driver
False      System Stopped OK      Ignore False
False
udfs    Udfs      c:\winnt\system32\drivers\udfs.sys
System Driver      False      Disabled Stopped OK
Normal      False      False
ultra66 ultra66      Not Available      Kernel Driver
False      Disabled Stopped OK      Normal False
False
update  Microcode Update Driver
c:\winnt\system32\drivers\update.sys      Kernel Driver
True      Manual Running OK      Normal False
True
usbhub  Microsoft USB Standard Hub Driver
c:\winnt\system32\drivers\usbhub.sys      Kernel Driver
True      Manual Running OK      Normal False
True
vgasave VgaSave      c:\winnt\system32\drivers\vga.sys
Kernel Driver      True      System Running OK
Ignore      False      True
wanarp  Remote Access IP ARP Driver
c:\winnt\system32\drivers\wanarp.sys      Kernel Driver
True      Manual Running OK      Normal False
True
wdica   WDICA      Not Available      Kernel Driver
False      Manual Stopped OK      Ignore False
False

```

[Environment Variables]

```

Variable Value      User Name
ComSpec  %SystemRoot%\system32\cmd.exe      <SYSTEM>
Os2LibPath %SystemRoot%\system32\os2\dll;
<SYSTEM>
Path
%SystemRoot%\system32;%SystemRoot%;%SystemRoot%\System32\
Wbem;C:\Program Files\Microsoft SQL Server\80\Tools\BINN
<SYSTEM>
windir  %SystemRoot%      <SYSTEM>
OS      Windows_NT      <SYSTEM>
PROCESSOR_ARCHITECTURE x86      <SYSTEM>
PROCESSOR_LEVEL      6      <SYSTEM>
PROCESSOR_IDENTIFIER x86 Family 6 Model 8 Stepping 6,
GenuineIntel      <SYSTEM>

```

```

PROCESSOR_REVISION      0806      <SYSTEM>
NUMBER_OF_PROCESSORS    2      <SYSTEM>
PATH
.COM; .EXE; .BAT; .CMD; .VBS; .VBE; .JS; .JSE; .WSF; .WSH
<SYSTEM>
TEMP %SystemRoot%\TEMP      <SYSTEM>
TMP %SystemRoot%\TEMP      <SYSTEM>
TEMP %USERPROFILE%\Local Settings\Temp
CLIENT10\Administrator
TMP %USERPROFILE%\Local Settings\Temp
CLIENT10\Administrator

[Jobs]

[ Following are sub-categories of this main category ]

[Print]

Document Size      Owner      Notify      Status      Time
Submitted Start Time Until TimeElapsed Time      Pages Printed
Job ID      Priority      ParametersDriver Name      Print Processor
Host Print Queue      Data Type Name
No print jobs

[Network Connections]

File
Local Name      Remote Name      Type      Status
User Name
No network connections information

[Running Tasks]

Name      Path      Process IDPriority      Min Working Set
Max Working Set      Start Time Version      Size      File Date
system idle process      Not Available      0      0      Not
Available      Not Available      Not Available      Unknown
Unknown      Unknown
system      Not Available      8      8      0
1413120      Not Available      Unknown      Unknown      Unknown
smss.exe      c:\winnt\system32\smss.exe      168      11
204800      1413120      3/19/2001 1:06:11 PM      5.00.2170.1
44.27 KB (45,328 bytes)      12/7/1999 7:00:00 AM
csrss.exe      Not Available      188      13      Not Available
Not Available      3/19/2001 1:06:16 PM      Unknown
Unknown      Unknown
winlogon.exe      c:\winnt\system32\winlogon.exe      184      13
204800      1413120      3/19/2001 1:06:17 PM      5.00.2182.1
173.27 KB (177,424 bytes)      12/7/1999 7:00:00 AM
services.exe      c:\winnt\system32\services.exe      236      9
204800      1413120      3/19/2001 1:06:18 PM      5.00.2134.1
86.77 KB (88,848 bytes)      12/7/1999 7:00:00 AM
lsass.exe      c:\winnt\system32\lsass.exe      248      13
204800      1413120      3/19/2001 1:06:18 PM      5.00.2184.1
32.77 KB (33,552 bytes)      12/7/1999 7:00:00 AM
svchost.exe      c:\winnt\system32\svchost.exe      408      8
204800      1413120      3/19/2001 1:06:22 PM      5.00.2134.1
7.77 KB (7,952 bytes)12/7/1999 7:00:00 AM
spoolsv.exe      c:\winnt\system32\spoolsv.exe      436      8
204800      1413120      3/19/2001 1:06:23 PM      5.00.2161.1
43.77 KB (44,816 bytes)      2/5/2001 9:57:15 AM
msdtc.exe      c:\winnt\system32\msdtc.exe      464      8
204800      1413120      3/19/2001 1:06:23 PM      1999.9.3421.3
6.77 KB (6,928 bytes)2/5/2001 10:44:17 AM
svchost.exe      c:\winnt\system32\svchost.exe      596      8
204800      1413120      3/19/2001 1:06:25 PM      5.00.2134.1
7.77 KB (7,952 bytes)12/7/1999 7:00:00 AM

```

```

llsrv.exe c:\winnt\system32\llsrv.exe 620 9
204800 1413120 3/19/2001 1:06:25 PM 5.00.2167.1
114.27 KB (117,008 bytes) 12/7/1999 7:00:00 AM
regsvc.exe c:\winnt\system32\regsvc.exe 664 8
204800 1413120 3/19/2001 1:06:26 PM 5.00.2155.1
65.27 KB (66,832 bytes) 12/7/1999 7:00:00 AM
rsvp.exe c:\winnt\system32\rsvp.exe 704 8
204800 1413120 3/19/2001 1:06:26 PM 5.00.2167.1
172.77 KB (176,912 bytes) 12/7/1999 7:00:00 AM
mstask.exec c:\winnt\system32\mstask.exe 724 8
204800 1413120 3/19/2001 1:06:27 PM 4.71.2137.1
115.27 KB (118,032 bytes) 2/5/2001 3:49:15 PM
tcpvcs.exe c:\winnt\system32\tcpvcs.exe 788 8
204800 1413120 3/19/2001 1:06:28 PM 5.00.2134.1
24.77 KB (25,360 bytes) 12/7/1999 7:00:00 AM
winmgmt.exe c:\winnt\system32\wbem\winmgmt.exe 880
8 204800 1413120 3/19/2001 1:06:34 PM
1.50.1085.0001 188.05 KB (192,567 bytes) 12/7/1999
7:00:00 AM
inetinfo.exe c:\winnt\system32\inetrv\inetinfo.exe 916
8 204800 1413120 3/19/2001 1:06:35 PM
5.00.0984 14.27 KB (14,608 bytes) 2/5/2001 10:44:54 AM
dfssvc.exe c:\winnt\system32\dfssvc.exe 948 8
204800 1413120 3/19/2001 1:06:35 PM 5.00.2191.1
85.27 KB (87,312 bytes) 12/7/1999 7:00:00 AM
explorer.exe c:\winnt\explorer.exe 1052 8
204800 1413120 3/19/2001 1:06:38 PM
5.00.2920.0000 232.77 KB (238,352 bytes) 12/7/1999
7:00:00 AM
promon.exe c:\winnt\system32\promon.exe 1088 8
204800 1413120 3/19/2001 1:06:39 PM 1.11
28.50 KB (29,184 bytes) 2/8/2001 12:31:42 PM
mmc.exe c:\winnt\system32\mmc.exe 1252 8
204800 1413120 3/19/2001 1:06:54 PM 5.00.2153.1
589.27 KB (603,408 bytes) 12/7/1999 7:00:00 AM
svchost.exe c:\winnt\system32\svchost.exe 1308 8
204800 1413120 3/19/2001 1:06:58 PM 5.00.2134.1
7.77 KB (7,952 bytes) 12/7/1999 7:00:00 AM
mdm.exe c:\winnt\system32\mdm.exe 1264 8
204800 1413120 3/19/2001 1:08:54 PM 6.00.8424
121.29 KB (124,200 bytes) 2/5/2001 10:46:52 AM

```

[Loaded Modules]

```

Name Version Size File Date Manufacturer Path
mdm.exe 6.00.8424 121.29 KB (124,200 bytes) 2/5/2001
10:46:52 AM Microsoft Corporation
c:\winnt\system32\mdm.exe
tapisrv.dll 5.00.2186.1 168.77 KB (172,816 bytes)
12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\tapisrv.dll
olepro32.dll 5.0.4512 160.27 KB (164,112 bytes)
12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\olepro32.dll
dmocx.dll 5.00.2134.1 23.27 KB (23,824 bytes)
12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\dmocx.dll
wbemprox.dll 1.50.1085.0001 40.05 KB (41,016 bytes)
12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\wbem\wbemprox.dll
rassapi.dll 5.00.2188.1 14.27 KB (14,608 bytes)
12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\rassapi.dll
adsnt.dll 5.00.2191.1 194.27 KB (198,928 bytes)
12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\adsnt.dll

```

```

dbghelp.dll 5.00.2195.1 159.27 KB (163,088
bytes) 12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\dbghelp.dll
localsec.dll 5.00.2134.1 227.27 KB (232,720
bytes) 12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\localsec.dll
devmgr.dll 5.00.2166.1 215.77 KB (220,944 bytes)
12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\devmgr.dll
filemgmt.dll 5.00.2134.1 287.27 KB (294,160
bytes) 12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\filemgmt.dll
pdh.dll 5.00.2174.1 143.27 KB (146,704 bytes)
12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\pdh.dll
smlogcfg.dll 5.00.2163.1 273.27 KB (279,824
bytes) 12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\smlogcfg.dll
cabinet.dll 5.00.2147.1 54.77 KB (56,080 bytes)
12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\cabinet.dll
msinfo32.dll 5.00.2177.1 312.27 KB (319,760
bytes) 2/5/2001 3:49:22 PM Microsoft Corporation
c:\program files\common files\microsoft shared\msinfo\msinfo32.dll
riched20.dll 5.30.23.1200 421.27 KB (431,376
bytes) 12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\riched20.dll
riched32.dll 5.00.2134.1 3.77 KB (3,856 bytes)
12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\riched32.dll
els.dll 5.00.2175.1 151.27 KB (154,896 bytes)
12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\els.dll
ntmsmgr.dll 1,0,0,1 427.77 KB (438,032 bytes)
12/7/1999 7:00:00 AM Microsoft Corporation and
HighGround Systems, Inc. c:\winnt\system32\ntmsmgr.dll
mmfutil.dll 1.50.1085.0000 32.06 KB (32,829 bytes)
12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\mmfutil.dll
logdrive.dll 1.50.1085.0000 200.06 KB (204,863
bytes) 12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\logdrive.dll
dfrgres.dll 5.00.2150.1 27.50 KB (28,160 bytes)
12/7/1999 7:00:00 AM Executive Software International, Inc.
c:\winnt\system32\dfrgres.dll
dfrgsnap.dll 5.00.2150.1 41.77 KB (42,768 bytes)
12/7/1999 7:00:00 AM Executive Software International, Inc.
c:\winnt\system32\dfrgsnap.dll
dmdskres.dll 2191.1.296.2 119.00 KB (121,856
bytes) 12/7/1999 7:00:00 AM Microsoft Corp.,
VERITAS Software c:\winnt\system32\dmdskres.dll
dmutil.dll 2191.1.296.2 41.77 KB (42,768 bytes)
12/7/1999 7:00:00 AM VERITAS Software Corp.
c:\winnt\system32\dmutil.dll
ntmsapi.dll 5.00.1948.1 50.27 KB (51,472 bytes)
12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\ntmsapi.dll
dmdskmgr.dll 2191.1.296.2 158.77 KB (162,576
bytes) 12/7/1999 7:00:00 AM Microsoft Corp.,
VERITAS Software c:\winnt\system32\dmdskmgr.dll
mycomput.dll 5.00.2134.1 107.77 KB (110,352
bytes) 12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\mycomput.dll
mmcmdmgr.dll 5.00.2178.1 815.27 KB (834,832
bytes) 12/7/1999 7:00:00 AM Microsoft Corporation
c:\winnt\system32\mmcmdmgr.dll

```



mmc.exe	5.00.2153.1	589.27 KB (603,408 bytes)	
12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\mmc.exe			
promon.exe	1.11	28.50 KB (29,184 bytes)	
2/8/2001 12:31:42 PM		Intel Corporation	
c:\winnt\system32\promon.exe			
jscript.dll	5.1.0.4615	476.06 KB (487,481 bytes)	12/7/1999
7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\jscript.dll			
imm32.dll	5.00.2180.1	93.77 KB (96,016 bytes)	
12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\imm32.dll			
mshtml.dll	5.00.2920.0000	230.27 KB (235,792 bytes)	
12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\mshtml.dll			
thumbvw.dll	5.00.2920.0000	183.27 KB (187,664 bytes)	
12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\thumbvw.dll			
usp10.dll	1.0325.2180.1	307.77 KB (315,152 bytes)	
12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\usp10.dll			
faxshell.dll	5.00.2134.1	8.27 KB (8,464 bytes)	12/7/1999
7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\faxshell.dll			
msacm32.dll	5.00.2134.1	65.27 KB (66,832 bytes)	
12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\msacm32.dll			
avifil32.dll	5.00.2134.1	76.27 KB (78,096 bytes)	
12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\avifil32.dll			
msvfw32.dll	5.00.2134.1	113.77 KB (116,496 bytes)	
12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\msvfw32.dll			
docprop2.dll	5.00.2178.1	297.77 KB (304,912 bytes)	
12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\docprop2.dll			
webvw.dll	5.00.2920.0000	1.06 MB (1,115,408 bytes)	
12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\webvw.dll			
msls31.dll	3.10.337.0	145.27 KB (148,752 bytes)	12/7/1999
7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\msls31.dll			
msdbg.dll	6.00.8424	67.50 KB (69,120 bytes)	2/5/2001
10:46:52 AM		Microsoft Corporation	
c:\winnt\system32\msdbg.dll			
shdoclc.dll	5.00.2920.0000	324.50 KB (332,288 bytes)	
12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\shdoclc.dll			
pdm.dll	6.00.8424	179.27 KB (183,574 bytes)	2/5/2001
10:46:52 AM		Microsoft Corporation	
c:\winnt\system32\pdm.dll			
mshtml.dll	5.00.2920.0000	2.25 MB (2,357,008 bytes)	
12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\mshtml.dll			
mlang.dll	5.00.2920.0000	510.77 KB (523,024 bytes)	
12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\mlang.dll			
urlmon.dll	5.00.2920.0000	426.77 KB (437,008 bytes)	
12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\urlmon.dll			
browselc.dll	5.00.2920.0000	34.50 KB (35,328 bytes)	
12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\browselc.dll			
wininet.dll	5.00.2920.0000	456.77 KB (467,728 bytes)	
12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\wininet.dll			
linkinfo.dll	5.00.2134.1	15.77 KB (16,144 bytes)	
12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\linkinfo.dll			
powrprof.dll	5.00.2920.0000	13.27 KB (13,584 bytes)	
12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\powrprof.dll			
batmeter.dll	5.00.2920.0000	20.27 KB (20,752 bytes)	
12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\batmeter.dll			
stobject.dll	5.00.2144.1	81.77 KB (83,728 bytes)	
12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\stobject.dll			
msi.dll	1.10.1029.0	1.71 MB (1,794,320 bytes)	
12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\msi.dll			
webcheck.dll	5.00.2920.0000	251.77 KB (257,808 bytes)	
12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\webcheck.dll			
ntshrui.dll	5.00.2134.1	46.77 KB (47,888 bytes)	
12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\ntshrui.dll			
mydocs.dll	5.00.2920.0000	55.77 KB (57,104 bytes)	
12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\mydocs.dll			
browseui.dll	5.00.2920.0000	793.27 KB (812,304 bytes)	
12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\browseui.dll			
shdocvw.dll	5.00.2920.0000	1.05 MB (1,104,144 bytes)	
12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\shdocvw.dll			
explorer.exe	5.00.2920.0000	232.77 KB (238,352 bytes)	
12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\explorer.exe			
dfssvc.exe	5.00.2191.1	85.27 KB (87,312 bytes)	
12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\dfssvc.exe			
iislog.dll	5.00.0984	76.27 KB (78,096 bytes)	2/5/2001
10:44:54 AM		Microsoft Corporation	
c:\winnt\system32\inetsrv\iislog.dll			
httpext.dll	0.9.3939.9	418.27 KB (428,304 bytes)	2/5/2001
10:44:54 AM		Microsoft Corporation	
c:\winnt\system32\inetsrv\httpext.dll			
rpcproxy.dll	5.00.2176.1	16.27 KB (16,656 bytes)	
2/5/2001 10:44:12 AM		Microsoft Corporation	
c:\winnt\system32\rpcproxy\rpcproxy.dll			
fpexedll.dll	4.0.2.3406	20.06 KB (20,541 bytes)	
2/5/2001 10:46:46 AM		Microsoft Corporation	
c:\program files\common files\microsoft shared\web server extensions\40\bin\fpexedll.dll			
md5filt.dll	5.00.0984	32.77 KB (33,552 bytes)	2/5/2001
10:44:59 AM		Microsoft Corporation	
c:\winnt\system32\inetsrv\md5filt.dll			
gzip.dll	5.00.0984	30.27 KB (30,992 bytes)	2/5/2001
10:44:58 AM		Microsoft Corporation	
c:\winnt\system32\inetsrv\gzip.dll			
compfilt.dll	5.00.0984	22.27 KB (22,800 bytes)	
2/5/2001 10:44:58 AM		Microsoft Corporation	
c:\winnt\system32\inetsrv\compfilt.dll			
sspifilt.dll	5.00.0984	43.27 KB (44,304 bytes)	2/5/2001
10:44:59 AM		Microsoft Corporation	
c:\winnt\system32\inetsrv\sspifilt.dll			
iscomlog.dll	5.00.0984	24.77 KB (25,360 bytes)	
2/5/2001 10:44:54 AM		Microsoft Corporation	
c:\winnt\system32\inetsrv\iscomlog.dll			
lonsint.dll	5.00.0984	11.77 KB (12,048 bytes)	2/5/2001
10:44:54 AM		Microsoft Corporation	
c:\winnt\system32\inetsrv\lonsint.dll			

inetsloc.dll	5.00.0984	20.27 KB (20,752 bytes)	2/5/2001	rapilib.dll	5.00.2167.1	25.27 KB (25,872 bytes)	
10:44:55 AM		Microsoft Corporation		12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\inetsloc.dll				c:\winnt\system32\rapilib.dll			
iisfecnv.dll	5.00.0984	7.27 KB (7,440 bytes)	2/5/2001 10:44:54 AM	rsvpsp.dll	5.00.2167.1	74.77 KB (76,560 bytes)	
Microsoft Corporation				12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\inetsrv\iisfecnv.dll				c:\winnt\system32\rsvpsp.dll			
isatq.dll	5.00.0984	61.27 KB (62,736 bytes)	2/5/2001	ntmarta.dll	5.00.2158.1	98.77 KB (101,136 bytes)	
10:44:56 AM		Microsoft Corporation		12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\inetsrv\isatq.dll				c:\winnt\system32\ntmarta.dll			
infocomm.dll	5.00.0984	234.27 KB (239,888 bytes)	2/5/2001 10:44:54 AM	provthrd.dll	1.50.1085.0000	68.07 KB (69,708 bytes)	
Microsoft Corporation				2/5/2001 3:49:14 PM		Microsoft Corporation	
c:\winnt\system32\inetsrv\infocomm.dll				c:\winnt\system32\wbem\provthrd.dll			
w3svc.dll	5.00.0984	347.27 KB (355,600 bytes)	2/5/2001	ntevt.dll	1.50.1085.0000	192.06 KB (196,669 bytes)	
10:45:00 AM		Microsoft Corporation		12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\inetsrv\w3svc.dll				c:\winnt\system32\wbem\ntevt.dll			
security.dll	5.00.2154.1	5.77 KB (5,904 bytes)	12/7/1999	framedyn.dll	1.50.1085.0000	164.05 KB (167,992 bytes)	
7:00:00 AM		Microsoft Corporation		12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\security.dll				c:\winnt\system32\wbem\framedyn.dll			
svcxext.dll	5.00.0984	39.77 KB (40,720 bytes)	2/5/2001	cimwin32.dll	1.50.1085.0000	1.03 MB (1,077,306 bytes)	
10:44:55 AM		Microsoft Corporation		12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\inetsrv\svcxext.dll				c:\winnt\system32\wbem\cimwin32.dll			
admexs.dll	5.00.0984	27.77 KB (28,432 bytes)	2/5/2001	wbemsvc.dll	1.50.1085.0000	140.07 KB (143,430 bytes)	
10:44:53 AM		Microsoft Corporation		12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\inetsrv\admexs.dll				c:\winnt\system32\wbem\wbemsvc.dll			
wamreg.dll	5.00.0984	46.27 KB (47,376 bytes)	2/5/2001 10:45:00 AM	wbemess.dll	1.50.1085.0001	352.05 KB (360,503 bytes)	
Microsoft Corporation				12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\inetsrv\wamreg.dll				c:\winnt\system32\wbem\wbemess.dll			
metadata.dll	5.00.0984	70.77 KB (72,464 bytes)	2/5/2001 10:44:54 AM	fastprox.dll	1.50.1085.0001	144.08 KB (147,534 bytes)	
Microsoft Corporation				12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\inetsrv\metadata.dll				c:\winnt\system32\wbem\fastprox.dll			
iismap.dll	5.00.0984	56.27 KB (57,616 bytes)	2/5/2001	wbemcore.dll	1.50.1085.0001	632.05 KB (647,224 bytes)	
10:44:55 AM		Microsoft Corporation		12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\iismap.dll				c:\winnt\system32\wbem\wbemcore.dll			
nsepm.dll	5.00.0984	43.27 KB (44,304 bytes)	2/5/2001	wbemcomn.dll	1.50.1085.0001	684.05 KB (700,472 bytes)	
10:44:54 AM		Microsoft Corporation		12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\inetsrv\nsepm.dll				c:\winnt\system32\wbem\wbemcomn.dll			
coadmin.dll	5.00.0984	39.77 KB (40,720 bytes)	2/5/2001 10:44:56 AM	winnmgmt.exe	1.50.1085.0001	188.05 KB (192,567 bytes)	
Microsoft Corporation				12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\inetsrv\coadmin.dll				c:\winnt\system32\wbem\winnmgmt.exe			
iisadmin.dll	5.00.0984	14.77 KB (15,120 bytes)	2/5/2001 10:44:54 AM	simptcp.dll	5.00.2134.1	19.27 KB (19,728 bytes)	
Microsoft Corporation				2/5/2001 10:44:12 AM		Microsoft Corporation	
c:\winnt\system32\inetsrv\iisadmin.dll				c:\winnt\system32\simptcp.dll			
rpref.dll	5.00.0984	4.27 KB (4,368 bytes)	2/5/2001 10:44:55 AM	tcpsvcs.exe	5.00.2134.1	24.77 KB (25,360 bytes)	
Microsoft Corporation				12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\inetsrv\rpref.dll				c:\winnt\system32\tcpsvcs.exe			
iisrtl.dll	5.00.0984	120.77 KB (123,664 bytes)	2/5/2001	msidle.dll	5.00.2920.0000	6.27 KB (6,416 bytes)	12/7/1999
10:44:55 AM		Microsoft Corporation		7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\iisrtl.dll				c:\winnt\system32\msidle.dll			
inetinfo.exe	5.00.0984	14.27 KB (14,608 bytes)	2/5/2001 10:44:54 AM	mstask.exe	4.71.2137.1	115.27 KB (118,032 bytes)	
Microsoft Corporation				2/5/2001 3:49:15 PM		Microsoft Corporation	
c:\winnt\system32\inetsrv\inetinfo.exe				c:\winnt\system32\mstask.exe			
netui1.dll	5.00.2134.1	210.27 KB (215,312 bytes)	12/7/1999 7:00:00 AM	traffic.dll	5.00.2139.1	30.77 KB (31,504 bytes)	
Microsoft Corporation				12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\netui1.dll				c:\winnt\system32\traffic.dll			
netui0.dll	5.00.2134.1	70.27 KB (71,952 bytes)	12/7/1999 7:00:00 AM	rsvp.exe	5.00.2167.1	172.77 KB (176,912 bytes)	
Microsoft Corporation				12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\netui0.dll				c:\winnt\system32\rsvp.exe			
ntlmanman.dll	5.00.2157.1	35.27 KB (36,112 bytes)	12/7/1999 7:00:00 AM	regsvc.exe	5.00.2155.1	65.27 KB (66,832 bytes)	
Microsoft Corporation				12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\ntlmanman.dll				c:\winnt\system32\regsvc.exe			
perfos.dll	5.00.2155.1	21.27 KB (21,776 bytes)	12/7/1999 7:00:00 AM	llsrpc.dll	5.00.2149.1	45.77 KB (46,864 bytes)	
Microsoft Corporation				12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\perfos.dll				c:\winnt\system32\llsrpc.dll			
psapi.dll	5.00.2134.1	28.27 KB (28,944 bytes)	12/7/1999 7:00:00 AM	llsrv.exe	5.00.2167.1	114.27 KB (117,008 bytes)	
Microsoft Corporation				12/7/1999 7:00:00 AM		Microsoft Corporation	
c:\winnt\system32\psapi.dll				c:\winnt\system32\llsrv.exe			
wshnetbs.dll	5.00.2134.1	7.77 KB (7,952 bytes)	12/7/1999 7:00:00 AM				
Microsoft Corporation							
c:\winnt\system32\wshnetbs.dll							

ntmsdba.dll	5.00.2187.1	167.77 KB (171,792 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\ntmsdba.dll
rasdlg.dll	5.00.2194.1	514.27 KB (526,608 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\rasdlg.dll
netcfgx.dll	5.00.2175.1	533.77 KB (546,576 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\netcfgx.dll
rasmans.dll	5.00.2188.1	146.77 KB (150,288 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\rasmans.dll
netshell.dll	5.00.2176.1	456.77 KB (467,728 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\netshell.dll
netman.dll	5.00.2175.1	88.77 KB (90,896 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\netman.dll
iashlpr.dll	5.00.2184.1	33.27 KB (34,064 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\iashlpr.dll
iasacct.dll	5.00.2134.1	28.27 KB (28,944 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\iasacct.dll
iasuserr.dll	5.00.2134.1	25.77 KB (26,384 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\iasuserr.dll
iasnap.dll	5.00.2134.1	58.77 KB (60,176 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\iasnap.dll
iaspipe.dll	5.00.2134.1	41.77 KB (42,768 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\iaspipe.dll
expsrv.dll	6.0.8540	370.27 KB (379,152 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\expsrv.dll
vbajet32.dll	6.1.8268	30.27 KB (30,992 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\vbajet32.dll
msjtes40.dll	4.00.2927.8	232.27 KB (237,840 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\msjtes40.dll
oledb32r.dll	2.60.6526.0	68.27 KB (69,904 bytes)	2/5/2001 5:04:24 PM	Microsoft Corporation	c:\program files\common files\system\ole db\oledb32r.dll
comdlg32.dll	5.00.2920.0000	236.77 KB (242,448 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\comdlg32.dll
msdart.dll	2.60.6526.0	144.27 KB (147,728 bytes)	2/5/2001 5:04:22 PM	Microsoft Corporation	c:\winnt\system32\msdart.dll
oledb32.dll	2.60.6526.0	448.27 KB (459,024 bytes)	2/5/2001 5:04:24 PM	Microsoft Corporation	c:\program files\common files\system\ole db\oledb32.dll
msjint40.dll	4.00.2927.2	148.27 KB (151,824 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\msjint40.dll
msjter40.dll	4.00.2927.2	52.27 KB (53,520 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\msjter40.dll
mwsstr10.dll	4.00.2927.10	600.27 KB (614,672 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\mwsstr10.dll
msjet40.dll	4.00.2927.4	1.43 MB (1,495,312 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\msjet40.dll
msjetoledb40.dll	4.00.2927.2	340.27 KB (348,432 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\msjetoledb40.dll
iasrad.dll	5.00.2139.1	94.27 KB (96,528 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	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\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
sens.dll	5.00.2163.1	36.77 KB (37,648 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\sens.dll
iaspolicy.dll	5.00.2134.1	25.27 KB (25,872 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\iaspolicy.dll
iassvcs.dll	5.00.2160.1	58.77 KB (60,176 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\iassvcs.dll
iasdo.dll	5.00.2157.1	262.27 KB (268,560 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\iasdo.dll
ntmssvc.dll	5.00.2187.1	390.77 KB (400,144 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\ntmssvc.dll
ias.dll	5.00.2134.1	7.27 KB (7,440 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\ias.dll
es.dll	1999.9.3422.21	231.77 KB (237,328 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\es.dll
mtxoci.dll	1999.9.3421.3	109.27 KB (111,888 bytes)	2/5/2001 10:44:18 AM	Microsoft Corporation	c:\winnt\system32\mtxoci.dll
resutils.dll	5.00.2191.1	39.77 KB (40,720 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\resutils.dll
clusapi.dll	5.00.2179.1	50.27 KB (51,472 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\clusapi.dll
msvcp50.dll	5.00.7051	552.50 KB (565,760 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\msvcp50.dll
xolehlp.dll	1999.9.3421.3	17.27 KB (17,680 bytes)	2/5/2001 10:44:17 AM	Microsoft Corporation	c:\winnt\system32\xolehlp.dll
msdtclog.dll	1999.9.3421.3	89.77 KB (91,920 bytes)	2/5/2001 10:44:17 AM	Microsoft Corporation	c:\winnt\system32\msdtclog.dll
mtxclu.dll	1999.9.3421.3	50.27 KB (51,472 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\mtxclu.dll
msdtcprx.dll	1999.9.3422.10	619.27 KB (634,128 bytes)	2/5/2001 10:44:18 AM	Microsoft Corporation	c:\winnt\system32\msdtcprx.dll
txfaux.dll	1999.9.3422.24	341.27 KB (349,456 bytes)	2/5/2001 10:44:17 AM	Microsoft Corporation	c:\winnt\system32\txfaux.dll
msdtctm.dll	1999.9.3422.12	1.02 MB (1,070,864 bytes)	2/5/2001 10:44:18 AM	Microsoft Corporation	c:\winnt\system32\msdtctm.dll
msdtc.exe	1999.9.3421.3	6.77 KB (6,928 bytes)	2/5/2001 10:44:17 AM	Microsoft Corporation	c:\winnt\system32\msdtc.exe

wmi.dll	5.00.2191.1	6.27 KB (6,416 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	ntdsa.dll	5.00.2195.1	993.27 KB (1,017,104 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\wmi.dll					c:\winnt\system32\ntdsa.dll				
admwprox.dll	5.00.0984	31.77 KB (32,528 bytes)	2/5/2001 10:44:55 AM	Microsoft Corporation	kdcsvc.dll	5.00.2181.1	133.77 KB (136,976 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\admwprox.dll					c:\winnt\system32\kdcsvc.dll				
inetpp.dll	5.00.2161.1	63.27 KB (64,784 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	sfmapi.dll	5.00.2134.1	38.77 KB (39,696 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\inetpp.dll					c:\winnt\system32\sfmapi.dll				
win32spl.dll	5.00.2162.1	92.27 KB (94,480 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	rassfm.dll	5.00.2168.1	21.27 KB (21,776 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\win32spl.dll					c:\winnt\system32\rassfm.dll				
usbmon.dll	5.00.2165.1	11.27 KB (11,536 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	mpr.dll	5.00.2146.1	53.27 KB (54,544 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\usbmon.dll					c:\winnt\system32\mpr.dll				
tcpmon.dll	5.00.2165.1	40.77 KB (41,744 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	schannel.dll	5.00.2170.1	139.77 KB (143,120 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\tcpmon.dll					c:\winnt\system32\schannel.dll				
pjlmon.dll	5.00.2165.1	12.77 KB (13,072 bytes)	11/30/1999 6:39:36 PM	Microsoft Corporation	netlogon.dll	5.00.2182.1	347.77 KB (356,112 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\pjlmon.dll					c:\winnt\system32\netlogon.dll				
cnbjmon.dll	5.00.2134.1	43.77 KB (44,816 bytes)	11/30/1999 6:38:48 PM	Microsoft Corporation	kerberos.dll	5.00.2181.1	196.77 KB (201,488 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\cnbjmon.dll					c:\winnt\system32\kerberos.dll				
localspl.dll	5.00.2191.1	244.77 KB (250,640 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	msprivs.dll	5.00.2154.1	41.50 KB (42,496 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\localspl.dll					c:\winnt\system32\msprivs.dll				
spoolss.dll	5.00.2161.1	61.77 KB (63,248 bytes)	2/5/2001 9:57:16 AM	Microsoft Corporation	samsrv.dll	5.00.2192.1	357.77 KB (366,352 bytes)	12/7/1999 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)	2/5/2001 9:57:15 AM	Microsoft Corporation	lsasrv.dll	5.00.2184.1	487.77 KB (499,472 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\spoolsv.exe					c:\winnt\system32\lsasrv.dll				
rpss.dll	5.00.2181.1	229.27 KB (234,768 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	lsass.exe	5.00.2184.1	32.77 KB (33,552 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\rpss.dll					c:\winnt\system32\lsass.exe				
svchost.exe	5.00.2134.1	7.77 KB (7,952 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	ntlsapi.dll	5.00.2134.1	6.77 KB (6,928 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\svchost.exe					c:\winnt\system32\ntlsapi.dll				
dssbase.dll	5.00.2150.1	140.77 KB (144,144 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	wmicore.dll	5.00.2178.1	70.77 KB (72,464 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\dssbase.dll					c:\winnt\system32\wmicore.dll				
oakley.dll	5.00.2174.1	420.27 KB (430,352 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	rasadhlp.dll	5.00.2168.1	7.27 KB (7,440 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\oakley.dll					c:\winnt\system32\rasadhlp.dll				
mfc42u.dll	6.00.8665.0	972.05 KB (995,384 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	winnr.dll	5.00.2160.1	18.77 KB (19,216 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\mfc42u.dll					c:\winnt\system32\winnr.dll				
polagent.dll	5.00.2183.1	108.27 KB (110,864 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	mr20.dll	5.00.2152.1	35.77 KB (36,624 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\polagent.dll					c:\winnt\system32\mr20.dll				
scecli.dll	5.00.2191.1	105.27 KB (107,792 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	wshhcpip.dll	5.00.2134.1	17.27 KB (17,680 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\scecli.dll					c:\winnt\system32\wshhcpip.dll				
atl.dll	3.00.8449	57.56 KB (58,938 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	msafd.dll	5.00.2153.1	54.27 KB (55,568 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\atl.dll					c:\winnt\system32\msafd.dll				
certcli.dll	5.00.2175.1	132.27 KB (135,440 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	msocket.dll	5.00.2152.1	62.27 KB (63,760 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\certcli.dll					c:\winnt\system32\msocket.dll				
esent.dll	6.0.3939.6	1.07 MB (1,120,016 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	xactsrv.dll	5.00.2134.1	90.27 KB (92,432 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\esent.dll					c:\winnt\system32\xactsrv.dll				
ntdsatq.dll	5.00.2181.1	31.27 KB (32,016 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	msgsvc.dll	5.00.2181.1	33.77 KB (34,576 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\ntdsatq.dll					c:\winnt\system32\msgsvc.dll				

browser.dll	5.00.2142.1	48.27 KB (49,424 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\browser.dll				
alrsvc.dll	5.00.2134.1	17.77 KB (18,192 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\alrsvc.dll				
trkwks.dll	5.00.2166.1	88.77 KB (90,896 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\trkwks.dll				
seclogon.dll	5.00.2135.1	15.77 KB (16,144 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\seclogon.dll				
psbase.dll	5.00.2146.1	111.77 KB (114,448 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\psbase.dll				
cryptsvc.dll	5.00.2181.1	61.77 KB (63,248 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\cryptsvc.dll				
cryptdll.dll	5.00.2135.1	41.27 KB (42,256 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\cryptdll.dll				
wkssvc.dll	5.00.2181.1	95.27 KB (97,552 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\wkssvc.dll				
srvsvc.dll	5.00.2178.1	79.27 KB (81,168 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\srvsvc.dll				
cfgmgr32.dll	5.00.2134.1	16.77 KB (17,168 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\cfgmgr32.dll				
dmserver.dll	2191.1.296.2	11.77 KB (12,048 bytes)	12/7/1999 7:00:00 AM	VERITAS Software Corp.
c:\winnt\system32\dmserver.dll				
winsta.dll	5.00.2134.1	36.27 KB (37,136 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\winsta.dll				
lmhsvc.dll	5.00.2134.1	9.27 KB (9,488 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\lmhsvc.dll				
dnssrslvr.dll	5.00.2181.1	88.27 KB (90,384 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\dnssrslvr.dll				
tapi32.dll	5.00.2182.1	123.27 KB (126,224 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\tapi32.dll				
rasman.dll	5.00.2188.1	54.77 KB (56,080 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\rasman.dll				
rasapi32.dll	5.00.2188.1	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				
adslrpc.dll	5.00.2172.1	127.77 KB (130,832 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\adslrpc.dll				
activeds.dll	5.00.2172.1	172.77 KB (176,912 bytes)	12/7/1999 7:00:00 AM	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\iphlpapi.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\icmp.dll				
dhcpcsvc.dll	5.00.2153.1	88.77 KB (90,896 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\dhcpcsvc.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	5.00.2160.1	56.27 KB (57,616 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\ntdsapi.dll				
scesrv.dll	5.00.2188.1	225.77 KB (231,184 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\scesrv.dll				
umpnrmgr.dll	5.00.2182.1	86.27 KB (88,336 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\umpnrmgr.dll				
services.exe	5.00.2134.1	86.77 KB (88,848 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\services.exe				
msv1_0.dll	5.00.2164.1	94.77 KB (97,040 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\msv1_0.dll				
clbcatq.dll	1999.9.3422.14	479.27 KB (490,768 bytes)	2/5/2001 10:44:12 AM	Microsoft Corporation
c:\winnt\system32\clbcatq.dll				
oleaut32.dll	2.40.4512	600.27 KB (614,672 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\oleaut32.dll				
cscui.dll	5.00.2172.1	227.27 KB (232,720 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\cscui.dll				
winspool.drv	5.00.2167.1	109.77 KB (112,400 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\winspool.drv				
winscard.dll	5.00.2134.1	77.27 KB (79,120 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\winscard.dll				
wlnotify.dll	5.00.2164.1	53.27 KB (54,544 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\wlnotify.dll				
csd.dll	5.00.2189.1	98.27 KB (100,624 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\csd.dll				
lz32.dll	5.00.2134.1	9.77 KB (10,000 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\lz32.dll				
version.dll	5.00.2134.1	15.77 KB (16,144 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\version.dll				
rsabase.dll	5.00.2150.1	128.77 KB (131,856 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\rsabase.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	5.00.2181.1	966.27 KB (989,456 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation
c:\winnt\system32\ole32.dll				
imagehlp.dll	5.00.2195.1	125.27 KB (128,272 bytes)	12/7/1999 7:00:00 AM	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\msasn1.dll				

```

crypt32.dll 5.131.2173.1      465.77 KB (476,944 bytes)
12/7/1999 7:00:00 AM      Microsoft Corporation
c:\winnt\system32\crypt32.dll
wintrust.dll 5.131.2143.1      162.27 KB (166,160
bytes) 12/7/1999 7:00:00 AM      Microsoft Corporation
c:\winnt\system32\wintrust.dll
setupapi.dll 5.00.2183.1      554.27 KB (567,568
bytes) 12/7/1999 7:00:00 AM      Microsoft Corporation
c:\winnt\system32\setupapi.dll
winmm.dll 5.00.2161.1      184.77 KB (189,200 bytes)
12/7/1999 7:00:00 AM      Microsoft Corporation
c:\winnt\system32\winmm.dll
comctl32.dll 5.81      540.27 KB (553,232 bytes)
12/7/1999 7:00:00 AM      Microsoft Corporation
c:\winnt\system32\comctl32.dll
shlwapi.dll 5.00.2920.0000      282.77 KB (289,552 bytes)
12/7/1999 7:00:00 AM      Microsoft Corporation
c:\winnt\system32\shlwapi.dll
shell32.dll 5.00.2920.0000      2.24 MB (2,352,400 bytes)
12/7/1999 7:00:00 AM      Microsoft Corporation
c:\winnt\system32\shell32.dll
msgina.dll 5.00.2191.1      309.77 KB (317,200 bytes)
12/7/1999 7:00:00 AM      Microsoft Corporation
c:\winnt\system32\msgina.dll
wsock32.dll 5.00.2152.1      21.27 KB (21,776 bytes)
12/7/1999 7:00:00 AM      Microsoft Corporation
c:\winnt\system32\wsock32.dll
dnsapi.dll 5.00.2181.1      129.77 KB (132,880 bytes)
12/7/1999 7:00:00 AM      Microsoft Corporation
c:\winnt\system32\dnsapi.dll
wldap32.dll 5.00.2168.1      155.77 KB (159,504
bytes) 12/7/1999 7:00:00 AM      Microsoft Corporation
c:\winnt\system32\wldap32.dll
ws2help.dll 5.00.2134.1      17.77 KB (18,192 bytes)
12/7/1999 7:00:00 AM      Microsoft Corporation
c:\winnt\system32\ws2help.dll
ws2_32.dll 5.00.2134.1      69.77 KB (71,440 bytes)
12/7/1999 7:00:00 AM      Microsoft Corporation
c:\winnt\system32\ws2_32.dll
samlib.dll 5.00.2160.1      46.27 KB (47,376 bytes)
12/7/1999 7:00:00 AM      Microsoft Corporation
c:\winnt\system32\samlib.dll
netrap.dll 5.00.2134.1      11.27 KB (11,536 bytes)
12/7/1999 7:00:00 AM      Microsoft Corporation
c:\winnt\system32\netrap.dll
netapi32.dll 5.00.2194.1      302.77 KB (310,032
bytes) 12/7/1999 7:00:00 AM      Microsoft Corporation
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.dll 5.00.2154.1      46.77 KB (47,888 bytes)
12/7/1999 7:00:00 AM      Microsoft Corporation
c:\winnt\system32\secur32.dll
sfc.dll 5.00.2164.1      84.27 KB (86,288 bytes)
12/7/1999 7:00:00 AM      Microsoft Corporation
c:\winnt\system32\sfc.dll
nddeapi.dll 5.00.2137.1      15.27 KB (15,632 bytes)
12/7/1999 7:00:00 AM      Microsoft Corporation
c:\winnt\system32\nddeapi.dll
userenv.dll 5.00.2185.1      361.27 KB (369,936 bytes)
12/7/1999 7:00:00 AM      Microsoft Corporation
c:\winnt\system32\userenv.dll
user32.dll 5.00.2180.1      393.27 KB (402,704 bytes)
12/7/1999 7:00:00 AM      Microsoft Corporation
c:\winnt\system32\user32.dll

```

```

gdi32.dll 5.00.2180.1      228.77 KB (234,256 bytes)
12/7/1999 7:00:00 AM      Microsoft Corporation
c:\winnt\system32\gdi32.dll
rpcrt4.dll 5.00.2193.1      434.27 KB (444,688 bytes)
12/7/1999 7:00:00 AM      Microsoft Corporation
c:\winnt\system32\rpcrt4.dll
advapi32.dll 5.00.2191.1      349.27 KB (357,648
bytes) 12/7/1999 7:00:00 AM      Microsoft Corporation
c:\winnt\system32\advapi32.dll
kernel32.dll 5.00.2191.1      715.27 KB (732,432
bytes) 12/7/1999 7:00:00 AM      Microsoft Corporation
c:\winnt\system32\kernel32.dll
msvcrt.dll 6.10.8637.0      288.09 KB (295,000 bytes)
12/7/1999 7:00:00 AM      Microsoft Corporation
c:\winnt\system32\msvcrt.dll
winlogon.exe 5.00.2182.1      173.27 KB (177,424
bytes) 12/7/1999 7:00:00 AM      Microsoft Corporation
c:\winnt\system32\winlogon.exe
sfcfiles.dll 5.00.2195.1      973.27 KB (996,624 bytes)
12/7/1999 7:00:00 AM      Microsoft Corporation
c:\winnt\system32\sfcfiles.dll
ntdll.dll 5.00.2163.1      469.77 KB (481,040 bytes)
12/7/1999 7:00:00 AM      Microsoft Corporation
c:\winnt\system32\ntdll.dll
smss.exe 5.00.2170.1      44.27 KB (45,328 bytes)
12/7/1999 7:00:00 AM      Microsoft Corporation
c:\winnt\system32\smss.exe

```

[Services]

Display Name	Name	State	Start Mode	Start Name
Service Type	Path	Error Control	Tag ID	
Alerter	Alerter	Running	Auto	Share Process
c:\winnt\system32\services.exe		Normal	LocalSystem	0
Application Management		AppMgmt	Stopped	Manual
Share Process	c:\winnt\system32\services.exe	Normal	LocalSystem	0
Computer Browser	Browser	Running	Auto	Share Process
c:\winnt\system32\services.exe		Normal	LocalSystem	0
Indexing Service	cisvc	Stopped	Manual	Share Process
c:\winnt\system32\cisvc.exe		Normal	LocalSystem	0
ClipBook	ClipSrv	Stopped	Manual	Own Process
c:\winnt\system32\clipsrv.exe		Normal	LocalSystem	0
Distributed File System	Dfs	Running	Auto	
Own Process	c:\winnt\system32\dfsrv.exe	Normal	LocalSystem	0
DHCP Client	Dhcp	Running	Auto	Share Process
c:\winnt\system32\services.exe		Normal	LocalSystem	0
Logical Disk Manager Administrative Service		Stopped	Manual	Share Process
c:\winnt\system32\dmadmin.exe	/com	Normal	LocalSystem	0
Logical Disk Manager		dmserver	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
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	Remote Procedure Call (RPC) Locator	RpcLocator			
c:\winnt\system32\inetrv\inetinfo.exe			Normal	LocalSystem	Stopped	Manual	Own Process		
0					c:\winnt\system32\locator.exe	Normal	LocalSystem	0	
Intersite Messaging	IsmServ	Stopped	Disabled	Own Process	Remote Procedure Call (RPC)	RpcSs	Running	Auto	
c:\winnt\system32\ismserv.exe			Normal	LocalSystem	Share Process	c:\winnt\system32\svchost -k rpsvc			
Kerberos Key Distribution Center			kdc	Stopped	Normal	LocalSystem	0		
Disabled	Share Process				QoS Admission Control (RSVP)	RSVP	Running	Auto	
Normal	LocalSystem	0			Own Process	c:\winnt\system32\rsvp.exe -s	Normal		
Server	lanmanserver	Running	Auto	Share Process	LocalSystem	0			
c:\winnt\system32\services.exe			Normal	LocalSystem	Security Accounts Manager	SamSs	Running	Auto	
Workstation	lanmanworkstation	Running	Auto		Share Process	c:\winnt\system32\lsass.exe	Normal		
Share Process	c:\winnt\system32\services.exe		Normal		LocalSystem	0			
LocalSystem	0				Smart Card Helper	SCardDrv	Stopped	Manual	Share Process
License Logging Service	LicenseService	Running			c:\winnt\system32\scardsvr.exe	Ignore	LocalSystem	0	
Auto	Own Process				Smart Card	SCardSvr	Stopped	Manual	Share Process
Normal	LocalSystem	0			c:\winnt\system32\scardsvr.exe	Ignore	LocalSystem	0	
TCP/IP NetBIOS Helper Service	LmHosts	Running	Auto		Task Scheduler	Schedule	Running	Auto	Share Process
Share Process	c:\winnt\system32\services.exe		Normal		c:\winnt\system32\mstask.exe	Normal	LocalSystem	0	
LocalSystem	0				RunAs Service	seclogon	Running	Auto	Share Process
Messenger	Messenger	Running	Auto	Share Process	c:\winnt\system32\services.exe	Ignore	LocalSystem	0	
c:\winnt\system32\services.exe			Normal	LocalSystem	System Event Notification	SENS	Running	Auto	
NetMeeting	Remote Desktop Sharing			Stopped	Share Process	c:\winnt\system32\svchost.exe -k netsvcs			
Manual	Own Process				Normal	LocalSystem	0		
Normal	LocalSystem	0			Internet Connection Sharing	SharedAccess	Stopped		
Distributed Transaction Coordinator			MSDTC	Running	Manual	Share Process	c:\winnt\system32\svchost.exe -k		
Auto	Own Process				netsvcs	Normal	LocalSystem	0	
Normal	LocalSystem	0			Simple TCP/IP Services	SimpTcp	Running	Auto	
Windows Installer	MSIServer	Stopped	Manual	Share Process	Share Process	c:\winnt\system32\tcpshvcs.exe	Normal		
c:\winnt\system32\msiexec.exe /v			Normal	LocalSystem	LocalSystem	0			
0					Print Spooler	Spooler	Running	Auto	Own Process
Network DDE	NetDDE	Stopped	Manual	Share Process	c:\winnt\system32\spoolsv.exe	Normal	LocalSystem	0	
c:\winnt\system32\netdde.exe			Normal	LocalSystem	Performance Logs and Alerts	SysmonLog	Stopped		
Network DDE	DSDMNetDDEdsdm	Stopped	Manual		Manual	Own Process	c:\winnt\system32\smlogsvc.exe		
Share Process	c:\winnt\system32\netdde.exe		Normal		Normal	LocalSystem	0		
LocalSystem	0				Telephony TapiSrv	Running	Manual	Share Process	
Net Logon	Netlogon	Stopped	Manual	Share Process	c:\winnt\system32\svchost.exe -k tapisrv	Normal	LocalSystem	0	
c:\winnt\system32\lsass.exe			Normal	LocalSystem	0				
Network Connections	Netman	Running	Manual	Share Process	Terminal Services	TermService	Stopped	Disabled	
c:\winnt\system32\svchost.exe -k netsvcs			Normal	LocalSystem	Own Process	c:\winnt\system32\termsrv.exe	Normal		
0					LocalSystem	0			
File Replication	NtFrs	Stopped	Manual	Own Process	Telnet	TlntSvr	Stopped	Manual	Own Process
c:\winnt\system32\ntfrs.exe			Ignore	LocalSystem	c:\winnt\system32\tlntsvr.exe	Normal	LocalSystem	0	
NT LM Security Support Provider					Distributed Link Tracking Server	TrkSvr	Stopped	Manual	
Manual	Share Process				Share Process	c:\winnt\system32\services.exe	Normal		
Normal	LocalSystem	0			LocalSystem	0			
Removable Storage	NtmsSvc	Running	Auto	Share Process	Distributed Link Tracking Client	TrkWks	Running	Auto	
c:\winnt\system32\svchost.exe -k netsvcs			Normal	LocalSystem	Share Process	c:\winnt\system32\services.exe	Normal		
0					LocalSystem	0			
Plug and Play	PlugPlay	Running	Auto	Share Process	Uninterruptible Power Supply	UPS	Stopped	Manual	
c:\winnt\system32\services.exe			Normal	LocalSystem	Own Process	c:\winnt\system32\ups.exe	Normal		
0					LocalSystem	0			
IPSEC Policy Agent	PolicyAgent	Running	Auto		Utility Manager	UtilMan	Stopped	Manual	Own Process
Share Process	c:\winnt\system32\lsass.exe		Normal		c:\winnt\system32\utilman.exe	Normal	LocalSystem	0	
LocalSystem	0				Windows Time	W32Time	Stopped	Manual	Share Process
Protected Storage	ProtectedStorage	Running	Auto		c:\winnt\system32\services.exe	Normal	LocalSystem	0	
Share Process	c:\winnt\system32\services.exe		Normal		World Wide Web Publishing Service	W3SVC	Running		
LocalSystem	0				Auto	Share Process			
Remote Access Auto Connection Manager	RasAuto	Stopped			c:\winnt\system32\inetrv\inetinfo.exe	Normal	LocalSystem	0	
Manual	Share Process				0				
c:\winnt\system32\svchost.exe -k					Windows Management Instrumentation	WinMgmt	Running		
netsvcs	Normal	LocalSystem	0		Auto	Own Process			
Remote Access Connection Manager	RasMan	Stopped			c:\winnt\system32\wbem\winmgmt.exe	Ignore	LocalSystem	0	
Manual	Share Process				0				
c:\winnt\system32\svchost.exe -k					Windows Management Instrumentation Driver Extensions				
netsvcs	Normal	LocalSystem	0		Wmi	Running	Manual	Share Process	
Routing and Remote Access	RemoteAccess	Stopped			c:\winnt\system32\services.exe	Normal	LocalSystem	0	
Disabled	Share Process				[Program Groups]				
Share Process	c:\winnt\system32\svchost.exe -k								
LocalSystem	0								
Remote Registry Service	RemoteRegistry	Running							
Auto	Own Process								
Normal	LocalSystem	0							

Group Name	Name	User Name		Item	Value
Accessories	Default User:Accessories	Default User	Default User	Version	5.00.2920.0000
Accessories\Accessibility	Default			Build	52920
User:Accessories\Accessibility	Default User			Product ID	51876-OEM-0000007-00000
Accessories\Entertainment	Default			Application Path	C:\Program Files\Internet Explorer
User:Accessories\Entertainment	Default User			Language	English (United States)
Accessories\System Tools	Default User:Accessories\System Tools	Default User	Default User	Active Printer	Not Available
Startup	Default User:Startup	Default User		Cipher Strength	56-bit
Accessories	All Users:Accessories	All Users		Content Advisor	Disabled
Accessories\Accessibility	All Users:Accessories\Accessibility	All Users		IEAK Install	No
Accessories\Communications	All			[File Versions]	
Users:Accessories\Communications	All Users			File	Version
Accessories\Entertainment	All Users:Accessories\Entertainment	All Users		Size	Date
Accessories\Games	All Users:Accessories\Games	All Users		Path	Company
Accessories\Microsoft Script Debugger	All			advapi32.dll	5.0.2191.1 349 KB 12/7/1999 7:00:00 AM
Users:Accessories\Microsoft Script Debugger	All Users			C:\WINNT\system32	Microsoft Corporation
Accessories\System Tools	All Users:Accessories\System Tools	All Users		advpack.dll	5.0.2920.0 87 KB 12/7/1999 7:00:00 AM
Administrative Tools	All Users:Administrative Tools	All Users		C:\WINNT\system32	Microsoft Corporation
Microsoft SQL Server	All Users:Microsoft SQL Server	All Users		browsec.dll	5.0.2920.0 35 KB 12/7/1999 7:00:00 AM
Startup	All Users:Startup	All Users		C:\WINNT\system32	Microsoft Corporation
Accessories	CLIENT10\Administrator:Accessories			browserui.dll	5.0.2920.0 793 KB 12/7/1999 7:00:00 AM
CLIENT10\Administrator				C:\WINNT\system32	Microsoft Corporation
Accessories\Accessibility	CLIENT10\Administrator:Accessories\Accessibility			ckcnv.exe	5.0.2189.1 9 KB 12/7/1999 7:00:00 AM
CLIENT10\Administrator				C:\WINNT\system32	Microsoft Corporation
Accessories\Entertainment	CLIENT10\Administrator:Accessories\Entertainment			comctl32.dll	5.81.2920.0 540 KB 12/7/1999 7:00:00 AM
CLIENT10\Administrator				C:\WINNT\system32	Microsoft Corporation
Accessories\System Tools	CLIENT10\Administrator:Accessories\System Tools			crypt32.dll	5.131.2173.1 466 KB 12/7/1999 7:00:00 AM
CLIENT10\Administrator				C:\WINNT\system32	Microsoft Corporation
Administrative Tools	CLIENT10\Administrator:Administrative Tools			ehngis.dll	<File Missing> Not Available Not Available
CLIENT10\Administrator				Not Available	Not Available
Startup	CLIENT10\Administrator:Startup			iemigrat.dll	<File Missing> Not Available Not Available
CLIENT10\Administrator				Available	Not Available Not Available
[Startup Programs]				iesetup.dll	5.0.2920.0 57 KB 12/7/1999 7:00:00 AM
Program	Command	User Name	Location	C:\WINNT\system32	Microsoft Corporation
Promon.exe	promon.exe	All Users		ieexplore.exe	5.0.2920.0 59 KB 12/7/1999 7:00:00 AM
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run				C:\Program Files\Internet Explorer	Microsoft Corporation
[OLE Registration]				imagehlp.dll	5.0.2195.1 125 KB 12/7/1999 7:00:00 AM
Object	Local Server			C:\WINNT\system32	Microsoft Corporation
Sound (OLE2)	sndrec32.exe			imghelp.dll	<File Missing> Not Available Not Available
Media Clip	mplay32.exe			Available	Not Available Not Available
Video Clip	mplay32.exe /avi			inseng.dll	5.0.2920.0 72 KB 12/7/1999 7:00:00 AM
MIDI Sequence	mplay32.exe /mid			C:\WINNT\system32	Microsoft Corporation
Sound	Not Available			jobexec.dll	5.0.0.1 47 KB 12/7/1999 7:00:00 AM
Media Clip	Not Available			C:\WINNT\system32	Microsoft Corporation
Image Document	"C:\Program Files\Windows NT\Accessories\ImageVue\KodakImg.exe"			jscript.dll	5.1.0.4615 476 KB 12/7/1999 7:00:00 AM
WordPad Document	"%ProgramFiles%\Windows NT\Accessories\WORDPAD.EXE"			C:\WINNT\system32	Microsoft Corporation
Windows Media Services DRM Storage object				jsproxy.dll	5.0.2920.0 13 KB 12/7/1999 7:00:00 AM
Bitmap Image	mspaint.exe			C:\WINNT\system32	Microsoft Corporation
[Internet Explorer 5]				msahtml.dll	<File Missing> Not Available Not Available
[ Following are sub-categories of this main category ]				Available	Not Available Not Available
[Summary]				mshtml.dll	5.0.2920.0 2302 KB 12/7/1999 7:00:00 AM
				C:\WINNT\system32	Microsoft Corporation
				msjava.dll	5.0.3234.0 918 KB 12/7/1999 7:00:00 AM
				C:\WINNT\system32	Microsoft Corporation
				msoss.dll	<File Missing> Not Available Not Available
				Not Available	Not Available
				msxml.dll	5.0.2920.0 521 KB 12/7/1999 7:00:00 AM
				C:\WINNT\system32	Microsoft Corporation
				occache.dll	5.0.2920.0 86 KB 12/7/1999 7:00:00 AM
				C:\WINNT\system32	Microsoft Corporation
				ole32.dll	5.0.2181.1 966 KB 12/7/1999 7:00:00 AM
				C:\WINNT\system32	Microsoft Corporation
				oleaut32.dll	2.40.4512.1 600 KB 12/7/1999 7:00:00 AM
				C:\WINNT\system32	Microsoft Corporation
				olepro32.dll	5.0.4512.1 160 KB 12/7/1999 7:00:00 AM
				C:\WINNT\system32	Microsoft Corporation



rsabase.dll 5.0.2150.1 129 KB 12/7/1999 7:00:00 AM  
 C:\WINNT\system32 Microsoft Corporation  
 rsaenh.dll <File Missing> Not Available Not Available  
 Not Available Not Available  
 rsapi32.dll <File Missing> Not Available Not Available  
 Not Available Not Available  
 rsasig.dll <File Missing> Not Available Not Available  
 Not Available Not Available  
 schannel.dll 5.0.2170.0 140 KB 12/7/1999 7:00:00 AM  
 C:\WINNT\system32 Microsoft Corporation  
 shdoc401.dll <File Missing> Not Available Not Available  
 Available Not Available Not Available  
 shdocvw.dll 5.0.2920.0 1078 KB 12/7/1999 7:00:00 AM  
 C:\WINNT\system32 Microsoft Corporation  
 shell32.dll 5.0.2920.0 2297 KB 12/7/1999 7:00:00 AM  
 C:\WINNT\system32 Microsoft Corporation  
 shlwapi.dll 5.0.2920.0 283 KB 12/7/1999 7:00:00 AM  
 C:\WINNT\system32 Microsoft Corporation  
 url.dll 5.0.2920.0 82 KB 12/7/1999 7:00:00 AM  
 C:\WINNT\system32 Microsoft Corporation  
 urlmon.dll 5.0.2920.0 427 KB 12/7/1999 7:00:00 AM  
 C:\WINNT\system32 Microsoft Corporation  
 vbscript.dll 5.1.0.4615 428 KB 12/7/1999 7:00:00 AM  
 C:\WINNT\system32 Microsoft Corporation  
 webcheck.dll 5.0.2920.0 252 KB 12/7/1999 7:00:00 AM  
 C:\WINNT\system32 Microsoft Corporation  
 win.com 5.0.2134.1 24 KB 12/7/1999 7:00:00 AM  
 C:\WINNT\system32 Microsoft Corporation  
 wininet.dll 5.0.2920.0 457 KB 12/7/1999 7:00:00 AM  
 C:\WINNT\system32 Microsoft Corporation  
 winsock.dll 3.10.0.103 3 KB 12/7/1999 7:00:00 AM  
 C:\WINNT\system32 Microsoft Corporation  
 wintrust.dll 5.131.2143.1 162 KB 12/7/1999  
 7:00:00 AM C:\WINNT\system32 Microsoft Corporation  
 wsock.vxd <File Missing> Not Available Not Available  
 Not Available Not Available  
 wsock32.dll 5.0.2152.1 21 KB 12/7/1999 7:00:00 AM  
 C:\WINNT\system32 Microsoft Corporation  
 wsock32n.dll <File Missing> Not Available Not Available  
 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	17343 MB

Available Disk Space 14794 MB  
 Maximum Cache Size 541 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	2/5/2001 to 1/12/2101	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: 9968wh\_3clients  
File Path: C:\bf\_430\9968wh\_3clients.PRO  
Version: 3

Number of Engines: 112

Name: RTE01A  
Description: RTE11\_toclient  
Directory: c:\rtelogs\RTE01A.log  
Machine: RTE10  
Parameter Set: PARAM2  
Index: 0  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE01A24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE01B  
Description: RTE12\_toclient  
Directory: c:\rtelogs\RTE01B.log  
Machine: RTE10  
Parameter Set: PARAM2  
Index: 3000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE01B24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE01C  
Description: RTE13\_toclient  
Directory: c:\rtelogs\RTE01C.log  
Machine: RTE10  
Parameter Set: PARAM2  
Index: 6000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE01C24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE01D  
Description: RTE14\_toclient  
Directory: c:\rtelogs\RTE01D.log  
Machine: RTE10  
Parameter Set: PARAM2  
Index: 9000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE01D24637018  
Connect Rate: 300  
Start Rate: 0

Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE01E  
Description: RTE15\_toclient  
Directory: c:\rtelogs\RTE01E.log  
Machine: RTE10  
Parameter Set: PARAM2  
Index: 12000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE01E24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE01F  
Description: RTE16\_toclient  
Directory: c:\rtelogs\RTE01F.log  
Machine: RTE10  
Parameter Set: PARAM2  
Index: 15000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE01F24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE01G  
Description: RTE17\_toclient  
Directory: c:\rtelogs\RTE01G.log  
Machine: RTE10  
Parameter Set: PARAM2  
Index: 18000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE01G24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE01H  
Description: RTE111\_toclient  
Directory: c:\rtelogs\RTE01H.log  
Machine: RTE10  
Parameter Set: PARAM2  
Index: 21000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE01H24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE01I  
Description: RTE112\_toclient  
Directory: c:\rtelogs\RTE01I.log  
Machine: RTE10  
Parameter Set: PARAM2  
Index: 24000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE01I24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE01J  
Description: RTE113\_toclient  
Directory: c:\rtelogs\RTE01J.log  
Machine: RTE10  
Parameter Set: PARAM2  
Index: 27000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE01J24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE01K  
Description: RTE114\_toclient  
Directory: c:\rtelogs\RTE01K.log  
Machine: RTE10  
Parameter Set: PARAM2  
Index: 30000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE01K24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE01L  
Description: RTE115\_toclient  
Directory: c:\rtelogs\RTE01L.log  
Machine: RTE10  
Parameter Set: PARAM2  
Index: 33000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE01L24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE01M  
Description: RTE116\_toclient  
Directory: c:\rtelogs\RTE01M.log

Machine: RTE10  
Parameter Set: PARAM2  
Index: 36000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE01M24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE01N  
Description: RTE117\_toclient  
Directory: c:\rtelogs\RTE01N.log  
Machine: RTE10  
Parameter Set: PARAM2  
Index: 39000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE01N24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE02A  
Description: RTE21\_toclient  
Directory: c:\rtelogs\RTE02A.log  
Machine: RTE20  
Parameter Set: PARAM2  
Index: 42000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE02A24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE02B  
Description: RTE22\_toclient  
Directory: c:\rtelogs\RTE02B.log  
Machine: RTE20  
Parameter Set: PARAM2  
Index: 45000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE02B24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE02C  
Description: RTE23\_toclient  
Directory: c:\rtelogs\RTE02C.log  
Machine: RTE20  
Parameter Set: PARAM2  
Index: 48000000  
Seed: 46329

Configured Users: 890  
Pipe Name: RTE02C24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE02D  
Description: RTE24\_toclient  
Directory: c:\rtelogs\RTE02D.log  
Machine: RTE20  
Parameter Set: PARAM2  
Index: 51000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE02D24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE02E  
Description: RTE25\_toclient  
Directory: c:\rtelogs\RTE02E.log  
Machine: RTE20  
Parameter Set: PARAM2  
Index: 54000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE02E24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE02F  
Description: RTE26\_toclient  
Directory: c:\rtelogs\RTE02F.log  
Machine: RTE20  
Parameter Set: PARAM2  
Index: 57000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE02F24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE02G  
Description: RTE27\_toclient  
Directory: c:\rtelogs\RTE02G.log  
Machine: RTE20  
Parameter Set: PARAM2  
Index: 60000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE02G24637018  
Connect Rate: 300  
Start Rate: 0

Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE02H  
Description: RTE211\_toclient  
Directory: c:\rtelogs\RTE02H.log  
Machine: RTE20  
Parameter Set: PARAM2  
Index: 63000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE02H24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE02I  
Description: RTE212\_toclient  
Directory: c:\rtelogs\RTE02I.log  
Machine: RTE20  
Parameter Set: PARAM2  
Index: 66000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE02I24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE02J  
Description: RTE213\_toclient  
Directory: c:\rtelogs\RTE02J.log  
Machine: RTE20  
Parameter Set: PARAM2  
Index: 69000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE02J24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE02K  
Description: RTE214\_toclient  
Directory: c:\rtelogs\RTE02K.log  
Machine: RTE20  
Parameter Set: PARAM2  
Index: 72000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE02K24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE02L  
Description: RTE215\_toclient  
Directory: c:\rtelogs\RTE02L.log  
Machine: RTE20  
Parameter Set: PARAM2  
Index: 75000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE02L24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE02M  
Description: RTE216\_toclient  
Directory: c:\rtelogs\RTE02M.log  
Machine: RTE20  
Parameter Set: PARAM2  
Index: 78000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE02M24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE02N  
Description: RTE217\_toclient  
Directory: c:\rtelogs\RTE02N.log  
Machine: RTE20  
Parameter Set: PARAM2  
Index: 81000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE02N24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE03A  
Description: RTE31\_toclient  
Directory: c:\rtelogs\RTE03A.log  
Machine: RTE30  
Parameter Set: PARAM2  
Index: 84000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE03A24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE03B  
Description: RTE32\_toclient  
Directory: c:\rtelogs\RTE03B.log

Machine: RTE30  
Parameter Set: PARAM2  
Index: 87000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE03B24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE03C  
Description: RTE33\_toclient  
Directory: c:\rtelogs\RTE03C.log  
Machine: RTE30  
Parameter Set: PARAM2  
Index: 90000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE03C24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE03D  
Description: RTE34\_toclient  
Directory: c:\rtelogs\RTE03D.log  
Machine: RTE30  
Parameter Set: PARAM2  
Index: 93000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE03D24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE03E  
Description: RTE35\_toclient  
Directory: c:\rtelogs\RTE03E.log  
Machine: RTE30  
Parameter Set: PARAM2  
Index: 96000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE03E24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE03F  
Description: RTE36\_toclient  
Directory: c:\rtelogs\RTE03F.log  
Machine: RTE30  
Parameter Set: PARAM2  
Index: 99000000  
Seed: 46329

Configured Users: 890  
Pipe Name: RTE03F24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE03G  
Description: RTE37\_toclient  
Directory: c:\rtelogs\RTE03G.log  
Machine: RTE30  
Parameter Set: PARAM2  
Index: 102000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE03G24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE03H  
Description: RTE311\_toclient  
Directory: c:\rtelogs\RTE03H.log  
Machine: RTE30  
Parameter Set: PARAM2  
Index: 105000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE03H24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE03I  
Description: RTE312\_toclient  
Directory: c:\rtelogs\RTE03I.log  
Machine: RTE30  
Parameter Set: PARAM2  
Index: 108000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE03I24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE03J  
Description: RTE313\_toclient  
Directory: c:\rtelogs\RTE03J.log  
Machine: RTE30  
Parameter Set: PARAM2  
Index: 111000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE03J24637018  
Connect Rate: 300  
Start Rate: 0

Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE03K  
Description: RTE314\_toclient  
Directory: c:\rtelogs\RTE03K.log  
Machine: RTE30  
Parameter Set: PARAM2  
Index: 114000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE03K24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE03L  
Description: RTE315\_toclient  
Directory: c:\rtelogs\RTE03L.log  
Machine: RTE30  
Parameter Set: PARAM2  
Index: 117000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE03L24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE03M  
Description: RTE316\_toclient  
Directory: c:\rtelogs\RTE03M.log  
Machine: RTE30  
Parameter Set: PARAM2  
Index: 120000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE03M24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE03N  
Description: RTE317\_toclient  
Directory: c:\rtelogs\RTE03N.log  
Machine: RTE30  
Parameter Set: PARAM2  
Index: 123000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE03N24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE04A  
Description: RTE41\_toclient  
Directory: c:\rtelogs\RTE04A.log  
Machine: RTE40  
Parameter Set: PARAM2  
Index: 126000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE04A24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE04B  
Description: RTE42\_toclient  
Directory: c:\rtelogs\RTE04B.log  
Machine: RTE40  
Parameter Set: PARAM2  
Index: 129000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE04B24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE04C  
Description: RTE43\_toclient  
Directory: c:\rtelogs\RTE04C.log  
Machine: RTE40  
Parameter Set: PARAM2  
Index: 132000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE04C24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE04D  
Description: RTE44\_toclient  
Directory: c:\rtelogs\RTE04D.log  
Machine: RTE40  
Parameter Set: PARAM2  
Index: 135000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE04D24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE04E  
Description: RTE45\_toclient  
Directory: c:\rtelogs\RTE04E.log

Machine: RTE40  
Parameter Set: PARAM2  
Index: 138000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE04E24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE04F  
Description: RTE46\_toclient  
Directory: c:\rtelogs\RTE04F.log  
Machine: RTE40  
Parameter Set: PARAM2  
Index: 141000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE04F24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE04G  
Description: RTE47\_toclient  
Directory: c:\rtelogs\RTE04G.log  
Machine: RTE40  
Parameter Set: PARAM2  
Index: 144000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE04G24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE04H  
Description: RTE411\_toclient  
Directory: c:\rtelogs\RTE04H.log  
Machine: RTE40  
Parameter Set: PARAM2  
Index: 147000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE04H24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE04I  
Description: RTE412\_toclient  
Directory: c:\rtelogs\RTE04I.log  
Machine: RTE40  
Parameter Set: PARAM2  
Index: 150000000  
Seed: 46329

Configured Users: 890  
Pipe Name: RTE04I24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE04J  
Description: RTE413\_toclient  
Directory: c:\rtelogs\RTE04J.log  
Machine: RTE40  
Parameter Set: PARAM2  
Index: 153000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE04J24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE04K  
Description: RTE414\_toclient  
Directory: c:\rtelogs\RTE04K.log  
Machine: RTE40  
Parameter Set: PARAM2  
Index: 156000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE04K24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE04L  
Description: RTE415\_toclient  
Directory: c:\rtelogs\RTE04L.log  
Machine: RTE40  
Parameter Set: PARAM2  
Index: 159000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE04L24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE04M  
Description: RTE416\_toclient  
Directory: c:\rtelogs\RTE04M.log  
Machine: RTE40  
Parameter Set: PARAM2  
Index: 162000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE04M24637018  
Connect Rate: 300  
Start Rate: 0

Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE04N  
Description: RTE417\_toclient  
Directory: c:\rtelogs\RTE04N.log  
Machine: RTE40  
Parameter Set: PARAM2  
Index: 165000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE04N24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE05A  
Description: RTE51\_toclient  
Directory: c:\rtelogs\RTE05A.log  
Machine: RTE50  
Parameter Set: PARAM2  
Index: 168000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE05A24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE05B  
Description: RTE52\_toclient  
Directory: c:\rtelogs\RTE05B.log  
Machine: RTE50  
Parameter Set: PARAM2  
Index: 171000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE05B24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE05C  
Description: RTE53\_toclient  
Directory: c:\rtelogs\RTE05C.log  
Machine: RTE50  
Parameter Set: PARAM2  
Index: 174000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE05C24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0



Name: RTE05D  
Description: RTE54\_toclient  
Directory: c:\rtelogs\RTE05D.log  
Machine: RTE50  
Parameter Set: PARAM2  
Index: 177000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE05D24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE05E  
Description: RTE55\_toclient  
Directory: c:\rtelogs\RTE05E.log  
Machine: RTE50  
Parameter Set: PARAM2  
Index: 180000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE05E24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE05F  
Description: RTE56\_toclient  
Directory: c:\rtelogs\RTE05F.log  
Machine: RTE50  
Parameter Set: PARAM2  
Index: 183000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE05F24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE05G  
Description: RTE57\_toclient  
Directory: c:\rtelogs\RTE05G.log  
Machine: RTE50  
Parameter Set: PARAM2  
Index: 186000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE05G24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE05H  
Description: RTE511\_toclient  
Directory: c:\rtelogs\RTE05H.log

Machine: RTE50  
Parameter Set: PARAM2  
Index: 189000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE05H24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE05I  
Description: RTE512\_toclient  
Directory: c:\rtelogs\RTE05I.log  
Machine: RTE50  
Parameter Set: PARAM2  
Index: 192000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE05I24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE05J  
Description: RTE513\_toclient  
Directory: c:\rtelogs\RTE05J.log  
Machine: RTE50  
Parameter Set: PARAM2  
Index: 195000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE05J24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE05K  
Description: RTE514\_toclient  
Directory: c:\rtelogs\RTE05K.log  
Machine: RTE50  
Parameter Set: PARAM2  
Index: 198000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE05K24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE05L  
Description: RTE515\_toclient  
Directory: c:\rtelogs\RTE05L.log  
Machine: RTE50  
Parameter Set: PARAM2  
Index: 201000000  
Seed: 46329

Configured Users: 890  
Pipe Name: RTE05L24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE05M  
Description: RTE516\_toclient  
Directory: c:\rtelogs\RTE05M.log  
Machine: RTE50  
Parameter Set: PARAM2  
Index: 204000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE05M24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE05N  
Description: RTE517\_toclient  
Directory: c:\rtelogs\RTE05N.log  
Machine: RTE50  
Parameter Set: PARAM2  
Index: 207000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE05N24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE06A  
Description: RTE61\_toclient  
Directory: c:\rtelogs\RTE06A.log  
Machine: RTE60  
Parameter Set: PARAM2  
Index: 210000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE06A24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE06B  
Description: RTE62\_toclient  
Directory: c:\rtelogs\RTE06B.log  
Machine: RTE60  
Parameter Set: PARAM2  
Index: 213000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE06B24637018  
Connect Rate: 300  
Start Rate: 0

Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE06C  
Description: RTE63\_toclient  
Directory: c:\rtelogs\RTE06C.log  
Machine: RTE60  
Parameter Set: PARAM2  
Index: 216000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE06C24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE06D  
Description: RTE64\_toclient  
Directory: c:\rtelogs\RTE06D.log  
Machine: RTE60  
Parameter Set: PARAM2  
Index: 219000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE06D24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE06E  
Description: RTE65\_toclient  
Directory: c:\rtelogs\RTE06E.log  
Machine: RTE60  
Parameter Set: PARAM2  
Index: 222000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE06E24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE06F  
Description: RTE66\_toclient  
Directory: c:\rtelogs\RTE06F.log  
Machine: RTE60  
Parameter Set: PARAM2  
Index: 225000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE06F24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE06G  
Description: RTE67\_toclient  
Directory: c:\rtelogs\RTE06G.log  
Machine: RTE60  
Parameter Set: PARAM2  
Index: 228000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE06G24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE06H  
Description: RTE611\_toclient  
Directory: c:\rtelogs\RTE06H.log  
Machine: RTE60  
Parameter Set: PARAM2  
Index: 231000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE06H24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE06I  
Description: RTE612\_toclient  
Directory: c:\rtelogs\RTE06I.log  
Machine: RTE60  
Parameter Set: PARAM2  
Index: 234000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE06I24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE06J  
Description: RTE613\_toclient  
Directory: c:\rtelogs\RTE06J.log  
Machine: RTE60  
Parameter Set: PARAM2  
Index: 237000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE06J24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE06K  
Description: RTE614\_toclient  
Directory: c:\rtelogs\RTE06K.log

Machine: RTE60  
Parameter Set: PARAM2  
Index: 240000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE06K24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE06L  
Description: RTE615\_toclient  
Directory: c:\rtelogs\RTE06L.log  
Machine: RTE60  
Parameter Set: PARAM2  
Index: 243000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE06L24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE06M  
Description: RTE616\_toclient  
Directory: c:\rtelogs\RTE06M.log  
Machine: RTE60  
Parameter Set: PARAM2  
Index: 246000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE06M24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE06N  
Description: RTE617\_toclient  
Directory: c:\rtelogs\RTE06N.log  
Machine: RTE60  
Parameter Set: PARAM2  
Index: 249000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE06N24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE07A  
Description: RTE71\_toclient  
Directory: c:\rtelogs\RTE07A.log  
Machine: RTE70  
Parameter Set: PARAM2  
Index: 252000000  
Seed: 46329

Configured Users: 890  
Pipe Name: RTE07A24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE07B  
Description: RTE72\_toclient  
Directory: c:\rtelogs\RTE07B.log  
Machine: RTE70  
Parameter Set: PARAM2  
Index: 255000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE07B24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE07C  
Description: RTE73\_toclient  
Directory: c:\rtelogs\RTE07C.log  
Machine: RTE70  
Parameter Set: PARAM2  
Index: 258000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE07C24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE07D  
Description: RTE74\_toclient  
Directory: c:\rtelogs\RTE07D.log  
Machine: RTE70  
Parameter Set: PARAM2  
Index: 261000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE07D24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE07E  
Description: RTE75\_toclient  
Directory: c:\rtelogs\RTE07E.log  
Machine: RTE70  
Parameter Set: PARAM2  
Index: 264000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE07E24637018  
Connect Rate: 300  
Start Rate: 0

Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE07F  
Description: RTE76\_toclient  
Directory: c:\rtelogs\RTE07F.log  
Machine: RTE70  
Parameter Set: PARAM2  
Index: 267000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE07F24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE07G  
Description: RTE77\_toclient  
Directory: c:\rtelogs\RTE07G.log  
Machine: RTE70  
Parameter Set: PARAM2  
Index: 270000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE07G24637018  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE07H  
Description: RTE711\_toclient  
Directory: c:\rtelogs\RTE07H.log  
Machine: RTE70  
Parameter Set: PARAM2  
Index: 273000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE07H24637023  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE07I  
Description: RTE712\_toclient  
Directory: c:\rtelogs\RTE07I.log  
Machine: RTE70  
Parameter Set: PARAM2  
Index: 276000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE07I24637023  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE07J  
Description: RTE713\_toclient  
Directory: c:\rtelogs\RTE07J.log  
Machine: RTE70  
Parameter Set: PARAM2  
Index: 279000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE07J24637023  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE07K  
Description: RTE714\_toclient  
Directory: c:\rtelogs\RTE07K.log  
Machine: RTE70  
Parameter Set: PARAM2  
Index: 282000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE07K24637023  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE07L  
Description: RTE715\_toclient  
Directory: c:\rtelogs\RTE07L.log  
Machine: RTE70  
Parameter Set: PARAM2  
Index: 285000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE07L24637023  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE07M  
Description: RTE716\_toclient  
Directory: c:\rtelogs\RTE07M.log  
Machine: RTE70  
Parameter Set: PARAM2  
Index: 288000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE07M24637023  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE07N  
Description: RTE717\_toclient  
Directory: c:\rtelogs\RTE07N.log

Machine: RTE70  
Parameter Set: PARAM2  
Index: 291000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE07N24637023  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE08A  
Description: RTE81\_toclient  
Directory: c:\rtelogs\RTE08A.log  
Machine: RTE80  
Parameter Set: PARAM2  
Index: 294000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE08A24637023  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE08B  
Description: RTE82\_toclient  
Directory: c:\rtelogs\RTE08B.log  
Machine: RTE80  
Parameter Set: PARAM2  
Index: 297000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE08B24637023  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE08C  
Description: RTE83\_toclient  
Directory: c:\rtelogs\RTE08C.log  
Machine: RTE80  
Parameter Set: PARAM2  
Index: 300000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE08C24637023  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE08D  
Description: RTE84\_toclient  
Directory: c:\rtelogs\RTE08D.log  
Machine: RTE80  
Parameter Set: PARAM2  
Index: 303000000  
Seed: 46329

Configured Users: 890  
Pipe Name: RTE08D24637023  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE08E  
Description: RTE85\_toclient  
Directory: c:\rtelogs\RTE08E.log  
Machine: RTE80  
Parameter Set: PARAM2  
Index: 306000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE08E24637023  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE08F  
Description: RTE86\_toclient  
Directory: c:\rtelogs\RTE08F.log  
Machine: RTE80  
Parameter Set: PARAM2  
Index: 309000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE08F24637023  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE08G  
Description: RTE87\_toclient  
Directory: c:\rtelogs\RTE08G.log  
Machine: RTE80  
Parameter Set: PARAM2  
Index: 312000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE08G24637023  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE08H  
Description: RTE811\_toclient  
Directory: c:\rtelogs\RTE08H.log  
Machine: RTE80  
Parameter Set: PARAM2  
Index: 315000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE08H24637023  
Connect Rate: 300  
Start Rate: 0

Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE08I  
Description: RTE812\_toclient  
Directory: c:\rtelogs\RTE08I.log  
Machine: RTE80  
Parameter Set: PARAM2  
Index: 318000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE08I24637023  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE08J  
Description: RTE813\_toclient  
Directory: c:\rtelogs\RTE08J.log  
Machine: RTE80  
Parameter Set: PARAM2  
Index: 321000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE08J24637023  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE08K  
Description: RTE814\_toclient  
Directory: c:\rtelogs\RTE08K.log  
Machine: RTE80  
Parameter Set: PARAM2  
Index: 324000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE08K24637023  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE08L  
Description: RTE815\_toclient  
Directory: c:\rtelogs\RTE08L.log  
Machine: RTE80  
Parameter Set: PARAM2  
Index: 327000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE08L24637023  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Name: RTE08M  
Description: RTE816\_toclient  
Directory: c:\rtelogs\RTE08M.log  
Machine: RTE80  
Parameter Set: PARAM2  
Index: 330000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE08M24637023  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 0

Name: RTE08N  
Description: RTE817\_toclient  
Directory: c:\rtelogs\RTE08N.log  
Machine: RTE80  
Parameter Set: PARAM2  
Index: 333000000  
Seed: 46329  
Configured Users: 890  
Pipe Name: RTE08N24637023  
Connect Rate: 300  
Start Rate: 0  
Max. Concurrency: 1000  
Concurrency Rate: 10  
CLIENT\_NURAND: 233  
CPU: 1

Number of User groups: 112

Driver Engine: RTE01A  
IIS Server: CLIENT10\_torte1  
SQL Server: SERV1  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 1 - 89  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE01B  
IIS Server: CLIENT10\_torte1  
SQL Server: SERV1  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 90 - 178  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE01C  
IIS Server: CLIENT10\_torte1  
SQL Server: SERV1  
Database: tpcc  
User: sa

Protocol: HTML  
w\_id Range: 179 - 267  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE01D  
IIS Server: CLIENT10\_torte1  
SQL Server: SERV1  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 268 - 356  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE01E  
IIS Server: CLIENT10\_torte1  
SQL Server: SERV1  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 357 - 445  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE01F  
IIS Server: CLIENT10\_torte3  
SQL Server: SERV1  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 446 - 534  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE01G  
IIS Server: CLIENT10\_torte3  
SQL Server: SERV1  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 535 - 623  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE01H  
IIS Server: CLIENT10\_torte2  
SQL Server: SERV1

Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 624 - 712  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE01H  
IIS Server: CLIENT10\_torte2  
SQL Server: SERV1  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 713 - 801  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE01J  
IIS Server: CLIENT10\_torte2  
SQL Server: SERV1  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 802 - 890  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE01K  
IIS Server: CLIENT10\_torte2  
SQL Server: SERV1  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 891 - 979  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE01L  
IIS Server: CLIENT10\_torte2  
SQL Server: SERV1  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 980 - 1068  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE01M

IIS Server: CLIENT10\_torte3  
SQL Server: SERV1  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 1069 - 1157  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE01N  
IIS Server: CLIENT10\_torte3  
SQL Server: SERV1  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 1158 - 1246  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE02A  
IIS Server: CLIENT20\_torte1  
SQL Server: SERV1  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 1247 - 1335  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE02B  
IIS Server: CLIENT20\_torte1  
SQL Server: SERV1  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 1336 - 1424  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE02C  
IIS Server: CLIENT20\_torte1  
SQL Server: SERV1  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 1425 - 1513  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No



Driver Engine: RTE02D  
IIS Server: CLIENT20\_torte1  
SQL Server: SERV1  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 1514 - 1602  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE02E  
IIS Server: CLIENT20\_torte1  
SQL Server: SERV1  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 1603 - 1691  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE02F  
IIS Server: CLIENT10\_torte4  
SQL Server: SERV1  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 1692 - 1780  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE02G  
IIS Server: CLIENT10\_torte4  
SQL Server: SERV1  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 1781 - 1869  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE02H  
IIS Server: CLIENT20\_torte2  
SQL Server: SERV1  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 1870 - 1958  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890

District id: 1  
Scale Down: No

Driver Engine: RTE02I  
IIS Server: CLIENT20\_torte2  
SQL Server: SERV1  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 1959 - 2047  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE02J  
IIS Server: CLIENT20\_torte2  
SQL Server: SERV1  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 2048 - 2136  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE02K  
IIS Server: CLIENT20\_torte2  
SQL Server: SERV1  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 2137 - 2225  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE02L  
IIS Server: CLIENT20\_torte2  
SQL Server: SERV1  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 2226 - 2314  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE02M  
IIS Server: CLIENT10\_torte4  
SQL Server: SERV1  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 2315 - 2403  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968

Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE02N  
IIS Server: CLIENT10\_torte4  
SQL Server: SERV1  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 2404 - 2492  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE03A  
IIS Server: CLIENT30\_torte1  
SQL Server: SERV2  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 2493 - 2581  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE03B  
IIS Server: CLIENT30\_torte1  
SQL Server: SERV2  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 2582 - 2670  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE03C  
IIS Server: CLIENT30\_torte1  
SQL Server: SERV2  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 2671 - 2759  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE03D  
IIS Server: CLIENT30\_torte1  
SQL Server: SERV2  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 2760 - 2848

w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE03E  
IIS Server: CLIENT30\_torte1  
SQL Server: SERV2  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 2849 - 2937  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE03F  
IIS Server: CLIENT30\_torte3  
SQL Server: SERV2  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 2938 - 3026  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE03G  
IIS Server: CLIENT30\_torte3  
SQL Server: SERV2  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 3027 - 3115  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE03H  
IIS Server: CLIENT30\_torte2  
SQL Server: SERV2  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 3116 - 3204  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE03I  
IIS Server: CLIENT30\_torte2  
SQL Server: SERV2  
Database: tpcc  
User: sa

Protocol: HTML  
w\_id Range: 3205 - 3293  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE03J  
IIS Server: CLIENT30\_torte2  
SQL Server: SERV2  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 3294 - 3382  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE03K  
IIS Server: CLIENT30\_torte2  
SQL Server: SERV2  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 3383 - 3471  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE03L  
IIS Server: CLIENT30\_torte2  
SQL Server: SERV2  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 3472 - 3560  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE03M  
IIS Server: CLIENT30\_torte3  
SQL Server: SERV2  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 3561 - 3649  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE03N  
IIS Server: CLIENT30\_torte3  
SQL Server: SERV2

Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 3650 - 3738  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE04A  
IIS Server: CLIENT40\_torte1  
SQL Server: SERV2  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 3739 - 3827  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE04B  
IIS Server: CLIENT40\_torte1  
SQL Server: SERV2  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 3828 - 3916  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE04C  
IIS Server: CLIENT40\_torte1  
SQL Server: SERV2  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 3917 - 4005  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE04D  
IIS Server: CLIENT40\_torte1  
SQL Server: SERV2  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 4006 - 4094  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE04E

IIS Server: CLIENT40\_torte1  
SQL Server: SERV2  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 4095 - 4183  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE04F  
IIS Server: CLIENT30\_torte4  
SQL Server: SERV2  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 4184 - 4272  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE04G  
IIS Server: CLIENT30\_torte4  
SQL Server: SERV2  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 4273 - 4361  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE04H  
IIS Server: CLIENT40\_torte2  
SQL Server: SERV2  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 4362 - 4450  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE04I  
IIS Server: CLIENT40\_torte2  
SQL Server: SERV2  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 4451 - 4539  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE04J  
IIS Server: CLIENT40\_torte2  
SQL Server: SERV2  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 4540 - 4628  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE04K  
IIS Server: CLIENT40\_torte2  
SQL Server: SERV2  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 4629 - 4717  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE04L  
IIS Server: CLIENT40\_torte2  
SQL Server: SERV2  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 4718 - 4806  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE04M  
IIS Server: CLIENT30\_torte4  
SQL Server: SERV2  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 4807 - 4895  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE04N  
IIS Server: CLIENT30\_torte4  
SQL Server: SERV2  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 4896 - 4984  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890

District id: 1  
Scale Down: No

Driver Engine: RTE05A  
IIS Server: CLIENT50\_torte1  
SQL Server: SERV3  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 4985 - 5073  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE05B  
IIS Server: CLIENT50\_torte1  
SQL Server: SERV3  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 5074 - 5162  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE05C  
IIS Server: CLIENT50\_torte1  
SQL Server: SERV3  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 5163 - 5251  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE05D  
IIS Server: CLIENT50\_torte1  
SQL Server: SERV3  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 5252 - 5340  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE05E  
IIS Server: CLIENT50\_torte1  
SQL Server: SERV3  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 5341 - 5429  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968

Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE05F  
IIS Server: CLIENT50\_torte3  
SQL Server: SERV3  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 5430 - 5518  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE05G  
IIS Server: CLIENT50\_torte3  
SQL Server: SERV3  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 5519 - 5607  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE05H  
IIS Server: CLIENT50\_torte2  
SQL Server: SERV3  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 5608 - 5696  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE05I  
IIS Server: CLIENT50\_torte2  
SQL Server: SERV3  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 5697 - 5785  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE05J  
IIS Server: CLIENT50\_torte2  
SQL Server: SERV3  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 5786 - 5874

w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE05K  
IIS Server: CLIENT50\_torte2  
SQL Server: SERV3  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 5875 - 5963  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE05L  
IIS Server: CLIENT50\_torte2  
SQL Server: SERV3  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 5964 - 6052  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE05M  
IIS Server: CLIENT50\_torte3  
SQL Server: SERV3  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 6053 - 6141  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE05N  
IIS Server: CLIENT50\_torte3  
SQL Server: SERV3  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 6142 - 6230  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE06A  
IIS Server: CLIENT60\_torte1  
SQL Server: SERV3  
Database: tpcc  
User: sa

Protocol: HTML  
w\_id Range: 6231 - 6319  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE06B  
IIS Server: CLIENT60\_torte1  
SQL Server: SERV3  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 6320 - 6408  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE06C  
IIS Server: CLIENT60\_torte1  
SQL Server: SERV3  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 6409 - 6497  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE06D  
IIS Server: CLIENT60\_torte1  
SQL Server: SERV3  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 6498 - 6586  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE06E  
IIS Server: CLIENT60\_torte1  
SQL Server: SERV3  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 6587 - 6675  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE06F  
IIS Server: CLIENT50\_torte4  
SQL Server: SERV3

Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 6676 - 6764  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE06G  
IIS Server: CLIENT50\_torte4  
SQL Server: SERV3  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 6765 - 6853  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE06H  
IIS Server: CLIENT60\_torte2  
SQL Server: SERV3  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 6854 - 6942  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE06I  
IIS Server: CLIENT60\_torte2  
SQL Server: SERV3  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 6943 - 7031  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE06J  
IIS Server: CLIENT60\_torte2  
SQL Server: SERV3  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 7032 - 7120  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE06K

IIS Server: CLIENT60\_torte2  
SQL Server: SERV3  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 7121 - 7209  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE06L  
IIS Server: CLIENT60\_torte2  
SQL Server: SERV3  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 7210 - 7298  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE06M  
IIS Server: CLIENT50\_torte4  
SQL Server: SERV3  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 7299 - 7387  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE06N  
IIS Server: CLIENT50\_torte4  
SQL Server: SERV3  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 7388 - 7476  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE07A  
IIS Server: CLIENT70\_torte1  
SQL Server: SERV4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 7477 - 7565  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE07B  
IIS Server: CLIENT70\_torte1  
SQL Server: SERV4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 7566 - 7654  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE07C  
IIS Server: CLIENT70\_torte1  
SQL Server: SERV4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 7655 - 7743  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE07D  
IIS Server: CLIENT70\_torte1  
SQL Server: SERV4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 7744 - 7832  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE07E  
IIS Server: CLIENT70\_torte1  
SQL Server: SERV4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 7833 - 7921  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE07F  
IIS Server: CLIENT70\_torte3  
SQL Server: SERV4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 7922 - 8010  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890

District id: 1  
Scale Down: No

Driver Engine: RTE07G  
IIS Server: CLIENT70\_torte3  
SQL Server: SERV4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 8011 - 8099  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE07H  
IIS Server: CLIENT70\_torte2  
SQL Server: SERV4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 8100 - 8188  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE07I  
IIS Server: CLIENT70\_torte2  
SQL Server: SERV4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 8189 - 8277  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE07J  
IIS Server: CLIENT70\_torte2  
SQL Server: SERV4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 8278 - 8366  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE07K  
IIS Server: CLIENT70\_torte2  
SQL Server: SERV4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 8367 - 8455  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968



Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE07L  
IIS Server: CLIENT70\_torte2  
SQL Server: SERV4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 8456 - 8544  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE07M  
IIS Server: CLIENT70\_torte3  
SQL Server: SERV4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 8545 - 8633  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE07N  
IIS Server: CLIENT70\_torte3  
SQL Server: SERV4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 8634 - 8722  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE08A  
IIS Server: CLIENT80\_torte1  
SQL Server: SERV4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 8723 - 8811  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE08B  
IIS Server: CLIENT80\_torte1  
SQL Server: SERV4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 8812 - 8900

w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE08C  
IIS Server: CLIENT80\_torte1  
SQL Server: SERV4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 8901 - 8989  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE08D  
IIS Server: CLIENT80\_torte1  
SQL Server: SERV4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 8990 - 9078  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE08E  
IIS Server: CLIENT80\_torte1  
SQL Server: SERV4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 9079 - 9167  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE08F  
IIS Server: CLIENT70\_torte4  
SQL Server: SERV4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 9168 - 9256  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE08G  
IIS Server: CLIENT70\_torte4  
SQL Server: SERV4  
Database: tpcc  
User: sa

Protocol: HTML  
w\_id Range: 9257 - 9345  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE08H  
IIS Server: CLIENT80\_torte2  
SQL Server: SERV4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 9346 - 9434  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE08I  
IIS Server: CLIENT80\_torte2  
SQL Server: SERV4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 9435 - 9523  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE08J  
IIS Server: CLIENT80\_torte2  
SQL Server: SERV4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 9524 - 9612  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE08K  
IIS Server: CLIENT80\_torte2  
SQL Server: SERV4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 9613 - 9701  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE08L  
IIS Server: CLIENT80\_torte2  
SQL Server: SERV4

Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 9702 - 9790  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE08M  
IIS Server: CLIENT70\_torte4  
SQL Server: SERV4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 9791 - 9879  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Driver Engine: RTE08N  
IIS Server: CLIENT70\_torte4  
SQL Server: SERV4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 9880 - 9968  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 9968  
Scale: Normal  
User Count: 890  
District id: 1  
Scale Down: No

Number of Parameter Sets: 2

		~Default				
		Default Parameter Set				
		Txn	Think	Key	RT	RT
Menu		Weight	Time	Time	Delay	Fence
		New Order	10.00		12.05	
18.01	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				
		Stock Level	1.00	5.05	2.01	
0.10	20.00	0.10				
		Order Status	1.00	10.05	2.01	
0.10	5.00	0.10				
		PARAM2				
		Editable Default Parameter Set				
		Txn	Think	Key	RT	RT
Menu		Weight	Time	Time	Delay	Fence
		New Order	44.80		12.05	
18.01	0.10	5.00	0.10			

0.10	5.00	Payment 0.10	43.05	12.05	3.01
0.10	5.00	Delivery 0.10	4.05	5.05	2.01
0.10	20.00	Stock Level 0.10	4.05	5.05	2.01
0.10	5.00	Order Status 0.10	4.05	10.05	2.01

# Appendix D: 180-Day Space

TPC-C 180-Day Space Requirements						
Warehouses	9,968	tpmC	121,319.23	tpmC/W	12.17	
Numbers are shown in Kbytes unless otherwise specified.						
Table	Rows	Data	Index	5% Space	8H Space	Total Space
Warehouse	9,968	1,088	208	64.80		1,360.80
District	99,680	11,232	224	572.80		12,028.80
Item	100,000	38,112	288	1,920.00		40,320.00
New-Order	89,712,000	1,598,464	4,192		797,440.00	2,400,096.00
History	299,040,000	17,853,152	58,400		3,487,986.67	21,399,538.67
Orders	299,040,000	9,764,576	3,578,584		2,598,365.81	15,941,525.81
Customer	299,040,000	217,483,648	13,572,552	11,552,810.00		242,609,010.00
Order-Line	2,990,389,741	199,359,328	471,936		38,913,924.68	238,745,188.68
Stock	996,800,000	318,976,008	675,584	15,982,579.60		335,634,171.60
<b>Totals</b>		<b>765,085,608</b>	<b>18,361,968</b>	<b>27,537,947.20</b>	<b>45,797,717.16</b>	<b>856,783,240</b>
Segment	LogDev Cnt.	Segment Size	Needed	Overhead		Not Needed
misc	24	294,912,000	278,540,059	2,785,401		13,586,540.65
big	24	584,908,800	578,243,182	5,782,432		883,186.58
master, msdb,model	4	53,248	53,248			0.00
tpcc_root	4	32,768	32,768			0.00
tempdb	4	34,816	34,816			0.00
<b>Totals</b>		<b>879,941,632</b>	<b>856,904,072</b>	<b>8,567,832</b>		<b>14,469,727.23</b>
Dynamic Space	226,977,056.00	Sum of Data for Order, Order-Line and History				
Static Space	592,576,299.60	Data + Index + 5% Space + Overhead - Dynamic Space				
Free Space	45,918,549.16	Total Segment Size - Dynamic Space - Static Space - Not needed				
Daily Growth	44,200,131.08	(Dynamic Space/Wc * 62.5)* tpmC				
Daily Spread	(20,381,647.45)	Free Space - 1.5 * Daily Growth (Zero If Negative)				
180-Day Space (KB)	8,548,599,893.64	Static Space + 180 (Daily Growth + Daily Spread)				
180-Day Space (GB)	8,152.58	180-Day Space in GB (Excludes OS,Paging and RDBMS Logs)				
Available (GB)	8,542.80	Total storage configured and available for database, minus logs, in a RAID-0 configuration				
<b>Log File Storage Requirements</b>						
Log Used (KB)	3,276,840,000	Total				
% Log Used	8.8109	% Log File Storage Used During Entire Run				
Total N-O Txn	5,593,990	Total Count of New-Order Transactions Completed during Entire Run				
Log per N-O Txn	5.16	KB of Log Storage Used per New-Order Transaction				
8 Hour Log (GB)	286.63	8-Hours of Log in GB (excluding space for redundancy)				
Log Configured (GB)	312.5	78.125GB per Node (RAID-10)				
Disk Capacity	MB	GB				
9.1GB 10K rpm	8,678	8.47				
18.2GB 10K rpm	17,356	16.95				
Space Usage	GB Needed	Disks Priced		GB Priced	GB Usable	TB Usable
180-Day (RAID-0)	8,152.58	336	9.1GB 15K rpm	2,847.60	2,847.60	
		336	18.2GB 15K rpm	5,695.20	5,695.20	
Total DB		672		8,542.80	8,542.80	
8hr Log (RAID-1)	286.63	48	18.2GB 15K rpm	813.60	406.80	
OS, SQL (RAID-1)	32.00	8	9.1GB 10K rpm	67.80	33.90	
<b>Total Space</b>	<b>8,471.21</b>	<b>728</b>		<b>9,424.20</b>	<b>8,983.50</b>	<b>8.77</b>

---

## 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

March 15, 2001

IBM

Tricia Thomas  
D23U/B060/Office E127  
3039 Cornwallis Road  
Research Triangle Park, NC  
27709

Tricia:

Here is the information you requested regarding pricing for several Microsoft products to be used in conjunction with your TPC-C V3.5 benchmark testing.

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

Part Number	Description	Unit Price	Quantity	Price
810-00846	<b>SQL Server 2000 Enterprise Edition</b> <i>Per processor licensing</i> <i>Discount schedule: Open Program Level C</i>	\$ 16,541	32	\$ 529,312
C11-00821	<b>Windows 2000 Server</b> <i>Server license only - No CALs</i> <i>Discount schedule: Open Program - No Level</i>	\$ 738	1	\$ 738
048-00317	<b>Visual C++ Professional 6.0 Win32</b>	\$ 549	1	\$ 549
	<b>5-year maintenance for above software</b>	\$ 2,095	1	\$ 41,900

All products are currently orderable through Microsoft's normal distribution channels.

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](mailto:jamiere@microsoft.com).

# Quote

**Giganet, Inc.**  
580 Main Street  
Bolton, MA 01740  
(978) 779-7200  
**(978) 7797-203 Fax**

**Quote No.:** 000000-00  
**Date:** March 1, 2001

**Ship to:** IBM Corp.  
**Bill to:** Chris King  
3039 Corwallis Road  
Dept. 23UA Bldg. 060/D133  
Research Triangle Park, NC 27709

F.O.B.	Terms
Bolton, MA	Net 30

Qty	Part Number/ Description	Unit Price	Ext. Price
6	cLAN-1000 Giganet Host Adapter Card	\$795.00	\$4,770.00
6	cLAN-A1011 10 Meter Copper Cable	\$135.00	\$810.00
3	cLAN5000 8 port Cluster Switch	\$6,250.00	\$18,750.00
	5-Year Maintenance Price (7x24x4)	\$15,000.00	\$75,000.00
	3-Year Maintenance Price (7x24x4)	\$15,000.00	\$45,000.00
	<b>Hardware TOTAL</b>		<b>\$24,330.00</b>
	Actual shipping charges will be billed		

Prices are valid for 90 days from the date of this quote.

If you have any questions regarding this quote, please contact the GigaNet Sales Department at (978) 779-7400.

Software House International  
2880 Zanker Blvd. #103  
San Jose, CA 95134  
Matthew Martin  
National Account Executive  
1-800-766-6357

Description	Part Number	Unit Price	Qty	Extended Price
8-Port 10/100Mbps Hub	DEH2924	23	13,799	\$317,377
8-Port 10/100Mbps Hub Nway Fast Ethernet Switch	NX-DSS8	25	29	\$725

Prices are valid for 90 days from March 16, 2001.