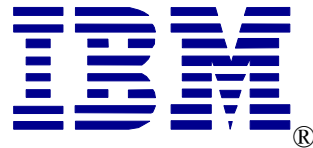


**TPC Benchmark™ C**  
**Full Disclosure Report**  
**for**  
**IBM® @server™ xSeries™ 445**  
**using**  
**Microsoft® SQL Server 2000 Enterprise Edition**  
**and**  
**Microsoft Windows® Server 2003, Datacenter Edition**

**TPC-C Revision 5.1**

**Submitted for Review**  
**July 18, 2003**



## **First Edition - July 2003**

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

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

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

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

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

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

### ***Trademarks***

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

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

### ***Notes***

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

<sup>2</sup> When referring to hard disk capacity, GB, or gigabyte, means one thousand million bytes. Total user-accessible capacity may be less.

---

## Abstract

IBM Corporation conducted the TPC Benchmark™ C on the IBM® @server™ xSeries™ 445 configured as a client/server system. This report documents the full disclosure information required by the TPC Benchmark C Standard Specification, Revision 5.1, including the methodology used to achieve the reported results. All testing fully complied with this revision level.

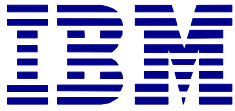
The software used on the xSeries 445 system includes Microsoft® Windows® Server 2003, Datacenter Edition with QFE, and Microsoft SQL Server 2000 Enterprise Edition, SP3 with QFE.

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

The benchmark results are summarized in the following table.

Hardware	Software	Total System Cost	tpmC	\$/tpmC	Total Solution Availability Date
IBM @server xSeries 445	Microsoft SQL Server 2000 Enterprise Edition SP3 with QFE  Microsoft Windows Server 2003 Datacenter Edition with QFE	\$1,599,308	190,510.02	\$8.39	Dec. 31, 2003

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



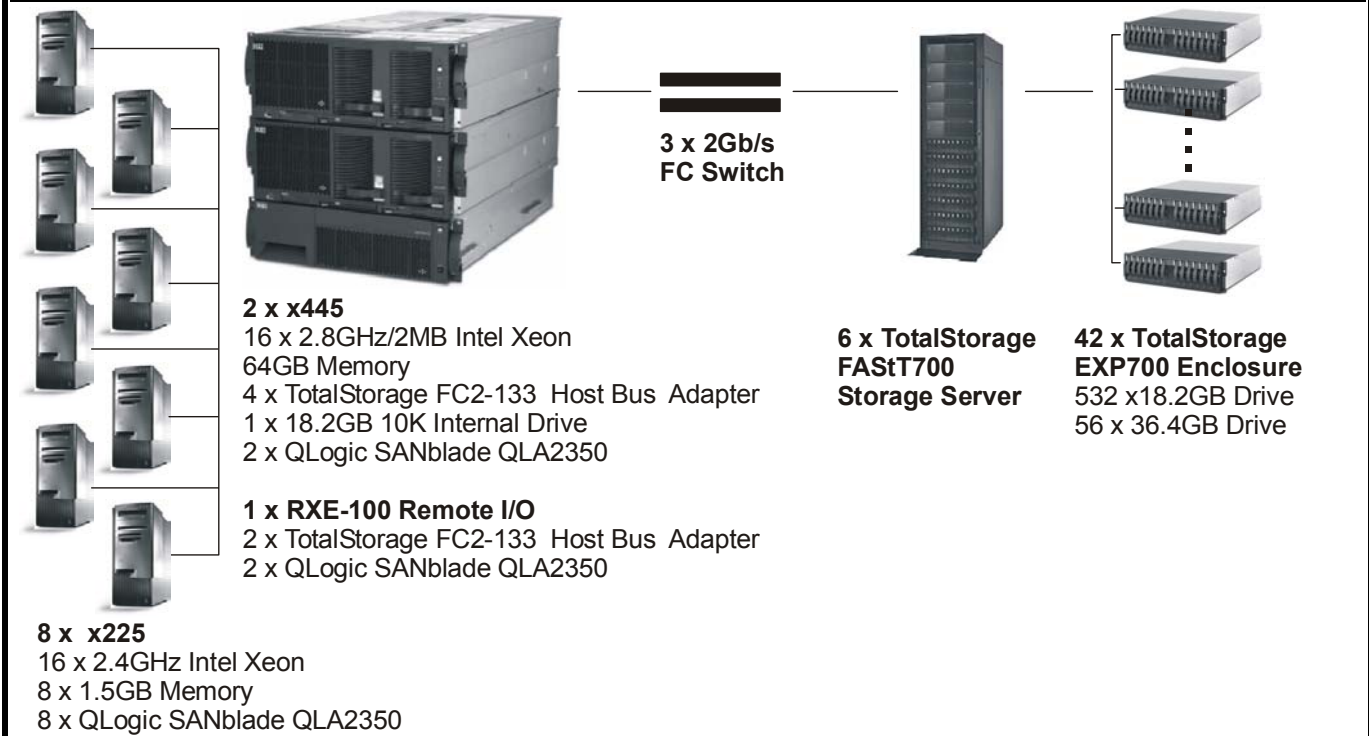
**IBM® @server™ xSeries™ 445 c/s**  
**with**  
**Microsoft® SQL Server 2000**

TPC-C Revision 5.1

Report Date: July 18, 2003

Total System Cost	TPC-C Throughput	Price/Performance	Availability Date
<b>\$1,599,308</b>	<b>190,510.02 tpmC</b>	<b>\$8.39 / tpmC</b>	<b>Dec. 31, 2003</b>

Database Server	Database Manager	Operating System	Other Software	Number of Users
<b>Processors/Cores/Threads: 16/16/32</b> <b>Intel Xeon Processor MP 2.8GHz</b>	<b>Microsoft SQL Server 2000 Enterprise Edition SP3 w/QFE</b>	<b>Microsoft Windows® Server 2003 Datacenter Edition w/QFE</b>	<b>Microsoft Visual C++ 6.0 Win32 Microsoft COM+</b>	<b>152,320</b>



System Component	Qty	Server:	Qty	Each Client:
Processors/Cores/Threads	16/16/32	2.8GHz Xeon Processor MP w/2MB L3 Cache	2/2/4	2.4GHz Xeon DP w/512KB L2 Cache
Cache		1GB ECC SDRAM	1	1.5GB Memory
Memory	64	TotalStorage FC2-133 Host Bus Adapter	1	Ultra320 SCSI Interface
Disk Controllers	4	18.2GB (15000 rpm)	1	18.2GB (10000 rpm)
Disk Drives	532	36.4GB (15000 rpm)		
	56	18.2GB (10000 rpm)		
	1	10919.85GB		
Total Storage		20/40GB SCSI Tape Drive		
Tape Drive	1			

IBM Corporation	IBM @server xSeries 445 c/s with Microsoft SQL Server 2000			TPC-C Revision 5.1			
					Report Date: July 18, 2003		
Description	Part Number	Third Party Brand	Pricing	Unit Price	Quantity	Extended Price	3-Yr. Maint. Price
<b>Server Hardware</b>							
x445 with 2 x 2.8GHz/2MB Xeon Processor	8870-4AX	IBM	1	25,599	2	51,198	6,780
xSeries 2.8GHz/2MB Processor	02R2064	IBM	1	6,599	12	79,188	0
xSeries 445 SMP Expansion Module	02R1870	IBM	1	4,849	2	9,698	0
1GB PC2100 DDR ECC SDRAM RDIMM	33L5039	IBM	1	659	64	42,176	0
IBM TotalStorage FASTT FC2-133 Host Bus Adapter	24P0960	IBM	1	1,485	4	5,940	0
18.2GB 10K Ultra160 SCSI Drive	06P5754	IBM	1	275	1	275	0
QLogic SANBlade QLA2350 FC-VI Adapter (2 spares)	QLA2350-BK	QLogic	3	2,095	6	12,570	0
3.5M Interconnect Management Cable Kit	31P6087	IBM	1	49	5	245	0
E54 15" (13.8" Viewable) Color Monitor	633147N	IBM	1	129	1	129	90
IBM Preferred Pro Full-Size Keyboard PS/2	31P7415	IBM	1	29	1	29	0
IBM Sleek 2-Button Mouse	28L3673	IBM	1	19	1	19	0
<b>Subtotal</b>						<b>201,467</b>	<b>6,870</b>
<b>Server Storage</b>							
IBM RXE-100 Remote Expansion Enclosure	8684-1RX	IBM	1	4,569	1	4,569	1,330
IBM Remote I/O PCI-X 6-Slot Expansion Kit	31P5998	IBM	1	1,699	1	1,699	0
IBM TotalStorage FASTT FC2-133 Host Bus Adapter	24P0960	IBM	1	1,485	2	2,970	0
IBM TotalStorage FASTT700 Storage Server	17421RU	IBM	1	46,499	6	278,994	4,560
IBM FASTT700 Mini Hub	19K1269	IBM	1	899	10	8,990	0
IBM Short Wave SFP Module	19K1271	IBM	1	499	180	89,820	0
IBM 1m LC-LC Fibre Channel Cable	19K1247	IBM	1	79	62	4,898	0
IBM 5m LC-LC Fibre Channel Cable	19K1248	IBM	1	129	40	5,160	0
IBM TotalStorage FASTT EXP700 Storage Exp. Unit	17401RU	IBM	1	6,000	42	252,000	31,920
2Gbps FC 36.4GB 15K Hot-Swap HDD	06P5772	IBM	1	1,115	56	62,440	0
2Gbps FC 18.2GB 15K Hot-Swap HDD	06P5771	IBM	1	681	532	362,292	0
20/40GB DDS/4 Tape AutoLoader	00N7991	IBM	1	699	1	699	0
IBM NetMEDIA Storage Expansion Unit EL	3551001	IBM	1	689	1	689	0
NetBAY42SX Standard Rack	9306421	IBM	1	1,439	5	7,195	840
IBM UPS 500	33L3477	IBM	1	99	1	99	0
Enterprise Rack Prep Fee	06P7514	IBM	1	500	5	2,500	0
Enterprise Rack Installation Fee	06P7515	IBM	1	150	5	750	0
Option Integration Fee	58P8665	IBM	1	110	5	550	0
Image Load Fee	06P7505	IBM	1	150	1	150	0
<b>Subtotal</b>						<b>1,086,464</b>	<b>38,650</b>
<b>Server Software</b>							
Microsoft SQL Server 2000 Enterprise Edition	810-00846	Microsoft	2	16,541	16	264,656	0
Database Software Support Package		Microsoft	2	1,950	3		5,850
IBM Preload Kit for Datacenter 2003 (1-16 Processors)	4816-4DU	IBM	1	54,555	1	54,555	41,850
IBM Maintenance Update Subscription (1-16 Processors - 1 Year)	4816-CDX	IBM	1	1,299	3	3,897	0
<b>Subtotal</b>						<b>323,108</b>	<b>47,700</b>
<b>Client Hardware</b>							
x225 with 2.4GHz/512KB Xeon DP, 512MB Memory	8647-3AX	IBM	1	1,269	8	10,152	5,512
2.4GHz/512KB Xeon DP Processor Upgrade	59P5108	IBM	1	699	8	5,592	0
512MB PC2100 ECC SDRAM RDIMM	33L5038	IBM	1	305	16	4,880	0
18.2GB 10K Ultra160 SCSI Drive	06P5754	IBM	1	275	8	2,200	0
QLogic SANblade QLA 2350 FC-VI Adapter	QLA2350-BK	QLogic	3	2,095	8	16,760	0
IBM 10/100 Dual-Port Server Adapter	22P4901	IBM	1	209	16	3,344	0
Adaptec Quartet64 4-Port Adapter	ANA62044	Adaptec	1	550	8	4,400	0
E54 15" (13.8" Viewable) Color Monitor	633147N	IBM	1	129	8	1,032	720
<b>Subtotal</b>						<b>48,360</b>	<b>6,232</b>
<b>Client Software</b>							
Microsoft Windows 2000 Server with COM+	C11-00821	Microsoft	2	738	8	5,904	0
Microsoft Visual C++ Professional 6.0	254-00170	Microsoft	2	109	1	109	0
<b>Subtotal</b>						<b>6,013</b>	<b>0</b>
<b>Network Components</b>							
IBM TotalStorage SAN Fibre Channel Switch Model F08	3534F08	IBM	1	7,000	4	28,000	4,608
IBM Short Wave SFP Module	19K1271	IBM	2	499	14	6,986	0
IBM 5M LC-LC Fibre Channel Cable	19K1248	IBM	2	129	12	1,548	0
<b>Subtotal</b>						<b>36,534</b>	<b>4,608</b>
<b>Discount</b>						<b>206,698</b>	
<b>Total</b>						<b>1,495,248</b>	<b>104,060</b>
Large purchase discount 14% on xSeries; 15% on TotalStorage. Pricing: 1 - IBM - 1-919-254-0367; 2 - Microsoft; 3 - QLogic				<b>Three-Year Cost of Ownership:</b> \$1,599,308			
Audited by Brad Askins, InfoSizing, Inc.				<b>tpmC:</b> 190,510.02			
				<b>\$/tpmC:</b> \$8.39			
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 specifications. If you find that stated prices are not available according to these terms, please inform the TPC at pricing@tpc.org.							

<b>Numerical Quantities Summary</b>			
<b>MQTh, Computed Maximum Qualified Throughput:</b> 190,510.02 tpmC			
<b>Response Times (in seconds)</b>	<b>90th Percentile</b>	<b>Average</b>	<b>Maximum</b>
<b>New-Order</b>	0.47	0.37	16.25
<b>Payment</b>	0.41	0.31	17.44
<b>Delivery (Interactive)</b>	0.11	0.16	4.58
<b>Stock Level</b>	1.08	0.70	12.76
<b>Order Status</b>	0.45	0.35	16.77
<b>Delivery (Deferred)</b>	0.19	0.14	4.64
<b>Menu</b>	0.11	0.17	4.61
<b>Response time delay added for emulated components:</b> 0.1 Seconds			
<b>Transaction Mix (in percent of total transactions)</b>			<b>Percent</b>
<b>New-Order</b>			44.94
<b>Payment</b>			43.03
<b>Delivery</b>			4.01
<b>Stock Level</b>			4.01
<b>Order Status</b>			4.01
<b>Keying/Think Times (in seconds)</b>	<b>Minimum</b>	<b>Average</b>	<b>Maximum</b>
<b>New-Order</b>	18.00 / 0.00	18.02 / 12.05	18.05 / 120.51
<b>Payment</b>	3.00 / 0.00	3.02 / 12.05	3.05 / 120.90
<b>Delivery</b>	2.00 / 0.00	2.02 / 5.06	2.04 / 50.51
<b>Stock Level</b>	2.00 / 0.00	2.02 / 5.05	2.04 / 50.51
<b>Order Status</b>	2.00 / 0.00	2.02 / 10.04	2.04 / 100.51
<b>Test Duration</b>			
<b>Ramp-up time</b>			53 minutes 45 seconds
<b>Measurement interval</b>			120 minutes
<b>Number of checkpoints</b>			4
<b>Checkpoint interval</b>			30 minutes
<b>Number of transactions (all types) completed in measurement interval</b>			52,906,744

## Table of Contents

<b>Abstract</b>	3
<b>Numerical Quantities Summary</b>	3
<b>Preface</b>	11
<b>General Items</b>	12
Application Code Disclosure and Definition Statements	12
Application Code Disclosure and Definition Statements	12
Benchmark Sponsor	12
Parameter Settings	12
Configuration Diagrams	12
<b>Clause 1: Logical Database Design Related Items</b>	15
Table Definitions	15
Physical Organization of the Database	15
Insert and Delete Operations	15
Horizontal or Vertical Partitioning	15
Replication	15
Table Attributes	15
<b>Clause 2: Transaction and Terminal Profiles Related Items</b>	16
Random Number Generation	16
Screen Layout	16
Terminal Verification	16
Intelligent Terminals	16
Transaction Profiles	16
Deferred Delivery Mechanism	17
<b>Clause 3: Transaction and System Properties Related Items</b>	18
Atomicity Requirements	18
Consistency Requirements	18
Isolation Requirements	19
Durability Requirements	19
<b>Clause 4: Scaling and Database Population Related Items</b>	21
Cardinality of Tables	21
Distribution of Tables and Logs	21
Database Model Implemented	22
Partitions/Replications Mapping	23
60-Day Space Requirement	23
<b>Clause 5: Performance Metrics and Response Time Related Items</b>	24
Measured tpmC	24
Response Times	24
Keying/Think Times	24
Response Time Frequency Distribution Curves	25
Performance Curve for Response Time vs. Throughput	27
New Order Think Time Distribution	28
Throughput vs. Elapsed Time	28
Steady State Methodology	29
Work Performed during Steady State	29
Checkpoints	29
Measurement Interval	29
Transaction Mix	29
Percentage of Total Mix	30
Number of Checkpoints	30
<b>Clause 6: SUT, Driver and Communication Definition Related Items</b>	32
Description of RTE	32
Emulated Components	32
Benchmarked and Targeted System Configuration Diagrams	32

Network Configuration .....	32
Network Bandwidth .....	32
Operator Intervention .....	32
<b>Clause 7: Pricing Related Items</b> .....	33
Hardware and Software Components .....	33
Availability Date .....	33
Measured tpmC .....	33
Country-Specific Pricing .....	33
Usage Pricing .....	33
System Pricing .....	34
<b>Clause 9: Audit Related Items</b> .....	35
Auditor .....	35
Availability of the Full Disclosure Report .....	35
<i>Attestation letter</i> .....	36
<b>Appendix A: Source Code</b> .....	38
Web Client Source Code .....	38
<i>db_odbc_dll.dsp</i> .....	38
<i>dlldata.c</i> .....	39
<i>error.h</i> .....	39
<i>install.c</i> .....	42
<i>install.dsp</i> .....	49
<i>install.h</i> .....	51
<i>install.rc</i> .....	51
<i>install_com.cpp</i> .....	54
<i>install_resource.h</i> .....	56
<i>isapi_dll.dsp</i> .....	57
<i>isapi_dll_resource.h</i> .....	58
<i>license.txt</i> .....	59
<i>methods.h</i> .....	60
<i>null-txns.sql</i> .....	62
<i>readregistry.cpp</i> .....	65
<i>readregistry.h</i> .....	66
<i>restore.vbs</i> .....	66
<i>runsqlcfg.vbs</i> .....	67
<i>rtetime.h</i> .....	69
<i>setup.vbs</i> .....	69
<i>spinlock.h</i> .....	76
<i>tm_com_dll.dsp</i> .....	77
<i>tpcc.cpp</i> .....	100
<i>tpcc.def</i> .....	123
<i>tpcc.h</i> .....	123
<i>tpcc.rc</i> .....	124
<i>tpcc_com.cpp</i> .....	125
<i>tpcc_com.h</i> .....	126
<i>tpcc_com_all.cpp</i> .....	128
<i>tpcc_com_all.def</i> .....	132
<i>tpcc_com_all.dsp</i> .....	132
<i>tpcc_com_all.h</i> .....	133
<i>tpcc_com_all.idl</i> .....	135
<i>tpcc_com_all.rc</i> .....	135
<i>tpcc_com_all.rgs</i> .....	136
<i>tpcc_com_all_i.c</i> .....	137
<i>tpcc_com_all_resource.h</i> .....	138
<i>tpcc_com_no.rgs</i> .....	138
<i>tpcc_com_os.rgs</i> .....	139



<i>tpcc_com_pay.rgs</i>	139
<i>tpcc_com_ps.def</i>	139
<i>tpcc_com_ps.dsp</i>	139
<i>tpcc_com_ps.h</i>	141
<i>tpcc_com_ps.idl</i>	143
<i>tpcc_com_ps_i.c</i>	143
<i>tpcc_com_ps_p.c</i>	145
<i>tpcc_com_sl.rgs</i>	165
<i>tpcc_odbc.cpp</i>	165
<i>tpcc_odbc.h</i>	173
<i>trans.h</i>	174
<i>txn_base.h</i>	176
<i>txnlog.h</i>	176
<i>webclnt.dsp</i>	179
Stored Procedures	180
<i>neword.sql</i>	180
<i>payment.sql</i>	182
<i>ordstat.sql</i>	184
<i>delivery.sql</i>	185
<i>Stocklev.sql</i>	186
<i>version.sql</i>	186
<b>Appendix B: Database Design</b>	188
Database Build	188
<i>createdb.sql</i>	188
<i>dbop1.sql</i>	189
<i>dbopt2.sql</i>	189
<i>idxcuscl.sql</i>	190
<i>idxcusnc.sql</i>	190
<i>idxdiscl.sql</i>	191
<i>idxhiscl.sql</i>	191
<i>idxitmcl.sql</i>	191
<i>idxnodcl.sql</i>	191
<i>idxodlcl.sql</i>	191
<i>idxordcl.sql</i>	192
<i>idxstkcl.sql</i>	192
<i>idxwarcl.sql</i>	192
<i>sqlshutdown.sql</i>	192
<i>tables.sql</i>	192
Load Source Code	194
<i>getargs.c</i>	194
<i>random.c</i>	196
<i>strings.c</i>	197
<i>time.c</i>	200
<i>tpcc.h</i>	200
<i>tpccldr.c</i>	201
<i>tpccldr.mak</i>	225
<i>verifytpccload.sql</i>	227
<i>version.sql</i>	228
<b>Appendix C: Tunable Parameters</b>	229
Microsoft SQL Server 2000 Configuration Parameters	229
Microsoft Windows Server 2003 Datacenter Edition	230
<i>Changes to the SUT</i>	230
<i>SUT System Information Report</i>	233
Disk Controller Configuration Parameters	273
<i>TotalStorage FC2-133 Host Bus Adapter 1</i>	273

<i>TotalStorage FC2-133 Host Bus Adapter 2</i> .....	273
<i>TotalStorage FC2-133 Host Bus Adapter 3</i> .....	274
<i>TotalStorage FC2-133 Host Bus Adapter 4</i> .....	275
<i>TotalStorage FC2-133 Host Bus Adapter 5</i> .....	276
<i>TotalStorage FC2-133 Host Bus Adapter 6</i> .....	277
Microsoft Windows 2000 Server .....	278
<i>Client System Information Report</i> .....	278
Registry Parameters .....	302
<i>Client 1</i> .....	302
<i>Client 2</i> .....	303
<i>Client 3</i> .....	305
<i>Client 4</i> .....	306
<i>Client 5</i> .....	307
<i>Client 6</i> .....	308
<i>Client 7</i> .....	309
<i>Client 8</i> .....	310
RTE Input Parameters .....	312
<b>Appendix D: 60-Day Space</b> .....	327
<b>Appendix E: Third-Party Quotations</b> .....	328

---

## 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 5.1.

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

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

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

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

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

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

---

## General Items

### Benchmark Sponsor

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

This benchmark was sponsored by International Business Machines Corporation.

### Application Code Disclosure and Definition Statements

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

Appendix A contains all source code implemented in this benchmark.

### Parameter Settings

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

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

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

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

### Configuration Diagrams

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

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

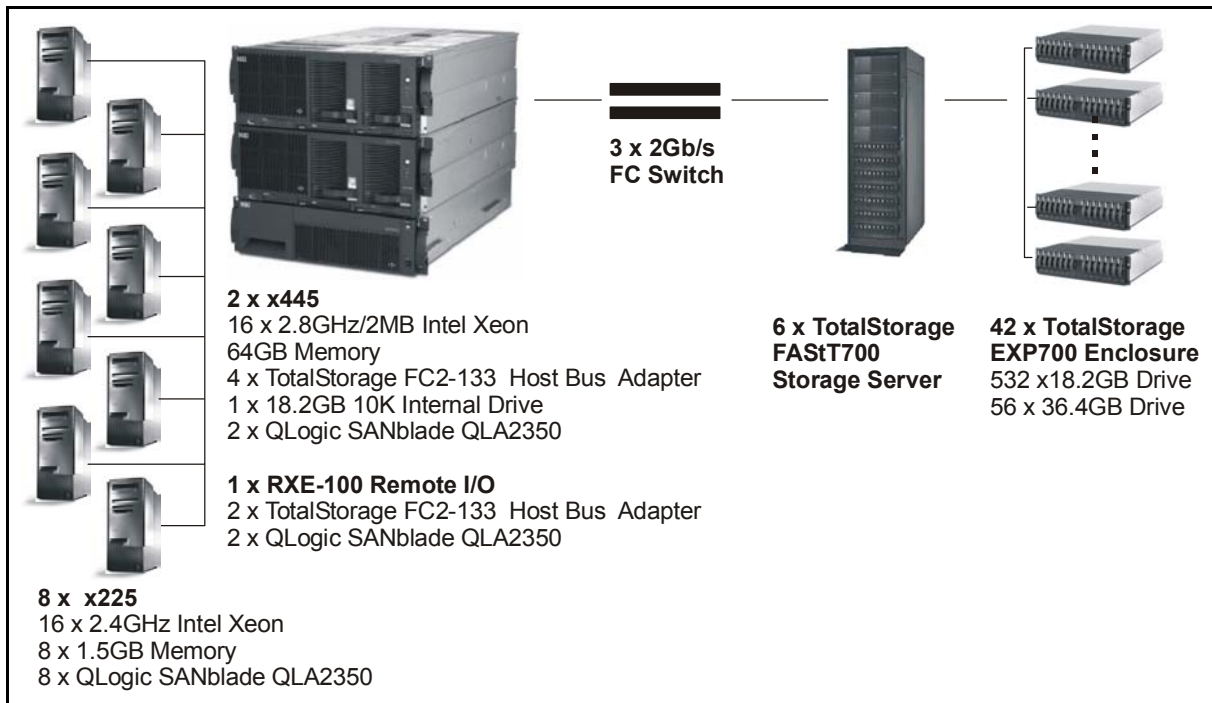
The Remote Terminal Emulator (RTE) used for these TPC Benchmark C tests is the Microsoft BenchCraft RTE. Under Version 5.1, the components of the configuration being emulated by the RTE are the workstations and the Ethernet hubs. Appendix C contains a listing of the RTE scripts and inputs used in the benchmark testing.

The measured configuration used eight IBM xSeries 225 systems, each configured with two 2.4GHz Xeon processors, as clients, which executed the terminal I/O and submitted transactions to COM+ servers, which were also running on the clients. These COM+ servers forwarded the transaction requests to the server, and returned the results to the RTE. Microsoft SQL Server 2000 Enterprise Edition was the DBMS executing on the server.

The server's disk subsystem capability was enhanced by using the IBM RXE-100 Expansion Enclosure to increase the number of PCI buses available. The RXE-100 added twelve PCI slots and six PCI buses to the server. The RAID HBA and LAN controllers were optimally distributed between the server and the RXE-100.

<b>Controller</b>	<b>PCI Slot Location</b>
TotalStorage FC2-133 Host Bus Adapter	Primary Server Slot 1
TotalStorage FC2-133 Host Bus Adapter	Primary Server Slot 5
TotalStorage FC2-133 Host Bus Adapter	Secondary Server Slot 1
TotalStorage FC2-133 Host Bus Adapter	Secondary Server Slot 3
QLogic SANblade QLA2350 FC -VI Adapter	Primary Server Slot 2
QLogic SANblade QLA2350 FC -VI Adapter	Secondary Server Slot 2
TotalStorage FC2-133 Host Bus Adapter	RXE-100 Slot 4
TotalStorage FC2-133 Host Bus Adapter	RXE-100 Slot 10
QLogic SANblade QLA2350 FC -VI Adapter	RXE-100 Slot 6
QLogic SANblade QLA2350 FC -VI Adapter	RXE-100 Slot 12

## Measured Configuration



The measured and priced configuration are identical.

---

## Clause 1: Logical Database Design Related Items

### Table Definitions

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

### Physical Organization of the Database

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

### Insert and Delete Operations

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

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

### Horizontal or Vertical Partitioning

*While there are few restrictions placed upon horizontal or vertical partitioning of tables and rows in the TPC-C benchmark (see Clause 1.6), any such partitioning must be disclosed. Partitioning was not used in this benchmark.*

### Replication

*Replication tables, if used, must be disclosed (see Clause 1.4.6). Replication was not used in this benchmark.*

### Table Attributes

*Additional and/or duplicated attributes in any table must be disclosed, along with a statement on the impact on performance (see Clause 1.4.7). No additional attributes were used in this benchmark.*

---

## Clause 2: Transaction and Terminal Profiles Related Items

### Random Number Generation

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

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

### Screen Layout

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

All screen layouts followed the TPC Benchmark C Standard Specification.

### Terminal Verification

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

The auditor verified terminal features by direct experimentation. The benchmarked configuration uses Microsoft Internet Explorer 6.0 and HTML scripts as the terminal interface.

### Intelligent Terminals

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

The terminals emulated in the priced configuration are IBM PC desktop computer systems. All processing of the input/output screens was handled by the x225 clients. The screen input/output was managed via HTML strings that comply with the HTML Version 2.0 specification. A listing of the code used to implement the intelligent terminals is provided in Appendix A. All data manipulation was handled by the x445 database server.

### Transaction Profiles

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

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

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

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

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



**Table 2-1. Transaction Statistics**

<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	0.85
Remote warehouse payment transactions	0.15
<b>Non-Primary Key Access</b>	
Payment transactions using C_LAST	60.01
Order-Status transactions using C_LAST	60.06
<b>Delivery</b>	
Delivery transactions skipped	0
<b>Transaction Mix</b>	
New-Order	44.94
Payment	43.03
Delivery	4.01
Stock Level	4.01
Order Status	4.01

### Deferred Delivery Mechanism

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

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

The source code is listed in Appendix A.

---

## Clause 3: Transaction and System Properties Related Items

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

### Atomicity Requirements

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

All ACID tests were conducted according to specification.

### Completed Transactions

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

1. The balance was retrieved from the CUSTOMER table for a random Customer, District and Warehouse, giving BALANCE\_1.
2. The Payment transaction was executed for the Customer, District and Warehouse used in step 1.
3. The balance was retrieved again for the Customer used in step 1 and step 2, giving BALANCE\_2. It was verified that BALANCE\_1 was greater than BALANCE\_2 by AMT.

### Aborted Transactions

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

1. The Payment application code was changed to execute a rollback of the transaction instead of performing the commit.
2. Using the balance, BALANCE\_2, from the CUSTOMER table retrieved for the completed transaction, the Payment transaction was executed for the Customer, District and Warehouse used in step 1 of section 3.1.1. The transaction rolled back due to the change in the application code from step 1.
3. The balance was retrieved again for the Customer used for step 2, giving BALANCE\_3. It was verified that BALANCE\_2 was equal to BALANCE\_3.

### Consistency Requirements

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

Consistency conditions one through four were tested using a bat file 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 the bat files to issue queries to the database. Each file included time stamps to demonstrate the concurrency of operations. The results of the queries were captured and placed in files. The auditor reviewed the results and verified that the isolation requirements had been met.

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

Case A was followed for Isolation test seven.

## Durability Requirements

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

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

## Loss of Data Test

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

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

## Loss of System (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. The test continued and the system continued to run for another 6 minutes after all users were connected to the server.
4. The server under test was powered off, which removed power from the system and the memory.
5. The server was powered on again.

6. SQL Server was started to initiate automatic recovery from its log.
7. Step 1 was repeated to obtain the current count of the total number of orders giving SUM2.
8. 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 Log Disk and Log Cache***

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. This test was executed on a full-scale benchmark run.
3. The test continued with a checkpoint issued and completed.
4. One disk from the log array was removed. Since the disk was RAID-1 mirrored, SQL Server continued to process transactions without interruption.
5. The test continued to run for another 5 minutes.
6. Since write caching and write cache mirroring were enabled on the log's Fibre Channel controller pair, one of the controllers and its cache was then failed. Disk I/O paused briefly while the remaining log controller's cache was flushed and the write policy was automatically changed to write-through mode.
7. The test was allowed to run for another 5 minutes.
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 stopped is greater than or equal to the sum of D\_NEXT\_O\_ID before the run, plus all new order transactions completed during the run minus any rollback transactions.

---

## Clause 4: Scaling and Database Population Related Items

### Cardinality of Tables

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

The database was built with 15,232 warehouses and the audited run used all 15,232 warehouses.

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

Table Name	Rows
Warehouse	15,232
District	152,320
Item	100,000
New Order	137,088,000
History	456,960,000
Orders	456,960,000
Customer	456,960,000
Order Line	4,852,606,786
Stock	1,523,200,000
Inactive Warehouses	0

### Distribution of Tables and Logs

*The distribution of tables and logs across all media must be explicitly depicted for the tested and priced systems. )*

Figure 4-2 depicts the priced database configuration of the tested system to meet the 8-hour steady state requirement

**Figure 4-2. Data Distribution for the Benchmarked Configuration**

Disk #	Drives	Partition	Size	Use
0	28 - 36.4GB	E: F:	431.65GB 41.50GB (NTFS)	Logfile MDF File
1	28 - 18.2GB	C:\mp\c11 C:\mp\m11 Q:	45.9GB 25.39GB 401.37GB (NTFS)	Customer and Stock Miscellaneous Files Backup files
2	28 - 18.2GB	C:\mp\c12 C:\mp\m12 R:	45.9GB 25.39GB 401.37GB (NTFS)	Customer and Stock Miscellaneous Files Backup files
3	28 - 18.2GB	C:\mp\c13 C:\mp\m13 S:	45.9GB 25.39GB 401.37GB (NTFS)	Customer and Stock Miscellaneous Files Backup files
4	28 - 18.2GB	C:\mp\c14 C:\mp\m14 T:	45.9GB 25.39GB 401.37GB (NTFS)	Customer and Stock Miscellaneous Files Backup files
5	28 - 18.2GB	C:\mp\c15 C:\mp\m15 U:	45.9GB 25.39GB 401.37GB (NTFS)	Customer and Stock Miscellaneous Files Backup files
6	28 - 18.2GB	C:\mp\c16 C:\mp\m16 V:	45.9GB 25.39GB 401.37GB (NTFS)	Customer and Stock Miscellaneous Files Backup files
7	28 - 18.2GB	C:\mp\c17 C:\mp\m17 W:	45.9GB 25.39GB 401.37GB (NTFS)	Customer and Stock Miscellaneous Files Backup files
8	28 - 18.2GB	C:\mp\c18 C:\mp\m18 x:	45.9GB 25.39GB 401.37GB (NTFS)	Customer and Stock Miscellaneous Files Backup files
9	28 - 18.2GB	C:\mp\c23 C:\mp\m23 c:\mp\disk23	45.9GB 25.39GB 401.37GB (NTFS)	Customer and Stock Miscellaneous Files Backup files
10	28 - 18.2GB	C:\mp\c24 C:\mp\m24 c:\mp\disk24	45.9GB 25.39GB 401.37GB (NTFS)	Customer and Stock Miscellaneous Files Backup files
11	28 - 18.2GB	C:\mp\c25 C:\mp\m25 c:\mp\disk25	45.9GB 25.39GB 401.37GB (NTFS)	Customer and Stock Miscellaneous Files Backup files
12	28 - 18.2GB	C:\mp\c26 C:\mp\m26 c:\mp\disk26	45.9GB 25.39GB 401.37GB (NTFS)	Customer and Stock Miscellaneous Files Backup files
13	28 - 18.2GB	C:\mp\c7 C:\mp\m7 M:	45.9GB 25.39GB 401.37GB (NTFS)	Customer and Stock Miscellaneous Files Backup files
14	28 - 18.2GB	C:\mp\c8 C:\mp\m8 N:	45.9GB 25.39GB 401.37GB (NTFS)	Customer and Stock Miscellaneous Files Backup files
15	28 - 18.2GB	C:\mp\c9 C:\mp\m9 O:	45.9GB 25.39GB 401.37GB (NTFS)	Customer and Stock Miscellaneous Files Backup files
16	28 - 18.2GB	C:\mp\c10 C:\mp\m10 P:	45.9GB 25.39GB 401.37GB (NTFS)	Customer and Stock Miscellaneous Files Backup files

17	28 - 18.2GB	C:\mp\c19 C:\mp\m19 C:\mp\disk19	45.9GB 25.39GB 401.37GB (NTFS)	Customer and Stock Miscellaneous Files Backup files
18	28 - 18.2GB	C:\mp\c20 C:\mp\m20 c:\mp\disk20	45.9GB 25.39GB 401.37GB (NTFS)	Customer and Stock Miscellaneous Files Backup files
19	28 - 18.2GB	C:\mp\c21 C:\mp\m21 c:\mp\disk21	45.9GB 25.39GB 401.37GB (NTFS)	Customer and Stock Miscellaneous Files Backup files
20	28 - 36.2GB	C:\mp\c22 C:\mp\m22 C:\mp\disk22	45.9GB 25.39GB 401.37GB (NTFS)	Customer and Stock Miscellaneous Files Backup files
21	1 - 18.2GB	C:	16.94GB (NTFS)	Operating System

## 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.*

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

## Partitions/Replications Mapping

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

The database was neither partitioned nor replicated.

## 60-Day Space Requirement

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

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

---

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

### Measured tpmC

*Measured tpmC must be reported.*

Measured tpmC: 190,510.02 tpmC

Price per tpmC: \$8.39 per tpmC

### Response Times

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

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

**Table 5-1. Response Times in Seconds**

Transaction Type	Average	Maximum	90 %-tile
New-Order	0.37	16.25	0.47
Payment	0.31	17.44	0.41
Delivery	0.16	4.58	0.11
Stock Level	0.70	12.76	1.08
Order Status	0.35	16.77	0.45
Delivery (Deferred)	0.14	4.64	0.19
Menu	0.17	4.61	0.11

### Keying/Think Times

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

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

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

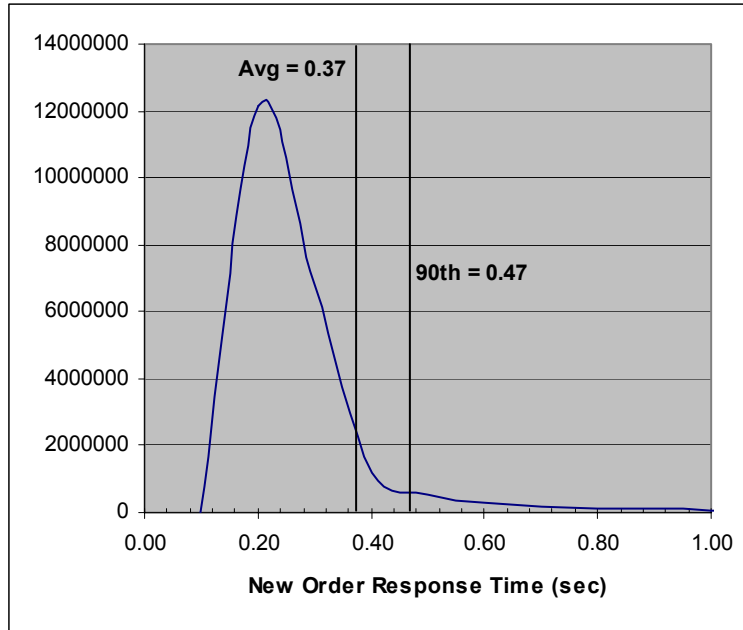
Transaction Type	Average	Minimum	Maximum
New-Order	18.02 / 12.05	18.00 / 0.00	18.05 / 120.51
Payment	3.02 / 12.05	3.00 / 0.00	3.05 / 120.90
Delivery	2.02 / 5.06	2.00 / 0.00	2.04 / 50.51
Stock Level	2.02 / 5.05	2.00 / 0.00	2.04 / 50.51
Order Status	2.02 / 10.04	2.00 / 0.00	2.04 / 100.51



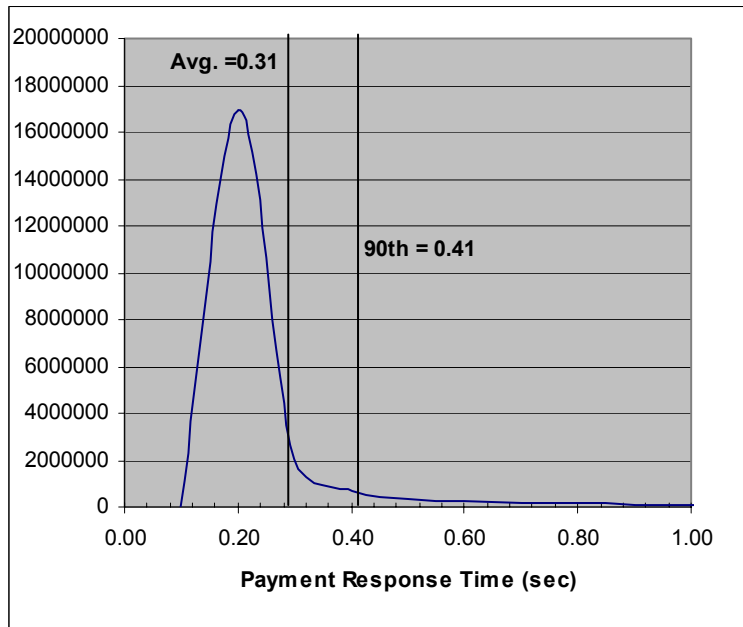
## Response Time Frequency Distribution Curves

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

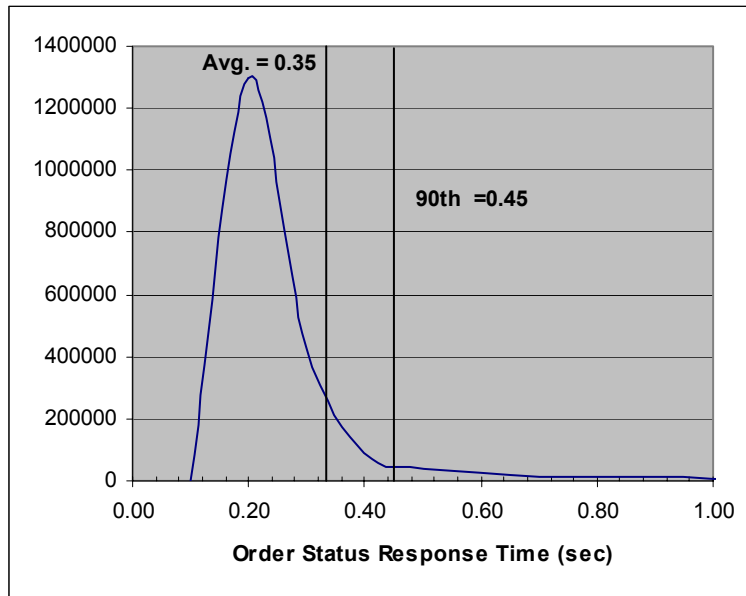
**Figure 5-1. New-Order Transaction - Response Time Frequency Distribution**



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



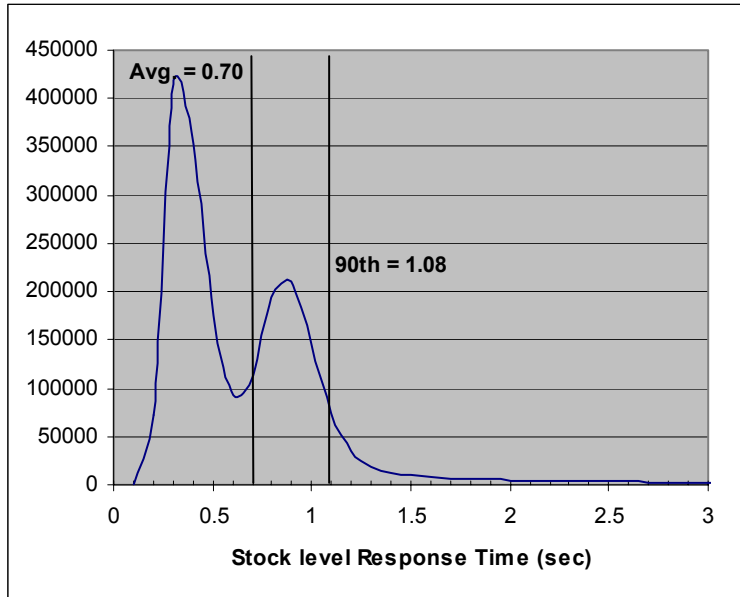
**Figure 5-3. Order-Status Transaction - Response Time Frequency Distribution**



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



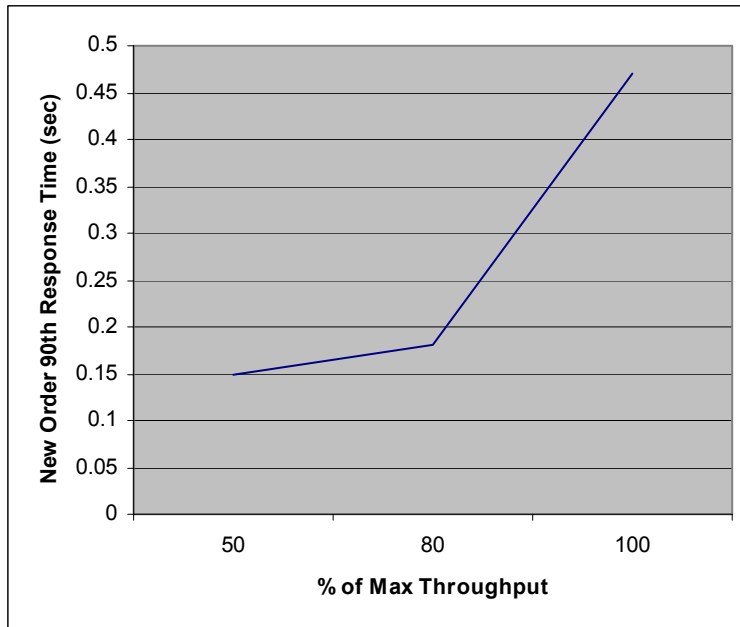
**Figure 5-5. Stock-Level Transaction - Response Time Frequency Distribution**



### Performance Curve for Response Time vs. Throughput

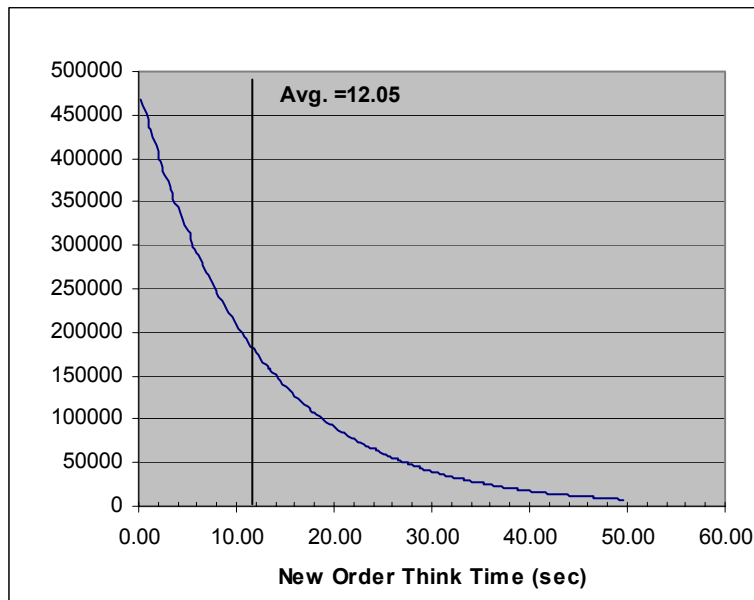
*The performance curve for response time vs. throughput (see Clause 5.6.2) must be reported for the New-Order transaction.*

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



## New Order Think Time Distribution

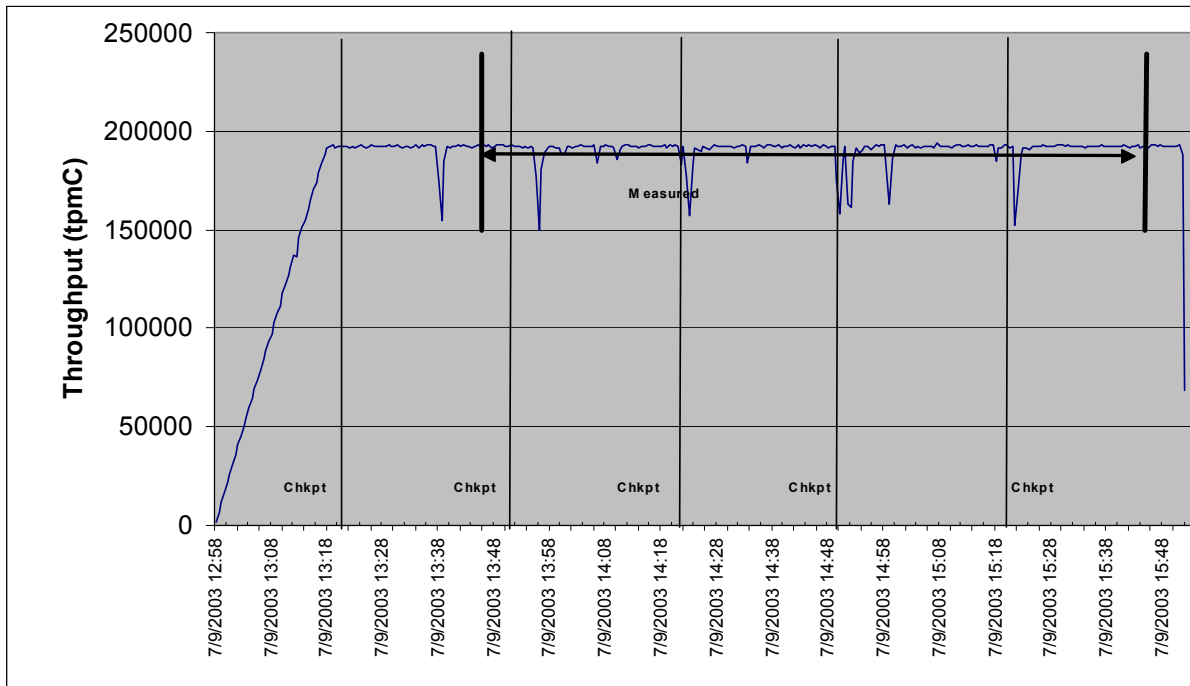
Figure 5-7. New-Order Think Time Distribution



## Throughput vs. Elapsed Time

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

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



## Steady State Methodology

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

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

## Work Performed during Steady State

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

### Transaction Flow

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

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

The RTE transmissions were sent to application processes running on the client machines through Ethernet LANs. These client application processes handled all screen I/O as well as all requests to the database on the server. The applications communicated with the database server over a 2Gb/s Fibre Channel network using Microsoft SQL Server ODBC library and RPC calls.

## Checkpoints

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

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

## Measurement Interval

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

The measurement interval was 120 minutes.

## Transaction Mix

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

See Table 5-3.

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

## 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	0.85
Remote warehouse payment transactions	0.15
<b>Non-Primary Key Access</b>	
Payment transactions using C_LAST	60.01
Order-Status transactions using C_LAST	60.06
<b>Delivery</b>	
Delivery transactions skipped	0
<b>Transaction Mix</b>	
New-Order	44.94
Payment	43.03
Delivery	4.01
Stock Level	4.01
Order-Status	4.01

## Number of Checkpoints

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

Checkpoints were performed during the ramp-up period and during each measured run interval. The first measurement interval checkpoint started 41 seconds after the start of the measurement interval. The four checkpoints in the measured interval are shown in Table 5-4.

**Table 5-4. Checkpoint Start Time and Duration**

<b>Checkpoint</b>	<b>Start Time</b>	<b>Duration</b>
1	1:51:41.10 p.m.	28 minutes
2	2:21:36.93 p.m.	28 minutes
3	2:51:33.02 p.m.	28 minutes
4	3:21:28.94 p.m.	28 minutes

The checkpoint interval was 30 minutes.

---

## Clause 6: SUT, Driver and Communication Definition Related Items

### Description of RTE

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

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

### Emulated Components

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

No components were emulated.

### Benchmarked and Targeted System Configuration Diagrams

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

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

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

### Network Configuration

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

See the measured and priced configuration diagrams for details about the network configuration.

### Network Bandwidth

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

The Ethernet used in the LAN connecting the clients and driver RTEs complies with the IEEE.802.3 standard. The Ethernet LAN had a bandwidth of 10Mbps. The LAN that connected the clients to the server was a Fibre Channel network whose bandwidth was 2Gb/s.

### Operator Intervention

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

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



---

## Clause 7: Pricing Related Items

### Hardware and Software Components

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

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

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

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

### Availability Date

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

The total system as priced will be available December 31, 2003.

### Measured tpmC

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

- Maximum Qualified Throughput: 190,510.02 tpmC
- Price per tpmC: \$8.39 per tpmC
- Three-year cost of ownership: \$1,599,308

### Country-Specific Pricing

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

The configuration is priced for the United States of America.

### Usage Pricing

*For any usage pricing, the sponsor must disclose:*

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

The component pricing based on usage is shown below:

- Microsoft Windows Server 2003 Datacenter Edition (1-16 processors)
- 16 Microsoft SQL Server 2000 Enterprise Edition (based on per-processor price)
- 8 Microsoft Windows 2000 Server

- 3-year support for hardware components (except for components for which 10 percent spares are provided)

## **System Pricing**

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

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

---

## Clause 9: Audit Related Items

### Auditor

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

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

### Availability of the Full Disclosure Report

*The Full Disclosure Report must be readily available to the public at a reasonable charge, similar to the charges for similar documents by the test sponsor. The report must be made available when results are made public. In order to use the phrase "TPC Benchmark<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.*

The TPC Benchmark C Full Disclosure Report can be obtained from [www.tpc.org](http://www.tpc.org).

Benchmark Sponsor: Kamran Amini  
 Manager, xSeries Performance  
 IBM Systems Group  
 3039 Cornwallis Road  
 Research Triangle Park, NC 27709

July 17, 2003

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

Platform: **IBM @server xSeries 445 c/s**  
 Operating system: **Microsoft Windows Server 2003 Datacenter Edition w/QFE**  
 Database Manager: **Microsoft SQL Server 2000 Enterprise Edition SP3 w/QFE**  
 Transaction Manager: **Microsoft COM+**

The results were:

CPU's Speed	Memory	Disks	NewOrder 90% Response Time	tpmC
<b>Server: IBM @server xSeries 445</b>				
16 x Xeon DP (2.8GHz)	64 GB Main (2MB L3 Cache)	533 x 18.2 GB 56 x 36.4 GB	0.47 Seconds	<b>190,510.02</b>
<b>Clients: Eight (8) IBM @server xSeries 225 (Specification for each)</b>				
2 x Xeon DP (2.4 GHz)	1.5 GB Main (512 KB L2 Cache per processor)	1 x 18.2 GB	n/a	n/a

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


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

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

Additional Audit Notes:

None.

Respectfully Yours,



François Raab, President



Bradley J. Askins, Auditor

# Appendix A: Source Code

## Web Client Source Code

### db\_odbc\_dll.dsp

```
# Microsoft Developer Studio Project File - Name="db_odbc_dll" - Package
Owner=<4>
# Microsoft Developer Studio Generated Build File, Format Version 6.00
# ** DO NOT EDIT **
```

```
# TARGETTYPE "Win32 (x86) Dynamic-Link Library" 0x0102
```

```
CFG=db_odbc_dll - Win32 IceCAP
!MESSAGE This is not a valid makefile. To build this project using NMAKE,
!MESSAGE use the Export Makefile command and run
!MESSAGE
!MESSAGE NMAKE /f "db_odbc_dll.mak".
!MESSAGE
!MESSAGE You can specify a configuration when running NMAKE
!MESSAGE by defining the macro CFG on the command line. For example:
!MESSAGE
!MESSAGE NMAKE /f "db_odbc_dll.mak" CFG="db_odbc_dll - Win32
IceCAP"
!MESSAGE
!MESSAGE Possible choices for configuration are:
!MESSAGE
!MESSAGE "db_odbc_dll - Win32 Release" (based on "Win32 (x86)
Dynamic-Link Library")
!MESSAGE "db_odbc_dll - Win32 Debug" (based on "Win32 (x86)
Dynamic-Link Library")
!MESSAGE "db_odbc_dll - Win32 IceCAP" (based on "Win32 (x86)
Dynamic-Link Library")
!MESSAGE
```

```
# Begin Project
# PROP AllowPerConfigDependencies 0
# PROP Scc_ProjName ""
# PROP Scc_LocalPath ""
CPP=cl.exe
MTL=midl.exe
RSC=rc.exe
```

```
!IF "$(CFG)" == "db_odbc_dll - 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 ".\obj"
# PROP Ignore_Export_Lib 0
# PROP Target_Dir ""
# ADD BASE CPP /nologo /MT /W3 /GX /O2 /D "WIN32" /D "NDEBUG" /D
"_WINDOWS" /YX /FD /c
# ADD CPP /nologo /MD /W3 /GX /O2 /D "WIN32" /D "NDEBUG" /D
"_WINDOWS" /YX /FD /c
# ADD BASE MTL /nologo /D "NDEBUG" /mktyplib203 /o /win32 "NUL"
# ADD MTL /nologo /D "NDEBUG" /mktyplib203 /o /win32 "NUL"
# ADD BASE RSC /I 0x409 /d "NDEBUG"
# ADD RSC /I 0x409 /d "NDEBUG"
BSC32=bscmake.exe
```

```
# ADD BASE BSC32 /nologo
# ADD BSC32 /nologo
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 odbccp32.lib
odbccp32.lib /nologo /subsystem:windows /dll /machine:I386
# ADD LINK32 kernel32.lib user32.lib gdi32.lib winspool.lib comdlg32.lib
advapi32.lib shell32.lib ole32.lib oleaut32.lib uuid.lib odbccp32.lib
/nologo /subsystem:windows /dll /machine:I386 /out:".bin\tpcc_odbc.dll"
```

```
!ELSEIF "$(CFG)" == "db_odbc_dll - 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 ".\obj"
# PROP Ignore_Export_Lib 0
# PROP Target_Dir ""
# ADD BASE CPP /nologo /MTd /W3 /Gm /GX /Zi /Od /D "WIN32" /D
"_DEBUG" /D "_WINDOWS" /YX /FD /c
# ADD CPP /nologo /MDd /W3 /GX /Zi /Od /D "WIN32" /D "_DEBUG" /D
"_WINDOWS" /YX /FD /c
# ADD BASE MTL /nologo /D "_DEBUG" /mktyplib203 /o /win32 "NUL"
# ADD MTL /nologo /D "_DEBUG" /mktyplib203 /o /win32 "NUL"
# ADD BASE RSC /I 0x409 /d "_DEBUG"
# ADD RSC /I 0x409 /d "_DEBUG"
BSC32=bscmake.exe
# ADD BASE BSC32 /nologo
# ADD BSC32 /nologo
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 odbccp32.lib
odbccp32.lib /nologo /subsystem:windows /dll /debug /machine:I386
/pdbtype:sept
# ADD LINK32 kernel32.lib user32.lib gdi32.lib winspool.lib comdlg32.lib
advapi32.lib shell32.lib ole32.lib oleaut32.lib uuid.lib odbccp32.lib
/nologo /subsystem:windows /dll /debug /machine:I386
/out:".bin\tpcc_odbc.dll" /pdbtype:sept
```

```
!ELSEIF "$(CFG)" == "db_odbc_dll - Win32 IceCAP"
```

```
# PROP BASE Use_MFC 0
# PROP BASE Use_Debug_Libraries 1
# PROP BASE Output_Dir "db_odbc_"
# PROP BASE Intermediate_Dir "db_odbc_"
# PROP BASE Ignore_Export_Lib 0
# PROP BASE Target_Dir ""
# PROP Use_MFC 0
# PROP Use_Debug_Libraries 1
# PROP Output_Dir ".\bin"
# PROP Intermediate_Dir ".\obj"
# PROP Ignore_Export_Lib 0
# PROP Target_Dir ""
# ADD BASE CPP /nologo /MDd /W3 /Gm /GX /Zi /Od /D "WIN32" /D
"_DEBUG" /D "_WINDOWS" /YX /FD /Gh /c
# ADD CPP /nologo /MD /W3 /Gm /GX /Zi /O2 /D "WIN32" /D "NDEBUG"
/D "_WINDOWS" /D "ICECAP" /YX /FD /Gh /c
# ADD BASE MTL /nologo /D "_DEBUG" /mktyplib203 /o /win32 "NUL"
# ADD MTL /nologo /D "_DEBUG" /mktyplib203 /o /win32 "NUL"
# ADD BASE RSC /I 0x409 /d "_DEBUG"
# ADD RSC /I 0x409 /d "_DEBUG"
BSC32=bscmake.exe
# ADD BASE BSC32 /nologo
```

```

# ADD BSC32 /nologo
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 odbccp32.lib
odbccp32.lib /nologo /subsystem:windows /dll /debug /machine:I386
/out:".bin\tpcc_odbc.dll" /pdbtype:sept
# ADD LINK32 icap.lib kernel32.lib user32.lib gdi32.lib winspool.lib
comdlg32.lib advapi32.lib shell32.lib ole32.lib oleaut32.lib uuid.lib odbccp32.lib
odbccp32.lib /nologo /subsystem:windows /dll /debug /machine:I386
/out:".bin\tpcc_odbc.dll" /pdbtype:sept

!ENDIF

# Begin Target

# Name "db_odbc_dll - Win32 Release"
# Name "db_odbc_dll - Win32 Debug"
# Name "db_odbc_dll - Win32 IceCAP"
# Begin Group "Source"

# PROP Default_Filter "*.cpp"
# Begin Source File

SOURCE=.src\tpcc_odbc.cpp
# End Source File
# End Group
# Begin Group "Header"

# PROP Default_Filter "*.h"
# Begin Source File

SOURCE=..\common\src\error.h
# End Source File
# Begin Source File

SOURCE=.src\tpcc_odbc.h
# End Source File
# Begin Source File

SOURCE=..\common\src\trans.h
# End Source File
# Begin Source File

SOURCE=..\common\src\txn_base.h
# End Source File
# End Group
# End Target
# End Project

```

## dlldata.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
/dlldata command line option

*****/
#include <rpcproxy.h>

```

```

#ifdef _cplusplus
extern "C" {
#endif

EXTERN_PROXY_FILE( tpcc_com_ps )

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

DLLDATA_ROUTINES( aProxyFileList, GET_DLL_CLSID )

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

/* end of generated dlldata 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

#ifdef _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
#define ERR_BUF_OVERFLOW
25 //Buffer overflow during receive
// 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_TYPE_SCHANNEL
57

#define ERR_INS_MEMORY "Insufficient Memory
to continue."
#define ERR_UNKNOWN "Unknown
error."
#define ERR_MSG_BUF_SIZE 512
#define INV_ERROR_CODE -1
#define ERR_INS_BUF_OVERFLOW "Insufficient Buffer size to receive
HTML pages."

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

    j += wsprintf(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;

    //short     m_errType;
};

```

```

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

    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,
        eFSeek,
        eFRead,
        eFWrite,
        eTmpFile,
        eSetFilePointer,
        eNew,
    };

    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;}
};

class CBufferOverflowErr : public CBaseErr
{
public:
    CBufferOverflowErr(int,LPTSTR);

    int ErrorType() {return ERR_BUF_OVERFLOW;}

    char *ErrorText() {return ERR_INS_BUF_OVERFLOW;}
};

```

## 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 versionDllMS;
DWORD versionDllLS;

// 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 CheckWWWService(void);
static BOOL StartWWWService(void);
static BOOL StopWWWService(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, (LPARAM)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);
            }
            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);
        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);
            default:
                break;
    }
    return FALSE;
}

}

PAINTSTRUCT ps;
MEMORYSTATUS memoryStatus;
OSVERSIONINFO VI;
char szTmp[256];
static char szDllPath[256];
static char szExePath[256];

switch(uMsg)
{
    case WM_INITDIALOG:
        GlobalMemoryStatus(&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( &Reg, sizeof(Reg) );
        Reg.dwNumberOfDeliveryThreads = 4;
        Reg.dwMaxConnections = 100;
        Reg.dwMaxPendingDeliveries = 100;
        Reg.eDB_Protocol = DBLIB;
        Reg.eTxnMon = None;
        strcpy(Reg.szDbServer,
"" );
        strcpy(Reg.szDbName,
"tpcc");
        strcpy(Reg.szDbUser,
"sa");
        strcpy(Reg.szDbPassword,
"" );

        iPoolThreadLimit = iMaxPhysicalMemory *
2;

        iThreadTimeout = 86400;
        iListenBackLog = 15;
        iAcceptExOutstanding = 40;

        ReadTPCCRegistrySettings( &Reg );
        ReadRegistrySettings();

        GetModuleFileName(hInst, szExePath,
sizeof(szExePath));
        GetVersionInfo(szDllPath, szExePath);

        wsprintf(szTmp, "Version %d.%2d.%3d",
versionExeMS, versionExeMM, versionExeLS);
        SetDlgItemText(hwnd, IDC_VERSION,
szTmp);

        SetDlgItemText(hwnd, IDC_PATH,
szDllPath);
}
}

```

```

Reg.szDbServer);
SetDlgItemText(hwnd, ED_DB_SERVER,
Reg.szDbUser);
SetDlgItemText(hwnd, ED_DB_USER_ID,
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
    CheckDlgButton(hwnd,
IDC_ODBC, 1);

// check OS version level for COM. Must be
at least Windows 2000
VI.dwOSVersionInfoSize = sizeof(VI);
GetVersionEx( &VI );
if (VI.dwMajorVersion < 5)
{
    HWND hDlg = GetDlgItem(
hwnd, IDC_TM_MTS );
    EnableWindow( hDlg, 0 ); //
    disable COM option
    if (Reg.eTxnMon == COM)
        Reg.eTxnMon = None;
}

CheckDlgButton(hwnd, IDC_TM_NONE,
0);
CheckDlgButton(hwnd, IDC_TM_TUXEDO,
0);
CheckDlgButton(hwnd, IDC_TM_MTS, 0);
CheckDlgButton(hwnd, 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, &ps);
        DrawIcon(ps.hdc, 0, 0, hIcon);
        EndPaint(hwnd, &ps);
        return TRUE;
    }
    break;
case WM_COMMAND:
    if ( HIWORD(wParam) == BN_CLICKED )
    {
        switch( LOWORD(wParam) )
        {
            case IDC_DBLIB:
                return
                TRUE;
            case IDC_ODBC:
                return
                TRUE;
            case IDOK:
                ProcessOK(hwnd, szDllPath);
                return
                TRUE;
            case IDCANCEL:
                EndDialog(hwnd, FALSE);
                return
                TRUE;
            default:
                return
                FALSE;
        }
    }
    break;
}
default:
    break;
}
return FALSE;
}

static void ProcessOK(HWND hwnd, char *szDllPath)
{
    int d;
    HWND hDlg;
    int rc;

    char szFullName[256];
    char szErrMsg[128];

    // read settings from dialog
    Reg.dwNumberOfDeliveryThreads = GetDlgItemInt(hwnd,
ED_THREADS, &d, FALSE);
    Reg.dwMaxConnections = GetDlgItemInt(hwnd,
ED_MAXCONNECTION, &d, FALSE);
    Reg.dwMaxPendingDeliveries = GetDlgItemInt(hwnd,
ED_MAXDELIVERIES, &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) )
                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) ocured 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 ocured 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 ocured 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 = 15;

        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, 16));
        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 = CheckWWWebService();
    if ( bSvcRunning )
    {
        SetDlgItemText(hDlg, IDC_STATUS, "Stopping Web
Service.");
        SendDlgItemMessage(hDlg, IDC_PROGRESS1,
PBM_STEPIT, 0, 0);
        UpdateDialog(hDlg);

        StopWWWebService();
        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);

//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;
    int iRc;

    // Registry key
    HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\InetStp\PathWWWRoot
is used to find the
    // IIS default web site directory and determine that IIS is installed.

    szDllPath[0] = 0;
    bRc = TRUE;
    if ( RegOpenKeyEx(HKEY_LOCAL_MACHINE,
"SOFTWARE\Microsoft\InetStp", 0, KEY_ALL_ACCESS, &hKey) ==
ERROR_SUCCESS )
    {
        sv = sizeof(szData);
        iRc = RegQueryValueEx( hKey, "PathWWWRoot",
NULL, NULL, szData, &sv ); // used by IIS 5.0 & 6.0
        if (iRc == ERROR_SUCCESS)
        {
            bRc = FALSE;
            strcpy(szDllPath, szData);
            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;

    versionDllMS = 0;
    versionDllLS = 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);
            versionDllMS = vs->dwProductVersionMS;
            versionDllLS = 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 CheckWWWWebService(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.dsp**

```

# Microsoft Developer Studio Project File - Name="install" - Package
Owner=<4>
# Microsoft Developer Studio Generated Build File, Format Version 6.00
# ** DO NOT EDIT **

# TARGETTYPE "Win32 (x86) Application" 0x0101

CFG=install - Win32 Release
!MESSAGE This is not a valid makefile. To build this project using NMAKE,
!MESSAGE use the Export Makefile command and run
!MESSAGE
!MESSAGE NMAKE /f "install.mak".
!MESSAGE
!MESSAGE You can specify a configuration when running NMAKE
!MESSAGE by defining the macro CFG on the command line. For example:
!MESSAGE
!MESSAGE NMAKE /f "install.mak" CFG="install - Win32 Release"
!MESSAGE

```

```
!MESSAGE Possible choices for configuration are:
!MESSAGE
!MESSAGE "install - Win32 Release" (based on "Win32 (x86) Application")
!MESSAGE "install - Win32 Debug" (based on "Win32 (x86) Application")
!MESSAGE
```

```
# Begin Project
# PROP AllowPerConfigDependencies 0
# PROP Scc_ProjName ""
# PROP Scc_LocalPath ""
CPP=cl.exe
MTL=midl.exe
RSC=rc.exe
```

```
!IF "$(CFG)" == "install - 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 ".\obj"
# PROP Ignore_Export_Lib 0
# PROP Target_Dir ""
# ADD BASE CPP /nologo /W3 /GX /O2 /D "WIN32" /D "NDEBUG" /D
 "_WINDOWS" /YX /c
# ADD CPP /nologo /W3 /GX /O2 /D "WIN32" /D "NDEBUG" /D
 "_WINDOWS" /YX /FD /c
# ADD BASE MTL /nologo /D "NDEBUG" /win32
# ADD MTL /nologo /D "NDEBUG" /mktyplib203 /win32
# ADD BASE RSC /I 0x409 /d "NDEBUG"
# ADD RSC /I 0x409 /d "NDEBUG"
BSC32=bscmake.exe
# ADD BASE BSC32 /nologo
# ADD BSC32 /nologo
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 odbcc32.lib
 odbccp32.lib /nologo /subsystem:windows /machine:I386
# ADD LINK32 version.lib comctl32.lib kernel32.lib user32.lib gdi32.lib
 winspool.lib comdlg32.lib advapi32.lib shell32.lib ole32.lib oleaut32.lib
 uuid.lib odbcc32.lib odbccp32.lib /nologo /subsystem:windows /machine:I386
/out:"..\bin\install.exe"
```

```
!ELSEIF "$(CFG)" == "install - 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 ".\obj"
# PROP Ignore_Export_Lib 0
# PROP Target_Dir ""
# ADD BASE CPP /nologo /W3 /Gm /GX /Zi /Od /D "WIN32" /D "_DEBUG"
 /D "_WINDOWS" /YX /c
# ADD CPP /nologo /W3 /Gm /GX /Zi /Od /D "WIN32" /D "_DEBUG" /D
 "_WINDOWS" /YX /FD /c
# ADD BASE MTL /nologo /D "_DEBUG" /win32
# ADD MTL /nologo /D "_DEBUG" /mktyplib203 /win32
# ADD BASE RSC /I 0x409 /d "_DEBUG"
# ADD RSC /I 0x409 /d "_DEBUG"
```

```
BSC32=bscmake.exe
# ADD BASE BSC32 /nologo
# ADD BSC32 /nologo
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 odbcc32.lib
 odbccp32.lib /nologo /subsystem:windows /debug /machine:I386
# ADD LINK32 version.lib comctl32.lib kernel32.lib user32.lib gdi32.lib
 winspool.lib comdlg32.lib advapi32.lib shell32.lib ole32.lib oleaut32.lib
 uuid.lib odbcc32.lib odbccp32.lib /nologo /subsystem:windows /debug
 /machine:I386 /out:"..\bin\install.exe"
```

```
!ENDIF
```

```
# Begin Target
```

```
# Name "install - Win32 Release"
# Name "install - Win32 Debug"
# Begin Group "Source Files"
```

```
# PROP Default_Filter "cpp;c;cxx;rc;def;r;odl;hpj;bat;for;f90"
# Begin Source File
```

```
SOURCE=\src\install.c
# End Source File
# Begin Source File
```

```
SOURCE=\src\install.rc
# ADD BASE RSC /I 0x409 /i "src"
# ADD RSC /I 0x409 /i "src" /i ".\src"
# End Source File
# Begin Source File
```

```
SOURCE=\src\install_com.cpp
# End Source File
# End Group
# Begin Group "Header Files"
```

```
# PROP Default_Filter "h;hpp;hxx;hm;inl;fi;fd"
# End Group
# Begin Group "Resource Files"
```

```
# PROP Default_Filter "ico;cur;bmp;dlg;rc2;rt;bin;cnt;rtf;gif;jpg;jpeg;jpe"
# Begin Source File
```

```
SOURCE=\SRC\ICON1.ICO
# End Source File
# Begin Source File
```

```
SOURCE=\SRC\ICON2.ICO
# End Source File
# End Group
# Begin Source File
```

```
SOURCE=\SRC\LICENSE.TXT
# End Source File
# Begin Source File
```

```
SOURCE=..\isapi_dll\bin\tpcc.dll
# End Source File
# Begin Source File
```

```
SOURCE=..\tm_com_dll\bin\tpcc_com.dll
# End Source File
# Begin Source File
```

```
SOURCE=..\tpcc_com_all\bin\tpcc_com_all.dll
# End Source File
```

```

# Begin Source File

SOURCE=..\tpcc_com_ps\bin\tpcc_com_ps.dll
# End Source File
# Begin Source File

SOURCE=..\db_dblib_dll\bin\tpcc_dblib.dll
# End Source File
# Begin Source File

SOURCE=..\db_odbc_dll\bin\tpcc_odbc.dll
# End Source File
# Begin Source File

SOURCE=..\tm_tuxedo_dll\bin\tpcc_tuxedo.dll
# End Source File
# Begin Source File

SOURCE=..\tuxapp\bin\tuxapp.exe
# End Source File
# End Target
# End Project

```

## 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 TEXTINCLUDE 2 resource.
//
#include "afxres.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 // _WIN32

//
// Dialog
//

IDD_DIALOG1 DIALOGEX 0, 0, 219, 351
STYLE DS_MODALFRAME | DS_CENTER | WS_MINIMIZEBOX |
WS_POPUP | WS_CAPTION |
WS_SYSMENU
CAPTION "TPC-C Web Client Installation Utility"
FONT 8, "MS Sans Serif"
BEGIN
    EDITTEXT    ED_THREADS,164,45,34,12,ES_RIGHT | ES_NUMBER,
                WS_EX_RTREADING
    EDITTEXT    ED_MAXDELIVERIES,164,59,34,12,ES_RIGHT |
ES_NUMBER,
                WS_EX_RTREADING
    EDITTEXT    ED_MAXCONNECTION,164,73,34,12,ES_RIGHT |
ES_NUMBER,
                WS_EX_RTREADING
    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_AUTORADIOBUTTON |
                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,131,152,67,12,ES_AUTOHSCROLL
    EDITTEXT    ED_DB_USER_ID,131,165,67,12,ES_AUTOHSCROLL
    EDITTEXT    ED_DB_PASSWORD,131,178,67,12,ES_AUTOHSCROLL
    EDITTEXT    ED_DB_NAME,131,191,67,12,ES_AUTOHSCROLL
    CONTROL    "DBLIB",IDC_DBLIB,"Button",BS_AUTORADIOBUTTON | WS_GROUP |
                WS_TABSTOP,45,219,39,12
    CONTROL    "ODBC",IDC_ODBC,"Button",BS_AUTORADIOBUTTON | WS_TABSTOP,
                91,219,39,12
    EDITTEXT    ED_IIS_MAX_THREAD_POOL_LIMIT,164,263,34,12,ES_RIGHT |
ES_NUMBER,WS_EX_RTREADING

```

```

EDITTEXT
ED_WEB_SERVICE_BACKLOG_QUEUE_SIZE,164,277,34,12,ES_RIGHT |
  ES_NUMBER,WS_EX_RTLEADING
EDITTEXT ED_IIS_THREAD_TIMEOUT,164,291,34,12,ES_RIGHT |
ES_NUMBER,
  WS_EX_RTLEADING
EDITTEXT ED_IIS_LISTEN_BACKLOG,164,305,34,12,ES_RIGHT |
ES_NUMBER,
  WS_EX_RTLEADING
DEFPUSHBUTTON "OK",IDOK,53,331,50,14
PUSHBUTTON "Cancel",IDCANCEL,119,331,50,14
EDITTEXT IDC_PATH,106,26,91,13,ES_AUTOHSCROLL |
ES_READONLY
LTEXT "Number of Delivery Threads:",IDC_STATIC,35,45,115,12
LTEXT "Max Number of Connections:",IDC_STATIC,35,73,115,12
RTEXT "Version 4.11",IDC_VERSION,120,4,89,9
LTEXT "IIS Max Thread Pool Limit:",IDC_STATIC,36,263,115,12
LTEXT "Web Service Backlog Queue
Size:",IDC_STATIC,36,277,115,
  12
LTEXT "IIS Thread Timeout (seconds):",IDC_STATIC,36,291,115,12
LTEXT "IIS Listen Backlog:",IDC_STATIC,36,307,115,10
GROUPBOX "Database
Interface",IDC_STATIC,35,208,163,27,WS_GROUP
LTEXT "Installation directory:",IDC_STATIC,35,29,71,10
GROUPBOX "Transaction Monitor",IDC_STATIC,33,90,165,37
LTEXT "Server Name:",IDC_STATIC,35,155,56,8
LTEXT "User ID:",IDC_STATIC,35,168,60,8
LTEXT "User Password:",IDC_STATIC,35,181,83,8
LTEXT "Database Name:",IDC_STATIC,35,194,54,8
GROUPBOX "SQL Server Connection
Properties",IDC_STATIC,22,139,187,
  102
GROUPBOX "Web Client Properties",IDC_STATIC,22,15,187,118
GROUPBOX "IIS Settings",IDC_STATIC,22,247,187,79
LTEXT "Max Pending Deliveries:",IDC_STATIC,35,59,115,12
END

IDD_DIALOG2 DIALOGEX 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_DIALOG3 DIALOG 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,7,77,12,SS_SUNKEN
END

IDD_DIALOG4 DIALOG 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, 209
    VERTGUIDE, 35
    VERTGUIDE, 198
    TOPMARGIN, 4
    BOTTOMMARGIN, 345
  END

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

////////////////////////////////////
//
// TPCDLL
//

IDR_TPCDLL      TPCDLL DISCARDABLE
"..\\..\visapi_dll\bin\tpcc.dll"

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

VS_VERSION_INFO VERSIONINFO
FILEVERSION 0,4,20,0
PRODUCTVERSION 0,4,20,0
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, 20, 0\0"
      VALUE "InternalName", "install\0"
      VALUE "LegalCopyright", "Copyright © 1999\0"
      VALUE "OriginalFilename", "install.exe\0"
      VALUE "ProductName", "Microsoft install\0"
      VALUE "ProductVersion", "0, 4, 20, 0\0"
    END
  END
  BLOCK "VarFileInfo"
  BEGIN

```

```

      VALUE "Translation", 0x409, 1200
    END
  END
#endif // !_MAC

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

IDR_LICENSE1    LICENSE DISCARDABLE  "license.txt"

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

IDR_DBLIB_DLL    DBLIB_DLL DISCARDABLE
"..\\..\db_dblib_dll\bin\tpcc_dblib.dll"

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

IDR_ODBC_DLL     ODBC_DLL DISCARDABLE
"..\\..\db_odbc_dll\bin\tpcc_odbc.dll"

////////////////////////////////////
//
// 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
//

IDR_COM_DLL      COM_DLL DISCARDABLE
"..\\..\tm_com_dll\bin\tpcc_com.dll"

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

IDR_COMPS_DLL    COM_PS_DLL DISCARDABLE
"..\\..\tpcc_com_ps\bin\tpcc_com_ps.dll"

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

```

```
IDR_COMALL_DLL COM_ALL_DLL DISCARDABLE
"..\\..\\tpcc_com_all\\bin\\tpcc_com_all.dll"
```

```
////////////////////////////////////
//
// COM_TYPLIB
//
```

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

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

```
#ifndef APSTUDIO_INVOKED
////////////////////////////////////
//
// Generated from the TEXTINCLUDE 3 resource.
//
```

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

### install\_com.cpp

```
/* FILE: INSTALL_COM.CPP
 * Microsoft TPC-C Kit Ver.
4.20.000
 * Copyright Microsoft, 1999
 * All Rights Reserved
 *
 * 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;
    _bstr_t bstrDllPath =
szDllPath;
    _variant_t vTmp, vKey;
    long lActProp, lCount,
    lCountCo, lCountItf, lCountMethod;
    bool bTmp;
```

```
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;
```

```
(if any) // iterate through applications to delete existing "TPC-C" application
```

```
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

hr = pCOMAdminCat->InstallComponent(bstrTemp,
bstrTemp2,
bstrTemp3,
bstrTemp4);
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;

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)
vTmp = (long)30;
hr = pCatalogObjectCo->put_Value(bstrTemp, vTmp);
if (!SUCCEEDED(hr)) goto Error;

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

```

```

collection                // save key to get the MethodsForInterface
                           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";
                               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 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 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

```

// Next default values for new objects

```

//
#ifdef APSTUDIO_INVOKED
#ifndef APSTUDIO_READONLY_SYMBOLS
#define _APS_NEXT_RESOURCE_VALUE 130
#define _APS_NEXT_COMMAND_VALUE 40001
#define _APS_NEXT_CONTROL_VALUE 1031
#define _APS_NEXT_SYMED_VALUE 101
#endif
#endif

```

## isapi\_dll.dsp

```

# Microsoft Developer Studio Project File - Name="isapi_dll" - Package
Owner=<<4>
# Microsoft Developer Studio Generated Build File, Format Version 6.00
# ** DO NOT EDIT **

```

# TARGETTYPE "Win32 (x86) Dynamic-Link Library" 0x0102

CFG=isapi\_dll - Win32 IceCAP

!MESSAGE This is not a valid makefile. To build this project using NMAKE,

!MESSAGE use the Export Makefile command and run

!MESSAGE

!MESSAGE NMAKE /f "isapi\_dll.mak".

!MESSAGE

!MESSAGE You can specify a configuration when running NMAKE

!MESSAGE by defining the macro CFG on the command line. For example:

!MESSAGE

!MESSAGE NMAKE /f "isapi\_dll.mak" CFG="isapi\_dll - Win32 IceCAP"

!MESSAGE

!MESSAGE Possible choices for configuration are:

!MESSAGE

!MESSAGE "isapi\_dll - Win32 Release" (based on "Win32 (x86)

Dynamic-Link Library")

!MESSAGE "isapi\_dll - Win32 Debug" (based on "Win32 (x86) Dynamic-Link

Library")

!MESSAGE "isapi\_dll - Win32 IceCAP" (based on "Win32 (x86)

Dynamic-Link Library")

!MESSAGE

```

# Begin Project
# PROP AllowPerConfigDependencies 0
# PROP Scc_ProjName ""
# PROP Scc_LocalPath ""
CPP=cl.exe
MTL=midl.exe
RSC=rc.exe

```

!IF "\$(CFG)" == "isapi\_dll - 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 ".\obj"
# PROP Ignore_Export_Lib 0
# PROP Target_Dir ""
# ADD BASE CPP /nologo /MT /W3 /GX /O2 /D "WIN32" /D "NDEBUG" /D
" _WINDOWS" /YX /FD /c
# ADD CPP /nologo /MD /W3 /GX /O2 /D "NDEBUG" /D "WIN32" /D
" _WINDOWS" /YX /FD /c
# ADD BASE MTL /nologo /D "NDEBUG" /mktyplib203 /o "NUL" /win32
# ADD MTL /nologo /D "NDEBUG" /mktyplib203 /o "NUL" /win32
# ADD BASE RSC /I 0x409 /d "NDEBUG"
# ADD RSC /I 0x409 /d "NDEBUG"
BSC32=bscmake.exe
# ADD BASE BSC32 /nologo
# ADD BSC32 /nologo
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 odbccp32.lib
# ADD LINK32 ..\common\txnl\lib\release\rttime.lib
..\common\txnl\lib\release\spinlock.lib ..\common\txnl\lib\release\error.lib
..\common\txnl\lib\release\txnl.lib wsock32.lib kernel32.lib user32.lib
gdi32.lib winspool.lib comdlg32.lib advapi32.lib shell32.lib ole32.lib
oleaut32.lib uuid.lib odbccp32.lib /nologo /subsystem:windows /dll
/machine:I386 /nodefaultlib:"LIBCMT" /out:".bin\tpcc.dll"
# SUBTRACT LINK32 /nodefaultlib

```

!ELSEIF "\$(CFG)" == "isapi\_dll - 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 ".\obj"
# PROP Ignore_Export_Lib 0
# PROP Target_Dir ""
# ADD BASE CPP /nologo /MTd /W3 /Gm /GX /ZI /Od /D "WIN32" /D
" _DEBUG" /D " _WINDOWS" /YX /FD /c
# ADD CPP /nologo /MDd /W3 /GX /ZI /Od /D " _DEBUG" /D "WIN32" /D
" _WINDOWS" /FR /YX /FD /c
# ADD BASE MTL /nologo /D " _DEBUG" /mktyplib203 /o "NUL" /win32
# ADD MTL /nologo /D " _DEBUG" /mktyplib203 /o "NUL" /win32
# ADD BASE RSC /I 0x409 /d " _DEBUG"
# ADD RSC /I 0x409 /d " _DEBUG"
BSC32=bscmake.exe
# ADD BASE BSC32 /nologo
# ADD BSC32 /nologo

```

```

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 odbcc32.lib
odbccp32.lib /nologo /subsystem:windows /dll /debug /machine:I386
/pdbtype:sept
# ADD LINK32 ..\common\txnlog\lib\debug\rtetime.lib
..\common\txnlog\lib\debug\spinlock.lib ..\common\txnlog\lib\debug\error.lib
..\common\txnlog\lib\debug\txnlog.lib wsock32.lib kernel32.lib user32.lib
gdi32.lib winspool.lib comdlg32.lib advapi32.lib shell32.lib ole32.lib
oleaut32.lib uuid.lib odbcc32.lib odbccp32.lib /nologo /subsystem:windows /dll
/debug /machine:I386 /nodefaultlib:"LIBCMDT" /out:".bin\tpcc.dll"
/pdbtype:sept
# SUBTRACT LINK32 /profile /pdb:none /nodefaultlib

!ELSEIF "$(CFG)" == "isapi_dll - Win32 IceCAP"

# PROP BASE Use_MFC 0
# PROP BASE Use_Debug_Libraries 1
# PROP BASE Output_Dir "isapi_dll"
# PROP BASE Intermediate_Dir "isapi_dll"
# PROP BASE Ignore_Export_Lib 0
# PROP BASE Target_Dir ""
# PROP Use_MFC 0
# PROP Use_Debug_Libraries 1
# PROP Output_Dir ".\bin"
# PROP Intermediate_Dir ".\obj"
# PROP Ignore_Export_Lib 0
# PROP Target_Dir ""
# ADD BASE CPP /nologo /MDd /W3 /GX /Zi /Od /D "_DEBUG" /D
"WIN32" /D "_WINDOWS" /FR /YX /FD /Gh /c
# ADD CPP /nologo /MD /W3 /GX /Zi /O2 /D "NDEBUG" /D "ICECAP" /D
"WIN32" /D "_WINDOWS" /FR /YX /FD /Gh /c
# ADD BASE MTL /nologo /D "_DEBUG" /mktyplib203 /o "NUL" /win32
# ADD MTL /nologo /D "_DEBUG" /mktyplib203 /o "NUL" /win32
# ADD BASE RSC /I 0x409 /d "_DEBUG"
# ADD RSC /I 0x409 /d "_DEBUG"
BSC32=bscmake.exe
# ADD BASE BSC32 /nologo
# ADD BSC32 /nologo
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 odbcc32.lib
odbccp32.lib /nologo /subsystem:windows /dll /debug /machine:I386
/out:".bin\tpcc.dll" /pdbtype:sept
# SUBTRACT BASE LINK32 /profile /pdb:none
# ADD LINK32 icap.lib ..\common\txnlog\lib\release\rtetime.lib
..\common\txnlog\lib\release\spinlock.lib ..\common\txnlog\lib\release\error.lib
..\common\txnlog\lib\release\txnlog.lib wsock32.lib kernel32.lib user32.lib
gdi32.lib winspool.lib comdlg32.lib advapi32.lib shell32.lib ole32.lib
oleaut32.lib uuid.lib odbcc32.lib odbccp32.lib /nologo /subsystem:windows /dll
/debug /machine:I386 /out:".bin\tpcc.dll" /pdbtype:sept
# SUBTRACT LINK32 /profile /pdb:none /map

!ENDIF

# Begin Target

# Name "isapi_dll - Win32 Release"
# Name "isapi_dll - Win32 Debug"
# Name "isapi_dll - Win32 IceCAP"
# Begin Group "Source"

# PROP Default_Filter "*.cpp, *.def, *.rc"
# Begin Source File

SOURCE=. \src\tpcc.cpp
# End Source File
# Begin Source File

```

```

SOURCE=. \src\tpcc.def
# End Source File
# Begin Source File

SOURCE=. \src\tpcc.rc
# End Source File
# End Group
# Begin Group "Header Files"

# PROP Default_Filter "*.h, *.hpp"
# Begin Source File

SOURCE=. \common\src\error.h
# End Source File
# Begin Source File

SOURCE=. \common\src\ReadRegistry.h
# End Source File
# Begin Source File

SOURCE=. \src\tpcc.h
# End Source File
# Begin Source File

SOURCE=. \db_dblib_dll\src\tpcc_dblib.h
# End Source File
# Begin Source File

SOURCE=. \db_odbc_dll\src\tpcc_odbc.h
# End Source File
# Begin Source File

SOURCE=. \tm_tuxedo_dll\src\tpcc_tux.h
# End Source File
# Begin Source File

SOURCE=. \common\src\trans.h
# End Source File
# Begin Source File

SOURCE=. \common\src\txn_base.h
# End Source File
# End Group
# End Target
# End Project

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 NONINFRINGEMENT.**

9. **NO LIABILITY FOR CONSEQUENTIAL DAMAGES. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, IN NO EVENT SHALL MICROSOFT OR ITS SUPPLIERS BE LIABLE FOR ANY SPECIAL, INCIDENTAL, INDIRECT, OR CONSEQUENTIAL DAMAGES WHATSOEVER (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION, OR ANY OTHER PECUNIARY LOSS) ARISING OUT OF THE USE OF OR INABILITY TO USE THE SOFTWARE PRODUCT, EVEN IF MICROSOFT HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. BECAUSE SOME STATES AND JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF LIABILITY FOR CONSEQUENTIAL OR INCIDENTAL DAMAGES, THE ABOVE LIMITATION MAY NOT APPLY TO YOU.**

10. **LIMITATION OF LIABILITY. MICROSOFT'S ENTIRE LIABILITY AND YOUR EXCLUSIVE REMEDY UNDER THIS EULA SHALL NOT EXCEED FIVE DOLLARS (US\$5.00).**

11. **MISCELLANEOUS**

This EULA is governed by the laws of the State of Washington, U.S.A. Should you have any questions concerning this EULA, or if you desire to contact Microsoft for any reason, please contact the Microsoft subsidiary serving your country, or write: Microsoft Sales Information Center/One Microsoft Way/Redmond, WA 98052-6399.

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

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

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

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

**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,
    VARIANT* txn_out) {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          Payment;
            DELIVERY_DATA        Delivery;
            STOCK_LEVEL_DATA     StockLevel;
            ORDER_STATUS_DATA    OrderStatus;
        };
    };

// 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)
        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(          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(      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(      VARIANT txn_in,
    VARIANT* txn_out) {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(      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;}
};

////////////////////////////////////
// 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(VARIANT txn_in, VARIANT*
    txn_out) {return E_NOTIMPL;}
    HRESULT __stdcall OrderStatus(      VARIANT txn_in,
    VARIANT* txn_out) {return E_NOTIMPL;}
};

```

### Null-txns.sql

```

-- File:  NULL-TXNS.SQL
--        Microsoft TPC-C Benchmark Kit Ver. 4.41
--        Copyright Microsoft, 2001
--
-- Purpose:  This script will create stored procs which accept the same paramete-
--            rs and
--            return correctly formed results sets to match the standard TPC-C stored
--            procs.  Of course, the advantage is that these stored procs place almost
--            no load on SQL Server and do not require a database.
--
--            The purpose of these stored procs is to size and test the web
--            client without
--            the need of a fully scaled database.
--
drop proc tpcc_delivery
drop proc tpcc_neworder
drop proc tpcc_orderstatus
drop proc tpcc_payment
drop proc tpcc_stocklevel
drop proc tpcc_version
drop table order_line_null
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

declare @delaytime varchar(30)

-- uniform random delay of 0 - 1 second; avg = 0.50
select @delaytime = '00:00:0' + cast((cast((rand()*1.00) as decimal(4,3)) as
char(5)))
waitfor delay @delaytime

select 3001, 3001, 3001, 3001, 3001, 3001, 3001, 3001, 3001, 3001, 3001, 3001

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,
        @i_id2     int = 0,
        @i_id3     int = 0,
        @i_id4     int = 0,
        @i_id5     int = 0,
        @i_id6     int = 0,
        @i_id7     int = 0,
        @i_id8     int = 0,
        @i_id9     int = 0,
        @i_id10    int = 0,
        @i_id11    int = 0,
        @i_id12    int = 0,
        @i_id13    int = 0,
        @i_id14    int = 0,
        @i_id15    int = 0,

        @s_w_id1  int = 0, @ol_qty1  smallint = 0,
        @s_w_id2  int = 0, @ol_qty2  smallint = 0,
        @s_w_id3  int = 0, @ol_qty3  smallint = 0,
        @s_w_id4  int = 0, @ol_qty4  smallint = 0,
        @s_w_id5  int = 0, @ol_qty5  smallint = 0,
        @s_w_id6  int = 0, @ol_qty6  smallint = 0,
        @s_w_id7  int = 0, @ol_qty7  smallint = 0,
        @s_w_id8  int = 0, @ol_qty8  smallint = 0,
        @s_w_id9  int = 0, @ol_qty9  smallint = 0,
        @s_w_id10 int = 0, @ol_qty10 smallint = 0,
        @s_w_id11 int = 0, @ol_qty11 smallint = 0,
        @s_w_id12 int = 0, @ol_qty12 smallint = 0,
        @s_w_id13 int = 0, @ol_qty13 smallint = 0,
        @s_w_id14 int = 0, @ol_qty14 smallint = 0,
        @s_w_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),
        @o_entry_d datetime,
        @li_no int,
        @o_id int,
        @commit_flag tinyint,
        @li_id int,
        @li_qty smallint

declare @delaytime varchar(30)

begin
-- uniform random delay of 0 - 0.6 second; avg = 0.3
select @delaytime = '00:00:0' + cast(cast((rand()*0.60) as decimal(4,3)) as
char(5))
waitfor delay @delaytime

-- process orderlines

select @commit_flag = 1, @li_no = 0

while (@li_no < @o_ol_cnt)
begin

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

select @li_no = @li_no + 1
select @i_price = 23.45, @li_qty = @li_no

if (@li_id = 999999)
begin
select ",0",0,0
select @commit_flag = 0
end

else
begin
select 'Item Name blah',17,'G', @i_price, @i_price *
@li_qty

end

end

-- return order data to client

select @w_tax = 0.1234,
        @d_tax = 0.0987,

```

```

        @o_id = 3001,
        @c_last = 'BAROUGHTABLE',
        @c_discount = 0.2198,
        @c_credit = 'GC',
        @o_entry_d = getdate()

select @w_tax,
        @d_tax,
        @o_id,
        @c_last,
        @c_discount,
        @c_credit,
        @o_entry_d,
        @commit_flag

end

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,
        @ol_cnt smallint

declare @delaytime varchar(30)

-- uniform random delay of 0 - 0.2 second; avg = 0.1
select @delaytime = '00:00:0' + cast(cast((rand()*0.20) as decimal(4,3)) as
char(5))
waitfor delay @delaytime

select
        @c_id = 113,
        @c_balance = -10.00,
        @c_first = '8YCodgytqCj8',
        @c_middle = 'OE',
        @c_last = 'OUGHTOUGHTABLE',
        @o_id = 3456,
        @o_entry_d = getdate(),
        @o_carrier_id = 1

select @ol_cnt = (rand() * 11) + 5
SET ROWCOUNT @ol_cnt

select
        ol_supply_w_id,
        ol_i_id,
        ol_quantity,
        ol_amount,
        ol_delivery_d
from order_line_null

select @c_id,
        @c_last,
        @c_first,
        @c_middle,

```

```

        @o_entry_d,
        @o_carrier_id,
        @c_balance,
        @o_id

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

declare @delaytime varchar(30)

-- uniform random delay of 0 - 0.3 second; avg = 0.15
select @delaytime = '00:00:0' + cast(cast((rand()*0.30) as decimal(4,3)) as
char(5))
waitfor delay @delaytime

select @screen_data = ""

-- get customer info and update balances

select
        @d_street_1 = 'rqSHHakqyV',

```

```

        @d_street_2 = 'zZ98nW3BR2s',
        @d_city    = 'ArNr4GNFV9',
        @d_state   = 'aV',
        @d_zip     = '453511111'

-- get warehouse data and update year-to-date

select
        @w_street_1 = 'rqSHHakqyV',
        @w_street_2 = 'zZ98nW3BR2s',
        @w_city     = 'ArNr4GNFV9',
        @w_state    = 'aV',
        @w_zip      = '453511111'

select
        @c_id       = 123,
        @c_balance  = -10000.00,
        @c_first    = 'KmR03Xureb',
        @c_middle   = 'OE',
        @c_last     = 'BAROUGHTBAR',
        @c_street_1 = 'QpGdOHjv8mR9vNI8V',
        @c_street_2 = 'dzKcCObBqbC3yu',
        @c_city     = 'zAKZXdC037FQxqj',
        @c_state    = 'QA',
        @c_zip      = '700311111',
        @c_phone    = '2967264064528555',
        @c_credit   = 'GC',
        @c_credit_lim = 50000.00,
        @c_discount = 0.3069,
        @c_since    = getdate(),
        @datetime   = getdate()

-- 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

create proc tpcc_stocklevel @w_id      int,

```



```

                @d_id          tinyint,
                @threshhold   smallint
as
declare @delaytime varchar(30)

-- uniform random delay of 0 - 3.6 second; avg = 1.8
select @delaytime = '00:00:0' + cast(cast((rand()*3.60) as decimal(4,3)) as
char(5))
waitfor delay @delaytime

select 49

GO

create proc tpcc_version
as
declare @version char(8)

begin
    select @version = '4.10.000'
    select @version as 'Version'
end

GO

CREATE TABLE order_line_null (
    [ol_i_id] [int] NOT NULL ,
    [ol_supply_w_id] [int] NOT NULL ,
    [ol_delivery_d] [datetime] NOT NULL ,
    [ol_quantity] [smallint] NOT NULL ,
    [ol_amount] [numeric](6, 2) NOT NULL
) ON [PRIMARY]
GO

insert into order_line_null values ( 101, 1, getdate(), 1, 123.45 )
insert into order_line_null values ( 102, 1, getdate(), 2, 123.45 )
insert into order_line_null values ( 103, 1, getdate(), 3, 123.45 )
insert into order_line_null values ( 104, 1, getdate(), 4, 123.45 )
insert into order_line_null values ( 105, 1, getdate(), 5, 123.45 )
insert into order_line_null values ( 106, 1, getdate(), 1, 123.45 )
insert into order_line_null values ( 107, 1, getdate(), 2, 123.45 )
insert into order_line_null values ( 108, 1, getdate(), 3, 123.45 )
insert into order_line_null values ( 109, 1, getdate(), 4, 123.45 )
insert into order_line_null values ( 110, 1, getdate(), 5, 123.45 )
insert into order_line_null values ( 111, 1, getdate(), 1, 123.45 )
insert into order_line_null values ( 112, 1, getdate(), 2, 123.45 )
insert into order_line_null values ( 113, 1, getdate(), 3, 123.45 )
insert into order_line_null values ( 114, 1, getdate(), 4, 123.45 )
insert into order_line_null values ( 115, 1, getdate(), 5, 123.45 )

```

## ReadRegistry.cpp

```

/*      FILE:          READREGISTRY.CPP
*
*      4.20.000
*
*      All Rights Reserved
*
*
*      Copyright Microsoft, 1999
*
*      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
*
*      Microsoft TPC-C Kit Ver.
4.20.000
*
*      Copyright Microsoft, 1999
*
*      All Rights Reserved
*
*
*      not audited
*
*      PURPOSE:       Header for registry related code.
*
*      Change history:
*
*      4.20.000 - first version
*/

enum DBPROTOCOL { Unspecified, ODBC, DBLIB };
const char *szDBNames[] = { "Unspecified", "ODBC", "DBLIB" };

enum TXNMON { None, TUXEDO, ENCINA, COM };
const char *szTxnMonNames[] = { "NONE", "TUXEDO", "ENCINA", "COM"
};

```

```

//This structure defines the data necessary to keep distinct for each terminal or
client connection.
typedef struct _TPCCREGISTRYDATA
{
    enum DBPROTOCOL eDB_Protocol;
    enum TXNMON eTxnMon;
    BOOL bCOM_SinglePool;
    DWORD dwMaxConnections;
    DWORD dwMaxPendingDeliveries;
    DWORD dwNumberOfDeliveryThreads;
    char szPath[128];
    char szDbServer[32];
    char szDbName[32];
    char szDbUser[32];
    char szDbPassword[32];
} TPCCREGISTRYDATA, *PTPCCREGISTRYDATA;

BOOL ReadTPCCRegistrySettings( TPCCREGISTRYDATA *pReg );

```

## restore.vbs

```

'-----
'--- FILE:    RESTORE.VBS
'---         Microsoft TPC-C Kit Ver. 4.41
'---         Copyright Microsoft, 2001
'---         All Rights Reserved
'---
'--- PURPOSE: This module executes a database restore
'---
'-----
'--- open an windows scripting object
'-----
set WshShell = CreateObject("WScript.Shell")
'-----
'--- display a banner message
'-----
wScript.Echo
"*****"
*****"
wScript.Echo "*"
wScript.Echo "*" Microsoft TPC-C V3 Benchmark Kit Ver. 4.41 - Restore
*"
wScript.Echo "*"
wScript.Echo
"*****"
*****"
'-----
'--- define function to check for any error messages
'-----
Function CheckSQLOutput(SQL_Out)
    ErrorFlag = 0
    Set SQL_fso = CreateObject("Scripting.FileSystemObject")
    Set SQL_Out_File = SQL_fso.OpenTextFile(SQL_Out,1)
    Do While SQL_Out_File.AtEndOfStream <> True
        SQL_Line = SQL_Out_File.ReadLine
        'first check to see if the output contains a message about
the login password
        If InStr(SQL_Line, "Login failed") Then
            'display the messages and get out of here
            ErrorFlag = 1
            wScript.Echo "The login for userid 'sa' failed."
            wScript.Echo "Please restart SETUP with the
correct password."
        Else

```

```

                If InStr(SQL_Line, "Msg") Then
'find out where the "Msg" indicator is in the line
LocMsg = InStr(SQL_Line, "Msg")
'find out where the comma is after the error code
LocComma = InStr(SQL_Line, ",")
'now isolate the error code
                ErrorCode = Mid(SQL_Line, (LocMsg + 4), (LocComma -
(LocMsg + 4)))
                Select Case ErrorCode
                Case "15069"
                    ErrorFlag = 1
                    wScript.Echo "One or more users are using the
database."
                    wScript.Echo "The requested operation cannot be
completed."
                Case "3201"
                    ErrorFlag = 1
                    wScript.Echo "Cannot open backup device."
                    wScript.Echo "Device error or device off-line."
                    wScript.Echo "SQL Server Error 3201."
                    wScript.Echo "See the SQL Server error log for more
details."
                End Select
                End If
            End If
        Loop
            SQL_Out_File.Close
            CheckSQLOutput = ErrorFlag
        End Function
'-----
'--- end function
'-----
'--- open a file system object
'-----
Set fs = CreateObject("Scripting.FileSystemObject")
'-----
'--- grab the current directory value
'-----
SetupDirectory = WshShell.CurrentDirectory & "\\"
'-----
'--- now calculate the other directories
'-----
ScriptDirectory = SetupDirectory & "SCRIPTS\"
LogDirectory = SetupDirectory & "LOGS\"
'-----
'--- check to see if the user passed in the server name and sa password
'-----
Set objArgs = wScript.Arguments
Select Case objArgs.Length
    Case 0
        '-----
        '--- the user did not pass us anything
        '--- grab the Computer Name from Windows
        '-----
        ServerName =
WshShell.ExpandEnvironmentStrings("%COMPUTERNAME%")
        '-----
        '--- prompt the user to confirm the server name
        '-----
        ServerName = InputBox("Enter your server
name",Test_Name,ServerName)
        Do While ServerName = ""
            rc = MsgBox ("You must enter a valid server
name.",21)
            If rc = 2 Then
                wScript.Echo ""

```

```

                wScript.Echo "TPC-C Setup
cancelled by user."
                wScript.Quit
            End If
            ServerName =
WshShell.ExpandEnvironmentStrings("%COMPUTERNAME%")
            ServerName = InputBox("Enter your server
name", "Database Server Name", ServerName)
        Loop
        '-----
        '--- prompt the user for the sa password
        '-----
        saPassword = InputBox("Enter the 'sa'
password",Test_Name)
        Case 1
            '-----
            '--- the user passed 1 argument, so assume it is the server
name
            '-----
            ServerName = objArgs(0)
            '-----
            '--- prompt the user for the sa password
            '-----
            saPassword = InputBox("Enter the 'sa'
password",Test_Name)
        Case 2
            '-----
            '--- the user passed 2 arguments, so try to use them
            '-----
            ServerName = objArgs(0)
            saPassword = objArgs(1)
        End Select
        '-----
        '--- now that we have all the variables filled in, let's get to work
        '-----
        If fs.FileExists(LogDirectory & "restore.log") Then
            fs.DeleteFile LogDirectory & "restore.log"
        End If
        Wscript.Echo "Restoring database from backup..."
        Set oExec = WshShell.Exec("osql -Usa -P" & saPassword & " -S" &
ServerName & " -e -i" & ScriptDirectory & NumberWarehouses &
".war\database\restore.sql -o" & LogDirectory & "restore.log")
        Do While oExec.Status = 0
            wScript.Sleep 100
        Loop
        rc = CheckSQLOutput(LogDirectory & "restore.log")
        If rc <> 0 Then
            wScript.Quit
        End If
        wScript.Echo ""
        wScript.Echo
        "*****"
        "*****"
        wScript.Echo "*"
        wScript.Echo "*" Microsoft TPC-C Benchmark Kit Ver. 4.41
        "*"
        wScript.Echo "*"
        wScript.Echo "*" Database restore complete.
        wScript.Echo "*"
        wScript.Echo
        "*****"
        "*****"

```

**runsqlcfg.vbs**

```

-----
'--- FILE:   RESTORE.VBS
'---       Microsoft TPC-C Kit Ver. 4.41
'---       Copyright Microsoft, 2001
'---       All Rights Reserved
'---
'--- PURPOSE: This module executes a database restore
'---
-----
'--- open an windows scripting object
'---
set WshShell = CreateObject("WScript.Shell")
'---
'--- display a banner message
'---
wScript.Echo
*****
wScript.Echo "*"
wScript.Echo "* Microsoft TPC-C V3 Benchmark Kit Ver. 4.41 - Configure
SQL Server      *"
wScript.Echo "*"
wScript.Echo
*****
'---
'--- define function to check for any error messages
'---
Function CheckSQLOutput(SQL_Out)
    ErrorFlag = 0
    Set SQL_fso = CreateObject("Scripting.FileSystemObject")
    Set SQL_Out_File = SQL_fso.OpenTextFile(SQL_Out,1)
    Do While SQL_Out_File.AtEndOfStream <> True
        SQL_Line = SQL_Out_File.ReadLine
        'first check to see if the output contains a message about
the login password
        If InStr(SQL_Line, "Login failed") Then
            'display the messages and get out of here
            ErrorFlag = 1
            Wscript.Echo "The login for userid 'sa'
failed."
            Wscript.Echo "Please restart SETUP with the
correct password."
        End If
    Loop
    SQL_Out_File.Close
    CheckSQLOutput = ErrorFlag
End Function
'---
'--- end function
'---
'--- open a file system object
'---
Set fs = CreateObject("Scripting.FileSystemObject")
'---
'--- grab the current directory value
'---
SetupDirectory = WshShell.CurrentDirectory & "\"
'---
'--- now calculate the other directories
'---
ScriptDirectory = SetupDirectory & "SCRIPTS\"
LogDirectory = SetupDirectory & "LOGS\"
'---
'--- check to see if the user passed in the server name and sa password
'---

```

```

Set objArgs = wScript.Arguments
Select Case objArgs.Length
    Case 0
        '-----
        '--- the user did not pass us anything
        '--- grab the Computer Name from Windows
        '-----
        ServerName =
WshShell.ExpandEnvironmentStrings("%COMPUTERNAME%")
        '-----
        '--- prompt the user to confirm the server name
        '-----
        ServerName = InputBox("Enter your server
name",Test_Name,ServerName)
        Do While ServerName = ""
            rc = MsgBox ("You must enter a valid server
name.",21)
            If rc = 2 Then
                wScript.Echo ""
                wScript.Echo "TPC-C Setup
cancelled by user."
                wScript.Quit
            End If
            ServerName =
WshShell.ExpandEnvironmentStrings("%COMPUTERNAME%")
            ServerName = InputBox("Enter your server
name","Database Server Name",ServerName)
        Loop
        '-----
        '--- prompt the user for the sa password
        '-----
        saPassword = InputBox("Enter the 'sa'
password",Test_Name)
        Case 1
            '-----
            '--- the user passed 1 argument, so assume it is the server
name
            '-----
            ServerName = objArgs(0)
            '-----
            '--- prompt the user for the sa password
            '-----
            saPassword = InputBox("Enter the 'sa'
password",Test_Name)
        Case 2
            '-----
            '--- the user passed 2 arguments, so try to use them
            '-----
            ServerName = objArgs(0)
            saPassword = objArgs(1)
        End Select
'-----
'--- now that we have all the variables filled in, let's get to work
'-----
If fs.FileExists(LogDirectory & "runsqlcfg.log") Then
    fs.DeleteFile LogDirectory & "runsqlcfg.log"
End If
'-----
'--- configure SQL Server
'-----
wScript.Echo " "
wScript.Echo "Configuring Microsoft SQL Server installation..."
Set oExec = WshShell.Exec("osql -Usa -P" & saPassword & " -S" &
ServerName & " -e -i" & ScriptDirectory & "utility\runsqlcfg.sql -o" &
LogDirectory & "runsqlcfg.log")
Do While oExec.Status = 0
    WScript.Sleep 100
Loop

```

```

rc = CheckSQLOutput(LogDirectory & "runsqlcfg.log")
If rc <> 0 Then
    Wscript.Quit
End If
wScript.Echo " "
wScript.Echo "SQL Server Configuration Complete."
'-----
'--- shutdown SQL Server
'-----
wScript.Echo " "
wScript.Echo "Shutting down SQL Server..."
Set oExec = WshShell.Exec("osql -Usa -P" & saPassword & " -S" &
ServerName & " -e -i" & ScriptDirectory & "utility\sqlshutdown.sql")
wScript.Echo " "
wScript.Echo "Waiting for SQL Server to shutdown..."
Set oExec = WshShell.Exec("..\tools\sleep\sleep.exe 20")
Do While oExec.Status = 0
    WScript.Sleep 100
Loop
'-----
'--- Restarting SQL Server
'-----
wScript.Echo " "
wScript.Echo "Restarting SQL Server..."
wScript.Echo " "
CMD_String = "start sqlservr.exe -c -t3502"
oExec = WshShell.Run(CMD_String, 2, false)
wScript.Echo ""
wScript.Echo
*****
wScript.Echo "*"
wScript.Echo "* Microsoft TPC-C Benchmark Kit Ver. 4.41"
*"
wScript.Echo "*"
wScript.Echo "* SQL Server configuration complete."
wScript.Echo "*"
wScript.Echo
*****
*****

```

### rtetime.h

```

/* FILE:
rtetime.h : header file
* Copyright 1997 Microsoft Corp., All rights reserved.
*
* Source code licensed to Tandem Computers for Internal
* use only. Redistribution of source or object files or
* any derivative works is prohibited. By agreement, this
* notice may not be removed.
*
* Authors: Charles Levine, Philip Durr
* Microsoft Corp.
*/

//FILE: RTETIME.H

#define MAX_JULIAN_TIME
0x7FFFFFFF
#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 );
}

```

### setup.vbs

```

'-----
'--- FILE: SETUP.VBS
'--- Microsoft TPC-C Kit Ver. 4.41
'--- Copyright Microsoft, 2001
'--- All Rights Reserved
'---
'--- PURPOSE: This module performs the tasks to create and populate a
TPC-C database
'-----
'-----
'--- open an windows scripting object
'-----
set WshShell = CreateObject("WScript.Shell")
'-----
'--- before we go any further, make sure that
'--- we are running Windows Scripting Host 5.6
'--- or higher
'-----
If WScript.Version < 5.6 Then
    WScript.Echo
    "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
    WScript.Echo "!!"
    WScript.Echo "!! You do not have the proper version of the
Windows Scripting Host !!!"
    WScript.Echo "!! installed. Please install the latest Windows
Scripting Host from !!!"
    WScript.Echo "!! ..\tools\wsh\scripten.exe and restart setup."
    "!!"
    WScript.Echo "!!"
    WScript.Echo
    "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
    WScript.Quit
End If
'-----
'--- display banner message
'-----
WScript.Echo
*****
*****
WScript.Echo "*"
WScript.Echo "* Microsoft TPC-C Benchmark Kit Ver. 4.41 - Setup"
*"
WScript.Echo "*"
WScript.Echo
*****
*****
'-----
'--- define function to check for any error messages
'-----
Function CheckSQLOutput(SQL_Out)
    ErrorFlag = 0
    Set SQL_fso = CreateObject("Scripting.FileSystemObject")

```

<pre> If SQL_fso.FileExists(SQL_Out) Then     Set SQL_Out_File = SQL_fso.OpenTextFile(SQL_Out,1)     Do While SQL_Out_File.AtEndOfStream &lt;&gt; True         SQL_Line = SQL_Out_File.ReadLine         'first check to see if the output contains a message about the login password         If InStr(SQL_Line, "Login failed") Then             'display the messages and get out             ErrorFlag = 1             wScript.Echo "The login for userid 'sa' failed."             wScript.Echo "Please restart SETUP with the correct password."         Else             If InStr(SQL_Line, "Msg") Then                 'find out where the                 LocMsg =                 'find out where the                 LocComma =                 'now isolate the error                 ErrorCode =                 Mid(SQL_Line, (LocMsg + 4), (LocComma - (LocMsg + 4)))                 Select Case ErrorCode                     Case "170"                         ErrorFlag = 1                         wScript.Echo "!!"                         wScript.Echo "Syntax Error."                         wScript.Echo "SQL Server Error 170."                         wScript.Echo "Check CREATEDB.SQL."                         wScript.Echo "!!"                     Case "1801"                         ErrorFlag = 1                         wScript.Echo "!!"                         wScript.Echo "Database 'tpcc' already exists."                         wScript.Echo "SQL Server Error 1801."                         wScript.Echo "Check CREATEDB.SQL."                         wScript.Echo "!!"                     Case "1802"                         ErrorFlag = 1                         wScript.Echo "!!"                         wScript.Echo "CREATE DATABASE failed."                         wScript.Echo "SQL Server Error 1802."                         wScript.Echo "Check CREATEDB.SQL." </pre>	<pre> wScript.Echo "!!" ErrorFlag = 1 wScript.Echo "!!" wScript.Echo "CREATE INDEX failed." wScript.Echo "SQL Server Error 1921." wScript.Echo "Check " &amp; SQL_Out &amp; "." wScript.Echo "!!"                 ErrorFlag = 1                 wScript.Echo                 "!!" wScript.Echo "BACKUP DATABASE is terminating abnormally." wScript.Echo "SQL Server Error 3013." wScript.Echo "Check the SQL Server error log for more details." wScript.Echo "!!"                 ErrorFlag = 1                 wScript.Echo                 "!!" wScript.Echo "Cannot open backup device." wScript.Echo "Device error or device off-line." wScript.Echo "SQL Server Error 3201." wScript.Echo "See the SQL Server error log for more details." wScript.Echo "!!"                 ErrorFlag = 1                 wScript.Echo                 "!!" wScript.Echo "Device Activation Error." wScript.Echo "SQL Server Error 5105." wScript.Echo "Check CREATEDB.SQL." wScript.Echo "!!" ErrorFlag = 1 wScript.Echo "!!" wScript.Echo "Cannot create one or more files because it already exists." wScript.Echo "SQL Server Error 5170." wScript.Echo "Check CREATEDB.SQL." wScript.Echo "!!"                 Case                 "15010","15012" </pre>	<pre> Case "1921" Case "3013" Case "3201" Case "5105" Case "5170" Case </pre>
--	---	---

```

ErrorFlag = 0
"15069"
ErrorFlag = 1
wScript.Echo "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
wScript.Echo "One or more users are using the database."
wScript.Echo "The requested operation cannot be completed."
wScript.Echo "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
Case Else
ErrorFlag = 1
wScript.Echo "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
wScript.Echo "An error occurred."
wScript.Echo "SQL Server Error Code: " & ErrorCode & "."
wScript.Echo "Check " & SQL_Out & " for more information."
wScript.Echo "!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!"
End Select
End If
Loop
SQL_Out_File.Close
End If
CheckSQLOutput = ErrorFlag
End Function
'-----
'--- end function
'-----
'--- define function to prompt for user input if necessary
'-----
Function GetUserInput(UserInput)
Select Case UserInput
Case "ServerName"
'--- pre-fill the prompt with the machine name
TempServerName =
WshShell.ExpandEnvironmentStrings("%COMPUTERNAME%")
'--- prompt the use for the setup particulars
TempResponse = InputBox("Enter your server
name","TPC-C Setup",TempServerName)
Do While TempResponse = ""
rc = MsgBox ("You must enter a
valid server name.",21)
If rc = 2 Then
wScript.Echo ""
wScript.Echo "TPC-C
Setup cancelled by user."
wScript.Quit
End If
TempResponse =
WshShell.ExpandEnvironmentStrings("%COMPUTERNAME%")
TempResponse = InputBox("Enter
your server name","TPC-C Setup",TempServerName)
Loop
Case "saPassword"
TempResponse = InputBox("Enter the 'sa'
password")
Case "NumberWarehouses"

```

```

TempResponse = InputBox("Enter the number
of warehouses to build","TPC-C Setup")
Do While TempResponse = ""
rc = MsgBox ("You must enter a
value for Number of Warehouses.",21)
If rc = 2 Then
wScript.Echo ""
wScript.Echo "TPC-C
Setup cancelled by user."
wScript.Quit
End If
TempResponse = InputBox("Enter
the number of warehouses to build","TPC-C Setup")
Loop
Case "BuildOption"
TempResponse = InputBox("Build Option" &
Chr(13) &
"(full,builddb,objects,objectsfull,bulkload,bulkloadfull,backup)","TPC-C
Setup","full")
Flag = 0
Do While Flag = 0
Select Case TempResponse
Case
"full","Full","FULL"
TempResponse = "full"
Flag = 1
Case
"builddb","BuildDB","Builddb","BUILDDB"
TempResponse = "builddb"
Flag = 1
Case
"objects","Objects","OBJECTS"
TempResponse = "objects"
Flag = 1
Case
"objectsfull","ObjectsFull","Objectsfull","OBJECTSFULL"
TempResponse = "objectsfull"
Flag = 1
Case
"bulkload","BulkLoad","Bulkload","BULKLOAD"
TempResponse = "bulkload"
Flag = 1
Case
"bulkloadfull","BulkLoadFull","Bulkloadfull","BULKLOADFULL"
TempResponse = "bulkloadfull"
Flag = 1
Case
"backup","Backup","BACKUP"
TempResponse = "backup"
Flag = 1
Case Else
rc =
If rc = 2
Then
wScript.Echo ""
wScript.Echo "TPC-C Setup cancelled by user."

```

```

wScript.Quit

End If
Flag = 0

TempResponse = InputBox("Build Option" & Chr(13) &
"(full,builddb,objects,objectsfull,bulkload,bulkloadfull,backup)",,"full")

End Select
Loop
Case "DatabaseType"
TempResponse = InputBox("Database Type"
& Chr(13) & "(normal or scale_down)","TPC-C Setup","normal")
'--- set flag
Flag = 0
Do While Flag = 0
Select Case TempResponse
Case
"normal","Normal","NORMAL"

TempResponse = "0"

Flag = 1
Case
"scale_down","Scale_Down","Scale_down","SCALE_DOWN"

TempResponse = "1"

Flag = 1
Case Else
rc =
If rc = 2

Then

wScript.Echo ""

wScript.Echo "TPC-C Setup cancelled by user."

wScript.Quit

End If
Flag = 0

TempResponse = InputBox("Database Type" & Chr(13) & "(normal or
scale_down)",,"normal")

End Select
Loop
End Select
GetUserInput = TempResponse
End Function
'-----
'--- end function
'-----
'--- Initialize an array of the TPC-C table names
'-----
Dim TableArray(8)
TableArray(0) = "warehouse"
TableArray(1) = "district"
TableArray(2) = "customer"
TableArray(3) = "history"
TableArray(4) = "new_order"
TableArray(5) = "orders"
TableArray(6) = "order_line"
TableArray(7) = "item"
TableArray(8) = "stock"
'-----
'--- Initialize an array of the TPC-C build log file names
'-----
Dim LogFileArray(21)

```

```

LogFileArray(0) = "version.log"
LogFileArray(1) = "removedb.log"
LogFileArray(2) = "createdb.log"
LogFileArray(3) = "tables.log"
LogFileArray(4) = "dbopt1.log"
LogFileArray(5) = "idxordcl.log"
LogFileArray(6) = "idxitmcl.log"
LogFileArray(7) = "idxwarcl.log"
LogFileArray(8) = "idxcuscl.log"
LogFileArray(9) = "idxnodcl.log"
LogFileArray(10) = "idxdiscl.log"
LogFileArray(11) = "idxstkcl.log"
LogFileArray(12) = "idxodlcl.log"
LogFileArray(13) = "idxcusnc.log"
LogFileArray(14) = "idxhiscl.log"
LogFileArray(15) = "idxordnc.log"
LogFileArray(16) = "bulkload.log"
LogFileArray(17) = "dbopt2.log"
LogFileArray(18) = "nurand_load.log"
LogFileArray(19) = "backupdev.log"
LogFileArray(20) = "backupdev.log"
LogFileArray(21) = "verifyload.log"
'-----
'--- open a file system object
'-----
Set fs = CreateObject("Scripting.FileSystemObject")
'-----
'--- grab the current directory value
'-----
SetupDirectory = WshShell.CurrentDirectory & "\"
'SetupDirectory = "C:\MSTPCC.441\"
'-----
'--- now calculate the other directories
'-----
ACIDDirectory = LEFT(SetupDirectory,(LEN(SetupDirectory)-6))
ScriptDirectory = SetupDirectory & "SCRIPTS\"
LogDirectory = SetupDirectory & "LOGS\"
'-----
'--- now determine if the user passed us any parameters.
'--- the order should be ServerName, sa Password, Number of Warehouses,
'--- Build Option, and Database Type
'-----
Set objArgs = wScript.Arguments
Select Case objArgs.Length
Case 0
'-----
'--- get the server name
'-----
ServerName = GetUserInput("ServerName")
'-----
'--- get the sa password
'-----
saPassword = GetUserInput("saPassword")
'-----
'--- get the number of warehouses
'-----
NumberWarehouses =
GetUserInput("NumberWarehouses")
'-----
'--- get the build option
'-----
BuildOption = GetUserInput("BuildOption")
'-----
'--- get the database type
'-----
DatabaseType = GetUserInput("DatabaseType")
Case 1
'-----

```



```

'--- assume that the server name was passed correctly
'-----
'--- store the server name
'-----
ServerName = objArgs(0)
'-----
'--- get the sa password
'-----
saPassword = GetUserInput("saPassword")
'-----
'--- get the number of warehouses
'-----
NumberWarehouses =
GetUserInput("NumberWarehouses")
'-----
'--- get the build option
'-----
BuildOption = GetUserInput("BuildOption")
'-----
'--- get the database type
'-----
DatabaseType = GetUserInput("DatabaseType")
If DatabaseType = "scale_down" or DatabaseType =
"Scale_Down" or DatabaseType = "Scale_down" Then
    DatabaseType = 1
Else
    DatabaseType = 0
End If
Case 2
'-----
'--- assume that the server name and sa password was
passed correctly
'-----
'--- store the server name
'-----
ServerName = objArgs(0)
'-----
'--- store the sa password
'-----
saPassword = objArgs(1)
'-----
'--- get the number of warehouses
'-----
NumberWarehouses =
GetUserInput("NumberWarehouses")
'-----
'--- get the build option
'-----
BuildOption = GetUserInput("BuildOption")
'-----
'--- get the database type
'-----
DatabaseType = GetUserInput("DatabaseType")
If DatabaseType = "scale_down" or DatabaseType =
"Scale_Down" or DatabaseType = "Scale_down" Then
    DatabaseType = 1
Else
    DatabaseType = 0
End If
Case 3
'-----
'--- assume that the server name,sa password, and number
of warehouses was passed correctly
'-----

```

```

'-----
'--- store the server name
'-----
ServerName = objArgs(0)
'-----
'--- store the sa password
'-----
saPassword = objArgs(1)
'-----
'--- store the number of warehouses
'-----
NumberWarehouses = objArgs(2)
'-----
'--- get the build option
'-----
BuildOption = GetUserInput("BuildOption")
'-----
'--- get the database type
'-----
DatabaseType = GetUserInput("DatabaseType")
If DatabaseType = "scale_down" or DatabaseType =
"Scale_Down" or DatabaseType = "Scale_down" Then
    DatabaseType = 1
Else
    DatabaseType = 0
End If
Case 4
'-----
'--- assume that the server name,sa password,number of
warehouses, and build option was passed correctly
'-----
'-----
'-----
'--- store the server name
'-----
ServerName = objArgs(0)
'-----
'--- store the sa password
'-----
saPassword = objArgs(1)
'-----
'--- store the number of warehouses
'-----
NumberWarehouses = objArgs(2)
'-----
'--- store the build option
'-----
BuildOption = objArgs(3)
'-----
'--- get the database type
'-----
DatabaseType = GetUserInput("DatabaseType")
If DatabaseType = "scale_down" or DatabaseType =
"Scale_Down" or DatabaseType = "Scale_down" Then
    DatabaseType = 1
Else
    DatabaseType = 0
End If
Case 5
'-----
'--- assume all the parameters were passed in correctly
'-----
'--- store the server name
'-----

```

```

ServerName = objArgs(0)
'-----
'--- store the sa password
'-----
saPassword = objArgs(1)
'-----
'--- store the number of warehouses
'-----
NumberWarehouses = objArgs(2)
'-----
'--- store the build option
'-----
BuildOption = objArgs(3)
'-----
'--- get the database type
'-----
DatabaseType = objArgs(4)
If DatabaseType = "scale_down" or DatabaseType =
"Scale_Down" or DatabaseType = "Scale_down" Then
    DatabaseType = 1
Else
    DatabaseType = 0
End If

End Select
'-----
'--- now that we have all the variables filled in, let's get to work
'--- cleanup any old .err files
'-----
For i = 0 to 8
    If fs.FileExists(LogPath & TableArray(i) & ".err") Then
        fs.DeleteFile LogPath & TableArray(i) & ".err"
    End If
Next
For i = 0 to 21
    If fs.FileExists(LogPath & LogFileArray(i)) Then
        fs.DeleteFile LogPath & LogFileArray(i)
    End If
Next
'-----
'--- now grab the version of SQL Server you are running this against
'-----
Set oExec = WshShell.Exec("isql -Usa -P" & saPassword & " -S" &
ServerName & " -e -i" & ScriptDirectory & "utility\version.sql -o" &
LogDirectory & "version.log")
Do While oExec.Status = 0
    wScript.Sleep 100
Loop
rc = CheckSQLOutput(LogDirectory & "version.log")
If rc <> 0 Then
    wScript.Quit
End If
If (BuildOption = "full" OR BuildOption = "bulddb") Then
    wScript.Echo "Removing any existing TPCC database and backup
devices..."
    Set oExec = WshShell.Exec("isql -Usa -P" & saPassword & " -S" &
ServerName & " -e -i" & ScriptDirectory & NumberWarehouses &
".war\database\removedb.sql -o" & LogDirectory & "removedb.log")
    Do While oExec.Status = 0
        wScript.Sleep 100
    Loop
    rc = CheckSQLOutput(LogDirectory & "removedb.log")
    If rc <> 0 Then
        wScript.Quit
    End If
    wScript.Echo "Building database files and database..."
    Set oExec = WshShell.Exec("isql -Usa -P" & saPassword & " -S" &
ServerName & " -e -i" & ScriptDirectory & NumberWarehouses &
".war\database\createdb.sql -o" & LogDirectory & "createdb.log")

```

```

Do While oExec.Status = 0
    wScript.Sleep 100
Loop
rc = CheckSQLOutput(LogDirectory & "createdb.log")
If rc <> 0 Then
    wScript.Quit
End If
End If
'-----
'--- build tables and stored procedures
'-----
If (BuildOption = "full" OR BuildOption = "bulddb" _
OR BuildOption = "objects" OR BuildOption = "objectsfull") Then
    wScript.Echo "Creating TPC-C database tables..."
    Set oExec = WshShell.Exec("isql -Usa -P" & saPassword & " -S" &
ServerName & " -e -i" & ScriptDirectory & NumberWarehouses &
".war\ddl\tables.sql -o" & LogDirectory & "tables.log")
    Do While oExec.Status = 0
        wScript.Sleep 100
    Loop
    rc = CheckSQLOutput(LogDirectory & "tables.log")
    If rc <> 0 Then
        wScript.Quit
    End If
    wScript.Echo "Creating database objects..."
    Set oExec = WshShell.Exec("isql -Usa -P" & saPassword & " -S" &
ServerName & " -e -i" & ScriptDirectory & "dml\neword.sql -o" &
LogDirectory & "sp_neword.log")
    Do While oExec.Status = 0
        wScript.Sleep 100
    Loop
    rc = CheckSQLOutput(LogDirectory & "sp_neword.log")
    If rc <> 0 Then
        wScript.Quit
    End If
    Set oExec = WshShell.Exec("isql -Usa -P" & saPassword & " -S" &
ServerName & " -e -i" & ScriptDirectory & "dml\payment.sql -o" &
LogDirectory & "sp_payment.log")
    Do While oExec.Status = 0
        wScript.Sleep 100
    Loop
    rc = CheckSQLOutput(LogDirectory & "sp_payment.log")
    If rc <> 0 Then
        wScript.Quit
    End If
    Set oExec = WshShell.Exec("isql -Usa -P" & saPassword & " -S" &
ServerName & " -e -i" & ScriptDirectory & "dml\ordstat.sql -o" &
LogDirectory & "sp_ordstat.log")
    Do While oExec.Status = 0
        wScript.Sleep 100
    Loop
    rc = CheckSQLOutput(LogDirectory & "sp_ordstat.log")
    If rc <> 0 Then
        wScript.Quit
    End If
    Set oExec = WshShell.Exec("isql -Usa -P" & saPassword & " -S" &
ServerName & " -e -i" & ScriptDirectory & "dml\delivery.sql -o" &
LogDirectory & "sp_delivery.log")
    Do While oExec.Status = 0
        wScript.Sleep 100
    Loop
    rc = CheckSQLOutput(LogDirectory & "sp_delivery.log")
    If rc <> 0 Then
        wScript.Quit
    End If
    Set oExec = WshShell.Exec("isql -Usa -P" & saPassword & " -S" &
ServerName & " -e -i" & ScriptDirectory & "dml\stocklev.sql -o" &
LogDirectory & "sp_stocklev.log")

```

```

Do While oExec.Status = 0
    wScript.Sleep 100
Loop
rc = CheckSQLOutput(LogDirectory & "sp_stocklev.log")
If rc <> 0 Then
    wScript.Quit
End If
Set oExec = WshShell.Exec("isql -Usa -P" & saPassword & " -S" &
ServerName & " -e -i" & ScriptDirectory & "dml\version.sql -o" &
LogDirectory & "sp_version.log")
Do While oExec.Status = 0
    wScript.Sleep 100
Loop
rc = CheckSQLOutput(LogDirectory & "sp_version.log")
If rc <> 0 Then
    wScript.Quit
End If
wScript.Echo "Database object creation complete..."
End If
If (BuildOption = "full" OR BuildOption = "bulddb" _
OR BuildOption = "objects" OR BuildOption = "objectsfull" _
OR BuildOption = "bulkload" OR BuildOption = "bulkloadfull") Then
    wScript.Echo "Setting database options before load..."
    Set oExec = WshShell.Exec("isql -Usa -P" & saPassword & " -S" &
ServerName & " -e -i" & ScriptDirectory & "utility\dbopt1.sql -o" &
LogDirectory & "dbopt1.log")
    Do While oExec.Status = 0
        wScript.Sleep 100
    Loop
    rc = CheckSQLOutput(LogDirectory & "dbopt1.log")
    If rc <> 0 Then
        wScript.Quit
    End If
    '-----
    '--- before we start tpccldr.exe, check the registry
    '--- to ensure that the Shared Memory Protocol is off.
    '--- if it is on, store the setting so we can return
    '--- the system to the pre-tpccldr state.
    '-----
    SharedMemoryRegKey =
WshShell.RegRead("HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\MS
SQLServer\Client\SharedMemoryOn")
    If SharedMemoryRegKey = 1 Then
        WshShell.RegWrite
"HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\MSSQLServer\Client\S
haredMemoryOn", 0, "REG_DWORD"
    End If
    wScript.Echo "Beginning data load and index creation..."
    CMD_String = SetupDirectory & "\loader\bin\tpccldr.exe"
    CMD_String = CMD_String & " -S" & ServerName
    CMD_String = CMD_String & " -Usa"
    CMD_String = CMD_String & " -P" & saPassword
    CMD_String = CMD_String & " -W" & NumberWarehouses
    CMD_String = CMD_String & " -f" & LogDirectory &
"bulkload.log"
    CMD_String = CMD_String & " -L" & LogDirectory
    CMD_String = CMD_String & " -d" & ScriptDirectory &
NumberWarehouses & ".war\ddl"
    CMD_String = CMD_String & " -c" & DatabaseType
    oExec = WshShell.Run(CMD_String, 2, true)
    '-----
    '--- now that the loader is finished, put the
    '--- SharedMemoryOn registry key back to its original
    '--- value.
    '-----
    If SharedMemoryRegKey = 1 Then

```

```

        WshShell.RegWrite
"HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\MSSQLServer\Client\S
haredMemoryOn", 1, "REG_DWORD"
    End If
    wScript.Echo "Setting database options after load..."
    Set oExec = WshShell.Exec("isql -Usa -P" & saPassword & " -S" &
ServerName & " -e -i" & ScriptDirectory & "utility\dbopt2.sql -o" &
LogDirectory & "dbopt2.log")
    Do While oExec.Status = 0
        wScript.Sleep 100
    Loop
    rc = CheckSQLOutput(LogDirectory & "dbopt2.log")
    If rc <> 0 Then
        wScript.Quit
    End If
    wScript.Echo "Data load and index creation complete."
    '-----
    '--- now parse the index creation logs
    '--- to see if there were any errors
    '--- there.
    '-----
    For i = 5 to 15
        rc = CheckSQLOutput(LogDirectory & LogFileArray(i))
        If rc <> 0 Then
            wScript.Quit
        End If
    Next
    wScript.Echo "Calculating initial database space usage...."
    Set oExec = WshShell.Exec("isql -Usa -P" & saPassword & " -S" &
ServerName & " -e -i" & ACIDDirectory & "space\scripts\spused.sql -o" &
ACIDDirectory & "space\spused.ver")
    Do While oExec.Status = 0
        wScript.Sleep 100
    Loop
    Set oExec = WshShell.Exec("isql -Usa -P" & saPassword & " -S" &
ServerName & " -e -i" & ACIDDirectory & "space\scripts\splog.sql -o" &
ACIDDirectory & "space\splog.ver")
    Do While oExec.Status = 0
        wScript.Sleep 100
    Loop
    Set oExec = WshShell.Exec("isql -Usa -P" & saPassword & " -S" &
ServerName & " -e -i" & ACIDDirectory & "space\scripts\spfiles.sql -o" &
ACIDDirectory & "space\spfiles.ver")
    Do While oExec.Status = 0
        wScript.Sleep 100
    Loop
    '-----
    '--- now that the loader is finished
    '--- check the .err files and if they
    '--- are of zero length, delete them.
    '-----
    Set fsErr = CreateObject("Scripting.FileSystemObject")
    Set fErr = fsErr.GetFolder(LogDirectory)
    Set fcErr = fErr.Files
    For Each fl In fcErr
        If fl.Type = "ERR File" Then
            If fl.Size = 0 Then
                fl.Delete
            End If
        End If
    Next
    Set fcErr = Nothing
    Set fErr = Nothing
    Set fsErr = Nothing
End If
If (BuildOption = "full" _
OR BuildOption = "objectsfull" _
OR BuildOption = "bulkloadfull" _

```

```

OR BuildOption = "backup") Then
    wScript.Echo "Creating Backup Device(s)..."
    Set oExec = WshShell.Exec("isql -Usa -P" & saPassword & " -S" &
ServerName & " -e -i" & ScriptDirectory & NumberWarehouses &
".war\database\backupdev.sql -o" & LogDirectory & "backupdev.log")
    Do While oExec.Status = 0
        wScript.Sleep 100
    Loop
    rc = CheckSQLOutput(LogDirectory & "backupdev.log")
    If rc <> 0 Then
        wScript.Quit
    End If
    wScript.Echo "Backing up database..."
    Set oExec = WshShell.Exec("isql -Usa -P" & saPassword & " -S" &
ServerName & " -e -i" & ScriptDirectory & NumberWarehouses &
".war\database\backup.sql -o" & LogDirectory & "backup.log")
    Do While oExec.Status = 0
        wScript.Sleep 100
    Loop
    rc = CheckSQLOutput(LogDirectory & "backup.log")
    If rc <> 0 Then
        wScript.Quit
    End If
    wScript.Echo "Database backup complete."
End If
If (BuildOption = "full"
OR BuildOption = "objectsfull"
OR BuildOption = "bulkloadfull") Then
    wScript.Echo "Verifying TPC-C database load..."
    Set oExec = WshShell.Exec("isql -Usa -P" & saPassword & " -S" &
ServerName & " -e -i" & ScriptDirectory & "utility\verifytpccload.sql -o" &
LogDirectory & "verifyload.log")
    Do While oExec.Status = 0
        wScript.Sleep 100
    Loop
    rc = CheckSQLOutput(LogDirectory & "verifyload.log")
    If rc <> 0 Then
        wScript.Quit
    End If
    wScript.Echo "Check logs\verifyload.log to verify database load."
End If
'-----
'--- display banner message
'-----
wScript.Echo
"*****"
wScript.Echo "*"
wScript.Echo "*" Microsoft TPC-C Benchmark Kit Ver. 4.41 - Setup Complete
*"
wScript.Echo "*"
wScript.Echo "*"
"*****"

```

## spinlock.h

```

/* FILE: SPINLOCK.H
*
* Copyright 1997 Microsoft Corp., All rights reserved.
*
* Source code licensed to Tandem Computers for Internal
* use only. Redistribution of source or object files or
* any derivative works is prohibited. By agreement, this
* notice may not be removed.
*
* Authors: Mike Parkes, Charles Levine, Philip Durr
* Microsoft Corp.

```

```

*/
#ifdef _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 memory
* 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
*sl );

    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

```

## tm\_com\_dll.dsp

# Microsoft Developer Studio Project File - Name="tm\_com\_dll" - Package Owner=<4>

# Microsoft Developer Studio Generated Build File, Format Version 6.00  
# \*\* DO NOT EDIT \*\*

# TARGETTYPE "Win32 (x86) Dynamic-Link Library" 0x0102

CFG=tm\_com\_dll - Win32 Debug

!MESSAGE This is not a valid makefile. To build this project using NMAKE,  
!MESSAGE use the Export Makefile command and run

```

!MESSAGE
!MESSAGE NMAKE /f "tm_com_dll.mak".
!MESSAGE
!MESSAGE You can specify a configuration when running NMAKE
!MESSAGE by defining the macro CFG on the command line. For example:
!MESSAGE
!MESSAGE NMAKE /f "tm_com_dll.mak" CFG="tm_com_dll - Win32
Debug"
!MESSAGE
!MESSAGE Possible choices for configuration are:
!MESSAGE
!MESSAGE "tm_com_dll - Win32 Release" (based on "Win32 (x86)
Dynamic-Link Library")
!MESSAGE "tm_com_dll - Win32 Debug" (based on "Win32 (x86)
Dynamic-Link Library")
!MESSAGE

# Begin Project
# PROP AllowPerConfigDependencies 0
# PROP Scc_ProjName ""
# PROP Scc_LocalPath ""
CPP=cl.exe
MTL=midl.exe
RSC=rc.exe

!IF "$(CFG)" == "tm_com_dll - 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 ".\obj"
# PROP Ignore_Export_Lib 0
# PROP Target_Dir ""
# ADD BASE CPP /nologo /MT /W3 /GX /O2 /D "WIN32" /D "NDEBUG" /D
"_WINDOWS" /YX /FD /c
# ADD CPP /nologo /MD /W3 /GX /O2 /D "WIN32" /D "NDEBUG" /D
"_WINDOWS" /YX /FD /c
# ADD BASE MTL /nologo /D "NDEBUG" /mktyplib203 /o "NUL" /win32
# ADD MTL /nologo /D "NDEBUG" /mktyplib203 /o "NUL" /win32
# ADD BASE RSC /I 0x409 /d "NDEBUG"
# ADD RSC /I 0x409 /d "NDEBUG"
BSC32=bscmake.exe
# ADD BASE BSC32 /nologo
# ADD BSC32 /nologo
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 odbcc32.lib
odbccp32.lib /nologo /subsystem:windows /dll /machine:I386
# ADD LINK32 kernel32.lib user32.lib gdi32.lib winspool.lib comdlg32.lib
advapi32.lib shell32.lib ole32.lib oleaut32.lib uuid.lib odbcc32.lib odbccp32.lib
/nologo /subsystem:windows /dll /machine:I386 /out:".bin\tpcc_com.dll"

!ELSEIF "$(CFG)" == "tm_com_dll - 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 ".\obj"

```

```

# PROP Ignore_Export_Lib 0
# PROP Target_Dir ""
# ADD BASE CPP /nologo /MTd /W3 /Gm /GX /ZI /Od /D "WIN32" /D
" _DEBUG" /D " _WINDOWS" /YX /FD /c
# ADD CPP /nologo /MDd /W3 /Gm /GX /ZI /Od /D "WIN32" /D " _DEBUG"
/D " _WINDOWS" /YX /FD /c
# ADD BASE MTL /nologo /D " _DEBUG" /mktyplib203 /o "NUL" /win32
# ADD MTL /nologo /D " _DEBUG" /mktyplib203 /o "NUL" /win32
# ADD BASE RSC /I 0x409 /d " _DEBUG"
# ADD RSC /I 0x409 /d " _DEBUG"
BSC32=bscmake.exe
# ADD BASE BSC32 /nologo
# ADD BSC32 /nologo
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
odbc32.lib /nologo /subsystem:windows /dll /debug /machine:I386
/pdbtype:sept
# ADD LINK32 kernel32.lib user32.lib gdi32.lib winspool.lib comdlg32.lib
advapi32.lib shell32.lib ole32.lib oleaut32.lib uuid.lib odbc32.lib odbc32.lib
/nologo /subsystem:windows /dll /debug /machine:I386
/out:"\bin\tpcc_com.dll" /pdbtype:sept

!ENDIF

# Begin Target

# Name "tm_com_dll - Win32 Release"
# Name "tm_com_dll - Win32 Debug"
# Begin Source File

SOURCE=\src\tpcc_com.cpp
# End Source File
# Begin Source File

SOURCE=\src\tpcc_com.h
# End Source File
# End Target
# End Project

```

## 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 <char.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

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 "410"

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
= 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

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

/* FUNCTION: DIIMain
*
* 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 WINAPI DIIMain(HANDLE hModule, DWORD ul_reason_for_call,
LPVOID lpReserved)
{
    DWORD i;
    char szEvent[LEN_ERR_STRING] = "0";
    char szLogFile[128];
    char szDllName[128];

```

```

// debugging...
// DebugBreak();

    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 );
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");
LoadLibrary( szDllName );
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() );
        }
// load DLL for database
connection
(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
        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-yymmdd-hhmm.log
                SYSTEMTIME Time;
                GetLocalTime( &Time
);
                wsprintf( szLogFile,
"%sdelivery-%2.2d%2.2d%2.2d-%2.2d%2.2d.log",
Reg.szPath, Time.wYear % 100, Time.wMonth, Time.wDay, Time.wHour,
Time.wMinute );
                txnDelilog = new
CTxnLog(szLogFile, TXN_LOG_WRITE);
                //write event into txn
log for START

```



```

txnDelilog->WriteCtrlRecToLog(TXN_EVENT_START,
szMyComputerName, sizeof(szMyComputerName));

// allocate structures for
delivery buffers and thread mgmt
HANDLE[dwNumDeliveryThreads];
pDeliHandles = new
DELIVERY_TRANSACTION[dwDelBuffSize];
pDelBuff = new
// launch
DeliveryWorkerThread to perform actual delivery txns
for(i=0;
i<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->WriteCtrlRecToLog(TXN_EVENT_STOP, szMyComputerName,
sizeof(szMyComputerName));
// This will
CTxnLog
*txnDelilogLocal = txnDelilog;
txnDelilog=
NULL;
delete
txnDelilogLocal;
}
delete [] pDeliHandles;
delete [] pDelBuff;
CloseHandle(
hWorkerSemaphore );
CloseHandle(
hDoneEvent );
DeleteCriticalSection(&DelBuffCriticalSection);
}
DeleteCriticalSection(&TermCriticalSection);
if (hLibInstanceTm != NULL)
FreeLibrary(
hLibInstanceTm );
hLibInstanceTm = NULL;

```

```

if (hLibInstanceDb != NULL)
FreeLibrary(
hLibInstanceDb = NULL;
Sleep(500);
break;
default:
/* nothing */;
}
}
catch (CBaseErr *e)
{
WriteMessageToEventLog( e->ErrorText() );
delete e;
TerminateExtension(0);
return FALSE;
}
catch (...)
{
WriteMessageToEventLog(TEXT("Unhandled
exception. DLL could not load."));
TerminateExtension(0);
return FALSE;
}
return TRUE;
}
/* FUNCTION: GetExtensionVersion
*
* PURPOSE: This function is called by the inet service when the DLL
is first loaded.
*
* ARGUMENTS: HSE_VERSION_INFO *pVer passed in
structure in which to place expected version number.
*
* RETURNS: TRUE inet service expected return value.
*/
BOOL WINAPI GetExtensionVersion(HSE_VERSION_INFO *pVer)
{
pVer->dwExtensionVersion =
MAKELONG(HSE_VERSION_MINOR, HSE_VERSION_MAJOR);
lstrcpy(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];
                sprintf( 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:
        // new-order selected from menu; display
new-order input form
        MakeNewOrderForm(TermId, NULL,
INPUT_FORM, szBuffer);
        break;

    case 3:
        // payment selected from menu; display
payment input form
        MakePaymentForm(TermId, NULL,
INPUT_FORM, szBuffer);
        break;

    case 4:
        // delivery selected from menu; display
delivery input form
        MakeDeliveryForm(TermId, NULL,
INPUT_FORM, szBuffer);
    }
}

```

```

        break;
    case 5:
        // order-status selected from menu; display
        order-status input form
        MakeOrderStatusForm(TermId, NULL,
        INPUT_FORM, szBuffer);
        break;
    case 6:
        // stock-level selected from menu; display
        stock-level input form
        MakeStockLevelForm(TermId, NULL,
        INPUT_FORM, szBuffer);
        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

```

```

assert(txnDelilog != NULL);

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;
    goto ErrorExit;
}
catch (...)
{
    WriteMessageToEventLog(TEXT("Unhandled exception
caught in DeliveryWorkerThread.));
    goto ErrorExit;
}

while (TRUE)
{
    try
    {
        //while delivery thread running, i.e. user has
        not requested termination
        while (TRUE)
        {
            // need to wait for multiple objects:
            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(short 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
GetIntKeyValue(&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 4.20)</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\"'"

```

```

TYPE="hidden" NAME="ERROR" VALUE="0">"
TYPE="hidden" NAME="FORMID" VALUE="1">"
TYPE="hidden" NAME="TERMIN" VALUE="0">"
TYPE="hidden" NAME="SYNCID" VALUE="0">"
TYPE="hidden" NAME="VERSION" VALUE=""
WEBCLIENT_VERSION ">"
);

printf( szTmp, "Configuration Settings: <BR><font
face="Courier New" color="blue"><PRE>"
<B>%s</B><BR>"
= <B>%s</B><BR>"
= <B>%d</B><BR>"
= <B>%d</B><BR>"
Deliveries = <B>%d</B><BR>"
szDBNames[Reg.eDB_Protocol], szTxnMonNames[Reg.eTxnMon],
Reg.dwMaxConnections,
dwNumDeliveryThreads, dwDelBuffSize );
strcat( szBuffer, szTmp);

if (Reg.eTxnMon == COM)
{
printf( 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
printf( szTmp, "Please enter your database options
for this connection:<BR>"
"DB Server
"DB User
ID = <INPUT NAME="db_user" SIZE=20 VALUE="<B>%s</B>">"
"DB
Password = <INPUT NAME="db_passwd" SIZE=20
VALUE="<B>%s</B>">"
"DB Name
= <INPUT NAME="db_name" SIZE=20 VALUE="<B>%s</B>">"

"</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
printf( szTmp, "Database options which will be
used by the transaction monitor:<BR>"
"DB Server
"DB User
ID = <INPUT NAME="db_user" SIZE=20 VALUE="<B>%s</B>">"
"DB
Password = <INPUT NAME="db_passwd" SIZE=20
VALUE="<B>%s</B>">"
"DB Name
= <INPUT NAME="db_name" SIZE=20 VALUE="<B>%s</B>">"

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

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);
}
}

```

```

"DB Server
"DB User
"DB
"DB Name
, Reg.szDbServer, Reg.szDbUser,
Reg.szDbPassword, Reg.szDbName );
strcat( szBuffer, szTmp);

printf( szTmp, "Please enter your Warehouse and District for
this session:<BR>"
"District ID
= <INPUT NAME="d_id" SIZE=2><BR>"

"</PRE></font><HR>"
"DB User
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);

iTotal = 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
*
* pKey         char
key value to look for
*
* pValue       char
character array into which to place key's value
*
* iMax         int
maximum length of key value array.
*
* err          WEBERROR
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
*              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
*
* pKey         char
key value to look for
*
* NoKeyErr     WEBERROR
error value to throw if key not found
*
* NotIntErr    WEBERROR
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 Occured</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-C Main
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 = sprintf(szForm,
                "<HTML><HEAD><TITLE>TPC-C Stock
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='TERMI'"
VALUE="\"%d\">"
                "<INPUT TYPE='hidden' NAME='SYNCID'"
VALUE="\"%d\">"
                "<PRE><font face='\"Courier\"'"
Stock-Level<BR>"
                "Warehouse: %4.4d 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>
                "<INPUT TYPE='submit' NAME='CMD'"
VALUE="\"Process\">"
                "<INPUT TYPE='submit' NAME='CMD'"
VALUE="\"Menu\">"
                "</FORM></HTML>");
    }
    else
    {
        sprintf(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>"
                "<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> <BR> <BR>";

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

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

    c = sprintf(szForm,
                "<HTML><HEAD><TITLE>TPC-C New
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='TERMI'"
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: %4.4d ",
                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: %4.4d 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,
"%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, " %4.4d
%6.6d %-24s %2.2d %3.3d %1.1s $%6.2f $%7.2f <BR>",
        pNewOrderData->OL[i].ol_supply_w_id,
        pNewOrderData->OL[i].ol_i_id,
        pNewOrderData->OL[i].ol_i_name,
        pNewOrderData->OL[i].ol_quantity,
        pNewOrderData->OL[i].ol_stock,
        pNewOrderData->OL[i].ol_brand_generic,
        pNewOrderData->OL[i].ol_i_price,
        pNewOrderData->OL[i].ol_amount );
        }
        }
        else
        {
            c += sprintf(szForm+c,
"%Disc:<BR>"
"Order Number: %8.8d Number
of Lines:      W_tax:      D_tax:<BR><BR>"

```

```

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 blnput, 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 ( !blnput )
{
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 ( blnput )
{
c += sprintf(szForm+c,
"<BR> <BR>Warehouse: %4.4d"
" District: <INPUT
NAME=\"DID*\" SIZE=1><BR> <BR> <BR> <BR> <BR>"
"Customer: <INPUT NAME=\"CID*\"
SIZE=4>"
" Cust-Warehouse: <INPUT NAME=\"CWI*\"
SIZE=4> "
" 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: %4.4d
District: %2.2d<BR>"
"%0-20s %0-20s<BR>"
"%0-20s %0-20s<BR>"
"%0-20s %0-2s %5.5s-%4.4s %0-20s %0-2s
%5.5s-%4.4s<BR> <BR>"
"Customer: %4.4d Cust-Warehouse: %4.4d
Cust-District: %2.2d<BR>"
"Name: %0-16s %0-2s %0-16s Since:
%2.2d-%2.2d-%4.4d<BR>"
" %0-20s Credit: %0-2s<BR>"
, Term.pClientData[iTermId].w_id,
pPaymentData->w_street_1,
pPaymentData->w_street_2,
pPaymentData->w_city,
pPaymentData->w_state, pPaymentData->w_zip, pPaymentData->w_zip+5,
pPaymentData->d_city,
pPaymentData->d_state, pPaymentData->d_zip, pPaymentData->d_zip+5

```

```

        , pPaymentData->c_id,
pPaymentData->c_w_id, pPaymentData->c_d_id
        , pPaymentData->c_first,
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
    );

    c += sprintf(szForm+c,
        "    %-20s    %5.2f<BR>",
        pPaymentData->c_street_2,
        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
            strcpy(szForm+c, "Cust-Data: <BR> <BR>
<BR> <BR>");

        strcat(szForm,
            " <BR></font></PRE><HR>"
            "TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..NewOrder..\">"
            "TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Payment..\">"
            "TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Delivery..\">"
            "TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Order-Status..\">"
            "TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Stock-Level..\">"
            "TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Exit..\">"
            "</BODY></FORM></HTML>");
    }
}

/* 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=\"TERMDID\"
VALUE=\"%d\">"
        "<INPUT TYPE=\"hidden\" NAME=\"SYNCID\"
VALUE=\"%d\">"
        "<PRE><font face=\"Courier\">
Order-Status<BR>"
        "Warehouse: %4.4d    ",
        ORDER_STATUS_FORM, iTermId,
        Term.pClientData[iTermId].iSyncId, Term.pClientData[iTermId].w_id);

    if ( bInput )
    {
        strcpy(szForm+c,
            "District: <INPUT NAME=\"DID*\"
SIZE=1><BR>"
            "Customer: <INPUT NAME=\"CID*\"
SIZE=4> Name:    <INPUT 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> <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_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, " %4.4d %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);
        }

        strncpy( 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: %4.4d<BR> <BR>",
                (!bInput && (pDeliveryData->exec_status_code !=
eOK)) ? ERR_TYPE_DELIVERY_POST : 0,
                DELIVERY_FORM, iTermId,
                Term.pClientData[iTermId].iSynclId, 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> </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> </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;

    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);

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

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

```

```

/* 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;

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

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

    pPayment =
Term.pClientData[iTermId].pTxn->BuffAddr_Payment();
    MakePaymentForm(iTermId, pPayment, OUTPUT_FORM,
szBuffer);
}

```

```

/* 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)
{
    char szTmp[26];
    int i;
    short items;
    int ol_i_id, ol_quantity;
    char *ptr = lpszQueryString;

    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
= (short)atoi(szTmp);

            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;
    }

/* 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)
{
    char    szTmp[26];
    char    *ptr = lpszQueryString;
    BOOL    bCustIdBlank;

    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 ( 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(pPaymentData->c_last) > 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 );
    }

/* 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(pOrderStatusData->c_last) >
LAST_NAME_LEN )
            throw new CWEBCLNT_ERR(
ERR_ORDERSTATUS_CLT_RANGE );
            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;
}

```

## 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

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 "410"

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

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

/* 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 occured in initialization
* TRUE
DLL successfully initialized
*/

BOOL WINAPI DllMain(HANDLE hModule, DWORD ul_reason_for_call,
LPVOID lpReserved)
{
    DWORD i;
    char szEvent[LEN_ERR_STRING] = "";
    char szLogFile[128];
    char szDllName[128];

    // debugging...
    // DebugBreak();

    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");
        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");
        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");
        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() );
    }

    // load DLL for database
    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");
        }
    }
}

```

<pre> 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(&amp;DelBuffCriticalSection); hWorkerSemaphore = CreateSemaphore( NULL, 0, dwDelBuffSize, NULL ); dwDelBuffFreeCount = dwDelBuffSize; InitJulianTime(NULL); // create unique log file name based on delilog-yymmdd-hhmm.log SYSTEMTIME Time; GetLocalTime( &amp;Time ); ); wsprintf( szLogFile, "%sdelivery-%2.2d%2.2d%2.2d-%2.2d%2.2d.log", Reg.szPath, Time.wYear % 100, Time.wMonth, Time.wDay, Time.wHour, Time.wMinute ); txndelilog = new CTxnLog(szLogFile, TXN_LOG_WRITE); //write event into txn log for START txndelilog-&gt;WriteCtrlRecToLog(TXN_EVENT_START, szMyComputerName, sizeof(szMyComputerName)); // allocate structures for delivery buffers and thread mgmt pDeliHandles = new HANDLE[dwNumDeliveryThreads]; pDelBuff = new DELIVERY_TRANSACTION[dwDelBuffSize]; // launch DeliveryWorkerThread to perform actual delivery txns for(i=0; i&lt;dwNumDeliveryThreads; i++) { </pre>	<pre> 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; delete [] pDelBuff; CloseHandle( hWorkerSemaphore ); CloseHandle( 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() ); } </pre>
---	---

```

        delete e;
        TerminateExtension(0);
        return FALSE;
    }
    catch (...)
    {
        WriteMessageToEventLog(TEXT("Unhandled
exception. DLL could not load.));
        TerminateExtension(0);
        return FALSE;
    }

    return TRUE;
}

/* FUNCTION: GetExtensionVersion
*
* PURPOSE:      This function is called by the inet service when the DLL
is first loaded.
*
* ARGUMENTS:   HSE_VERSION_INFO      *pVer      passed in
structure in which to place expected version number.
*
* RETURNS:     TRUE      inet service expected return value.
*/

BOOL WINAPI GetExtensionVersion(HSE_VERSION_INFO *pVer)
{
    pVer->dwExtensionVersion =
    MAKELONG(HSE_VERSION_MINOR, HSE_VERSION_MAJOR);
    lstrcpy(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.
*
* ARGUMENTS:   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:
        // new-order selected from menu; display
        new-order input form
        INPUT_FORM, szBuffer);
        MakeNewOrderForm(TermId, NULL,
        break;
    case 3:
        // payment selected from menu; display
        payment input form
        INPUT_FORM, szBuffer);
        MakePaymentForm(TermId, NULL,
        break;
    case 4:
        // delivery selected from menu; display
        delivery input form
        INPUT_FORM, szBuffer);
        MakeDeliveryForm(TermId, NULL,
        break;
    case 5:
        // order-status selected from menu; display
        order-status input form
        INPUT_FORM, szBuffer);
        MakeOrderStatusForm(TermId, NULL,
        break;
    case 6:
        // stock-level selected from menu; display
        stock-level input form
        INPUT_FORM, szBuffer);
        MakeStockLevelForm(TermId, NULL,
        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"));
}

```

```

    _sprintf(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

    assert(txnDeliLog != NULL);

    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];
        sprintf( 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;
        goto ErrorExit;
    }
    catch (...)
    {
        WriteMessageToEventLog(TEXT("Unhandled exception
caught in DeliveryWorkerThread. "));
        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];
        sprintf( 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(short 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 *pSynclId)
{
    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;
    *pTermId = 0;
    // default is the login screen

    // 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 4.20)</BIG></B> <BR> <BR>"
"<font
face=\\"Courier New\\"><PRE>"
" __DATE__ ", " __TIME__ " <BR>"
"Source:
" __FILE__ " (" __TIMESTAMP__ ") <BR>"
"</PRE></font>"
" <FORM
" <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=\\""
WEBCIENT_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>"
"DB Server
= <B>%s</B><BR>"
"DB User
ID = <B>%s</B><BR>"
"DB
Password = <B>%s</B><BR>"
"DB 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=4><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 iTotal;

    EnterCriticalSection(&TermCriticalSection);

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

    LeaveCriticalSection(&TermCriticalSection);

    wsprintf( szBuffer,

```

```

        "<HTML><HEAD><TITLE>TPC-C Web
Client Stats</TITLE></HEAD>"
        "<BODY><B><BIG> Total Active
Connections: %d </BIG></B><BR></BODY></HTML>"
        , iTot);
}

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)
        wsprintf( szTmp+strlen(szTmp), " Error=%d",
m_SystemErr );
    m_szErrorText = new char[strlen(szTmp)+1];
    strcpy( m_szErrorText, szTmp );
    return m_szErrorText;
}
/* FUNCTION: GetKeyValue
*
* PURPOSE: This function parses a http formatted string for specific
key values.
*
* ARGUMENTS: 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
 *
 * pKey          char
key value to look for
 *
 * NoKeyErr      WEBERROR
error value to throw if key not found
 *
 * NotIntErr     WEBERROR
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=value0. 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 TPC.C.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\">"

```

```

VALUE="%d\>"      "<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=\"TERMIN\"
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)
{
    sprintf(szForm,
        "<HTML><HEAD><TITLE>TPC-C Main
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=\"TERMIN\"
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 = sprintf(szForm,
        "<HTML><HEAD><TITLE>TPC-C Stock
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=\"TERMIN\"
VALUE="%d\>"
        "<INPUT TYPE=\"hidden\" NAME=\"SYNCID\"
VALUE="%d\>"
        "<PRE><font face=\"Courier\">
Stock-Level<BR>"
        "Warehouse: %4.4d 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
    {
        sprintf(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></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..\">"

```

```

VALUE="\"..Exit..\">"
    "<INPUT TYPE=\"submit\" NAME=\"CMD\"
    "</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> <BR>";

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

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

    c = sprintf(szForm,
        "<HTML><HEAD><TITLE>TPC-C New
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=\"TERMIN\"
VALUE=\"%d\">"
        "<INPUT TYPE=\"hidden\" NAME=\"SYCNID\"
VALUE=\"%d\">"
        "<PRE><font face=\"Courier\">
New Order<BR>"
        , bValid ? 0 : ERR_BAD_ITEM_ID,
NEW_ORDER_FORM, iTermId, Term.pClientData[iTermId].iSynclId);

    if ( bInput )
    {
        c += sprintf(szForm+c, "Warehouse: %4.4d ",
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>
NAME=\"Qty00*\" SIZE=1><BR>"

```

```

    "<INPUT NAME=\"SP01*\" SIZE=4>
<INPUT NAME=\"IID01*\" SIZE=6>
NAME=\"Qty01*\" SIZE=1><BR>"
    "<INPUT NAME=\"SP02*\" SIZE=4>
<INPUT NAME=\"IID02*\" SIZE=6>
NAME=\"Qty02*\" SIZE=1><BR>"
    "<INPUT NAME=\"SP03*\" SIZE=4>
<INPUT NAME=\"IID03*\" SIZE=6>
NAME=\"Qty03*\" SIZE=1><BR>"
    "<INPUT NAME=\"SP04*\" SIZE=4>
<INPUT NAME=\"IID04*\" SIZE=6>
NAME=\"Qty04*\" SIZE=1><BR>"
    "<INPUT NAME=\"SP05*\" SIZE=4>
<INPUT NAME=\"IID05*\" SIZE=6>
NAME=\"Qty05*\" SIZE=1><BR>"
    "<INPUT NAME=\"SP06*\" SIZE=4>
<INPUT NAME=\"IID06*\" SIZE=6>
NAME=\"Qty06*\" SIZE=1><BR>"
    "<INPUT NAME=\"SP07*\" SIZE=4>
<INPUT NAME=\"IID07*\" SIZE=6>
NAME=\"Qty07*\" SIZE=1><BR>"
    "<INPUT NAME=\"SP08*\" SIZE=4>
<INPUT NAME=\"IID08*\" SIZE=6>
NAME=\"Qty08*\" SIZE=1><BR>"
    "<INPUT NAME=\"SP09*\" SIZE=4>
<INPUT NAME=\"IID09*\" SIZE=6>
NAME=\"Qty09*\" SIZE=1><BR>"
    "<INPUT NAME=\"SP10*\" SIZE=4>
<INPUT NAME=\"IID10*\" SIZE=6>
NAME=\"Qty10*\" SIZE=1><BR>"
    "<INPUT NAME=\"SP11*\" SIZE=4>
<INPUT NAME=\"IID11*\" SIZE=6>
NAME=\"Qty11*\" SIZE=1><BR>"
    "<INPUT NAME=\"SP12*\" SIZE=4>
<INPUT NAME=\"IID12*\" SIZE=6>
NAME=\"Qty12*\" SIZE=1><BR>"
    "<INPUT NAME=\"SP13*\" SIZE=4>
<INPUT NAME=\"IID13*\" SIZE=6>
NAME=\"Qty13*\" SIZE=1><BR>"
    "<INPUT NAME=\"SP14*\" SIZE=4>
<INPUT NAME=\"IID14*\" SIZE=6>
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: %4.4d 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,
                    "%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, " %4.4d
%6.6d %-24s %2.2d %3.3d %1.1s $%6.2f $%7.2f <BR>",
                        pNewOrderData->OL[i].ol_supply_w_id,
                        pNewOrderData->OL[i].ol_i_id,
                        pNewOrderData->OL[i].ol_i_name,
                        pNewOrderData->OL[i].ol_quantity,
                        pNewOrderData->OL[i].ol_stock,
                        pNewOrderData->OL[i].ol_brand_generic,
                        pNewOrderData->OL[i].ol_i_price,
                        pNewOrderData->OL[i].ol_amount );
        }
        else
        {
            c += 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 )

```

```

Transaction committed.
        c += sprintf(szForm+c, "Execution Status:
Total: %8.2f ",
                    pNewOrderData->total_amount);
    else
        c += sprintf(szForm+c, "Execution Status:
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 blnInput, 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 ( !blnInput )
    {
        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: %4.4d"
            "          District: <INPUT
NAME=\"DID*\" SIZE=1><BR> <BR> <BR> <BR> <BR>"
            "Customer: <INPUT NAME=\"CID*\"
SIZE=4>"
            "Cust-Warehouse: <INPUT NAME=\"CWI*\"
SIZE=4> "
            "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: %4.4d
District: %2.2d<BR>"
            "%-20s          %-20s<BR>"
            "%-20s          %-20s<BR>"
            "%-20s %-2s %-5.5s-%4.4s   %-20s %-2s
%5.5s-%4.4s<BR> <BR>"
            "Customer: %4.4d Cust-Warehouse: %4.4d
Cust-District: %2.2d<BR>"
            "Name:  %-16s %-2s %-16s   Since:
%-20s          Credit: %-2s<BR>"
            , Term.pClientData[iTermId].w_id,
            pPaymentData->d_id,
            pPaymentData->w_street_1,
            pPaymentData->w_street_2,
            pPaymentData->d_street_2,
            pPaymentData->w_city,
            pPaymentData->w_state, pPaymentData->w_zip, pPaymentData->w_zip+5,
            pPaymentData->d_city,
            pPaymentData->d_state, pPaymentData->d_zip, pPaymentData->d_zip+5,
            pPaymentData->c_id,
            pPaymentData->c_d_id,
            pPaymentData->c_first,
            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
        );
        c += sprintf(szForm+c,

```

```

            "          %-20s          %%Disc:
%5.2f<BR>",
            pPaymentData->c_street_2,
            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
                strcpy(szForm+c, "Cust-Data: <BR> <BR>
<BR> <BR>");
        strcat(szForm,
            " <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>");
    }
}
/* 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;

```



```

VALUE="%">"
Delivery<BR>"
    "Warehouse: %4.4d<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
    {
        sprintf( 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> </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;

    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);

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

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

/* 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;

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

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

    pPayment =
Term.pClientData[iTermId].pTxn->BuffAddr_Payment();
    MakePaymentForm(iTermId, pPayment, OUTPUT_FORM,
szBuffer);
}

/* 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)

```



```

{
    char    szTmp[26];
    int     i;
    short   items;
    int     ol_i_id, ol_quantity;
    char    *ptr = lpszQueryString;

    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
= (short)atoi(szTmp);

            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;
    }

    /* 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)
    {
        char    szTmp[26];
        char    *ptr = lpszQueryString;
        BOOL    bCustIdBlank;

        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 ( 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(pPaymentData->c_last) > 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 );
}

/* 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(pOrderStatusData->c_last) >
LAST_NAME_LEN )
            throw new CWEBCLNT_ERR(
ERR_ORDERSTATUS_CLT_RANGE );
        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;
}

```

## tpcc.def

LIBRARY TPCC.DLL

EXPORTS

```

GetExtensionVersion @1
HttpExtensionProc @2
TerminateExtension @3

```

## tpcc.h

```

// File: TPCC.H
// Microsoft TPC-C Kit Ver. 4.41
// Copyright Microsoft, 1996, 1997,
1998, 1999, 2000, 2001
// Purpose: Header file for TPC-C database loader

```

```

// Build number of TPC Benchmark Kit
#define TPCKIT_VER "4.41"

```

```

// 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 DEFLDPACKSIZE 32768
#define LOADER_RES_FILE "C:\\MSTPCC.440\\SETUP\\logs\\load.out"
#define LOG_PATH "C:\\MSTPCC.440\\SETUP\\LOGS\\";
#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 table_warehouse; //
set if loading WAREHOUSE, DISTRICT, and STOCK
    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 *log_path;
    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();

```

## tpcc.rc

//Microsoft Developer Studio generated resource script.

```

//
#include "resource.h"

```

```

#define APSTUDIO_READONLY_SYMBOLS
//

```

// Generated from the TEXTINCLUDE 2 resource.

```

//
#include "afxres.h"

```

```

//
#define APSTUDIO_READONLY_SYMBOLS

```

```

//
// 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

```

```

#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", "Microsoft0"

```

```

VALUE "FileDescription", "TPC-C HTML DLL Server (DBLIB)0"

```

```

VALUE "FileVersion", "0, 4, 0, 0\0"

```

```

VALUE "InternalName", "tpcc0"

```

```

VALUE "LegalCopyright", "Copyright © 19970"

```

```

VALUE "OriginalFilename", "tpcc.dll\0"

```

```

VALUE "ProductName", "Microsoft tpcc\0"

```

```

VALUE "ProductVersion", "0, 4, 0, 0\0"

```

```

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

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

#ifdef APSTUDIO_INVOKED
////////////////////////////////////
//
// Generated from the TEXTINCLUDE 3 resource.
//

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

tpcc_com.cpp

/*      FILE:          TPCC_COM.CPP
 *
 *      4.20.000
 *
 *      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
#defineDllDecl __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.h"

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

// wrapper routine for class constructor
__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_pTxn = NULL;
    m_pNewOrder = NULL;
    m_pPayment = NULL;
    m_pStockLevel = NULL;
    m_pOrderStatus = NULL;

    m_bSinglePool = bSinglePool;

    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);

        // 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_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);

    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);
    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].cElements);
        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].cElements);
    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].cElements);
    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].cElements);
    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

```

```

*
*          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"

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

class 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
        } 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);

```

## ***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 <atlibase.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_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");
                hLibInstanceDb = LoadLibrary(
szDllName );
                if (hLibInstanceDb == NULL)
                    throw new
CCOMPONENT_ERR( ERR_LOADDLL_FAILED, szDllName,
GetLastError() );

                // get function pointer to wrapper
                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");
        hLibInstanceDb = 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)
    wsprintf( 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* pObjContext = NULL;

    // get our object context
    HRESULT hr = CoGetObjectContext( IID_IObjectContext, (void
**) &pObjContext );
    pObjContext->SetComplete();
    ReleaseInterface(pObjContext);
    return hr;
}

//
// called by the ctor activator
//
//
//
// 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));

        m_pTxn->NewOrder(); // do the
actual txn

        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;

        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));

        m_pTxn->Payment(); // do the actual txn

        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;

        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 (...)
    {
}
}

        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));

        m_pTxn->StockLevel();

        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;

        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));

        m_pTxn->OrderStatus();

        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;

        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.dsp***

```

# Microsoft Developer Studio Project File - Name="tpcc_com_all" - Package
Owner=<4>
# Microsoft Developer Studio Generated Build File, Format Version 6.00
# ** DO NOT EDIT **

# TARGETTYPE "Win32 (x86) Dynamic-Link Library" 0x0102

CFG=tpcc_com_all - Win32 Debug
!MESSAGE This is not a valid makefile. To build this project using NMAKE,
!MESSAGE use the Export Makefile command and run
!MESSAGE
!MESSAGE NMAKE /f "tpcc_com_all.mak".
!MESSAGE
!MESSAGE You can specify a configuration when running NMAKE
!MESSAGE by defining the macro CFG on the command line. For example:
!MESSAGE
!MESSAGE NMAKE /f "tpcc_com_all.mak" CFG="tpcc_com_all - Win32
Debug"
!MESSAGE
!MESSAGE Possible choices for configuration are:
!MESSAGE
!MESSAGE "tpcc_com_all - Win32 Release" (based on "Win32 (x86)
Dynamic-Link Library")
!MESSAGE "tpcc_com_all - Win32 Debug" (based on "Win32 (x86)
Dynamic-Link Library")
!MESSAGE

# Begin Project
# PROP AllowPerConfigDependencies 0
# PROP Scc_ProjName ""
# PROP Scc_LocalPath ""
CPP=cl.exe
MTL=midl.exe
RSC=rc.exe

!IF "$(CFG)" == "tpcc_com_all - 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 ".\obj"
# PROP Ignore_Export_Lib 0
# PROP Target_Dir ""
# ADD BASE CPP /nologo /MT /W3 /GX /O2 /D "WIN32" /D "NDEBUG" /D
" _WINDOWS" /YX /FD /c
# ADD CPP /nologo /MT /W3 /GX /O2 /D "WIN32" /D "NDEBUG" /D
" _WINDOWS" /YX /FD /c
# ADD BASE MTL /nologo /D "NDEBUG" /mktyplib203 /o "NUL" /win32
# ADD MTL /nologo /D "NDEBUG" /mktyplib203 /o "NUL" /win32
# ADD BASE RSC /I 0x409 /d "NDEBUG"

```

```

# ADD RSC /I 0x409 /d "NDEBUG"
BSC32=bscmake.exe
# ADD BASE BSC32 /nologo
# ADD BSC32 /nologo
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 odbcc32.lib
odbccp32.lib /nologo /subsystem:windows /dll /machine:I386
# ADD LINK32 ..\db_dblib_dll\bin\tpcc_dblib.lib
..\db_odbc_dll\bin\tpcc_odbc.lib kernel32.lib user32.lib gdi32.lib winspool.lib
comdlg32.lib advapi32.lib shell32.lib ole32.lib oleaut32.lib uuid.lib odbcc32.lib
odbccp32.lib /nologo /subsystem:windows /dll /machine:I386

!ELSEIF "$(CFG)" == "tpcc_com_all - 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 ".\obj"
# PROP Ignore_Export_Lib 0
# PROP Target_Dir ""
# ADD BASE CPP /nologo /MTd /W3 /Gm /GX /ZI /Od /D "WIN32" /D
"_DEBUG" /D "_WINDOWS" /YX /FD /c
# ADD CPP /nologo /MTd /W3 /Gm /GX /ZI /Od /D "WIN32" /D "_DEBUG"
/D "_WINDOWS" /YX /FD /c
# ADD BASE MTL /nologo /D "_DEBUG" /mktyplib203 /o "NUL" /win32
# ADD MTL /nologo /D "_DEBUG" /mktyplib203 /o "NUL" /win32
# ADD BASE RSC /I 0x409 /d "_DEBUG"
# ADD RSC /I 0x409 /d "_DEBUG"
BSC32=bscmake.exe
# ADD BASE BSC32 /nologo
# ADD BSC32 /nologo
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 odbcc32.lib
odbccp32.lib /nologo /subsystem:windows /dll /debug /machine:I386
/pdbtype:sept
# ADD LINK32 ..\db_dblib_dll\bin\tpcc_dblib.lib
..\db_odbc_dll\bin\tpcc_odbc.lib kernel32.lib user32.lib gdi32.lib winspool.lib
comdlg32.lib advapi32.lib shell32.lib ole32.lib oleaut32.lib uuid.lib odbcc32.lib
odbccp32.lib /nologo /subsystem:windows /dll /debug /machine:I386
/pdbtype:sept

!ENDIF

# Begin Target

# Name "tpcc_com_all - Win32 Release"
# Name "tpcc_com_all - Win32 Debug"
# Begin Group "Source"

# PROP Default_Filter "*.cpp, *.c"
# Begin Source File

SOURCE=.\src\tpcc_com_all.cpp
# SUBTRACT CPP /YX
# End Source File
# Begin Source File

SOURCE=.\src\tpcc_com_all.def
# End Source File
# Begin Source File

```

```

SOURCE=.\src\tpcc_com_all.idl

!IF "$(CFG)" == "tpcc_com_all - Win32 Release"

# PROP Ignore_Default_Tool 1
# Begin Custom Build - Performing MIDL step
InputPath=.\src\tpcc_com_all.idl

BuildCmds= \
    midl /Oicf /h "tpcc_com_all.h" /iid "tpcc_com_all_i.c"
".\src\tpcc_com_all.idl" /out ".\src"

".\src\tpcc_com_all.tlb" : $(SOURCE) "$(INTDIR)" "$(OUTDIR)"
$(BuildCmds)

".\src\tpcc_com_all.h" : $(SOURCE) "$(INTDIR)" "$(OUTDIR)"
$(BuildCmds)

".\src\tpcc_com_all_i.c" : $(SOURCE) "$(INTDIR)" "$(OUTDIR)"
$(BuildCmds)
# End Custom Build

!ELSEIF "$(CFG)" == "tpcc_com_all - Win32 Debug"

# PROP Ignore_Default_Tool 1
# Begin Custom Build - Performing MIDL step
InputPath=.\src\tpcc_com_all.idl

BuildCmds= \
    midl /Oicf /h "tpcc_com_all.h" /iid "tpcc_com_all_i.c"
".\src\tpcc_com_all.idl" /out ".\src"

".\src\tpcc_com_all.tlb" : $(SOURCE) "$(INTDIR)" "$(OUTDIR)"
$(BuildCmds)

".\src\tpcc_com_all.h" : $(SOURCE) "$(INTDIR)" "$(OUTDIR)"
$(BuildCmds)

".\src\tpcc_com_all_i.c" : $(SOURCE) "$(INTDIR)" "$(OUTDIR)"
$(BuildCmds)
# End Custom Build

!ENDIF

# End Source File
# End Group
# Begin Group "Header"

# PROP Default_Filter "*.h"
# Begin Source File

SOURCE=.\src\Methods.h
# End Source File
# Begin Source File

SOURCE=.\src\resource.h
# End Source File
# End Group
# Begin Source File

SOURCE=.\src\tpcc_com_all.rc
# End Source File
# End Target
# End Project

```

***Tpcc\_com\_all.h***

```

#pragma warning( disable: 4049 ) /* more than 64k source lines */

/* this ALWAYS GENERATED file contains the definitions for the interfaces
*/

/* File created by MIDL compiler version 5.03.0280 */
/* at Thu Dec 13 23:13:14 2001
*/
/* Compiler settings for \src\tpcc_com_all.idl:
Oicf (OptLev=i2), W1, Zp8, env=Win32 (32b run), ms_ext, c_ext
error checks: allocation ref bounds_check enum stub_data
VC __declspec() decoration level:
__declspec(uuid()), __declspec(selectany), __declspec(novtable)
DECLSPEC_UUID(), MIDL_INTERFACE()
*/
//@@@MIDL_FILE_HEADING( )

/* 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__

/* 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__ */

#endif /* __tpcc_com_all_h__ */

#endif /* __Payment_FWD_DEFINED__ */
#define __Payment_FWD_DEFINED__

#ifdef __cplusplus
typedef class Payment Payment;
#else
typedef struct Payment Payment;
#endif /* __cplusplus */

#endif /* __Payment_FWD_DEFINED__ */

#ifndef __StockLevel_FWD_DEFINED__
#define __StockLevel_FWD_DEFINED__

#ifdef __cplusplus
typedef class StockLevel StockLevel;
#else
typedef struct StockLevel StockLevel;
#endif /* __cplusplus */

#endif /* __StockLevel_FWD_DEFINED__ */

/* header files for imported files */
#include "oaidl.h"
#include "ocidl.h"
#include "tpcc_com_ps.h"

#ifdef __cplusplus
extern "C" {
#endif

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;

#ifndef __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
#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 "oaidl.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"

////////////////////////////////////
#undef APSTUDIO_READONLY_SYMBOLS

////////////////////////////////////
// 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

#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
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_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
//

STRINGTABLE DISCARDABLE
BEGIN
    IDS_PROJNAME     "tpcc_com_all"
END

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

#ifdef APSTUDIO_INVOKED
////////////////////////////////////
//
// Generated from the TEXTINCLUDE 3 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

```

#pragma warning( disable: 4049 ) /* more than 64k source lines */

/* this ALWAYS GENERATED file contains the IIDs and CLSIDs */

/* link this file in with the server and any clients */

/* File created by MIDL compiler version 5.03.0280 */
/* at Thu Dec 13 23:13:14 2001
*/
/* Compiler settings for \src\tpcc_com_all.idl:
Oicf (OptLev=i2), W1, Zp8, env=Win32 (32b run), ms_ext, c_ext
error checks: allocation ref bounds_check enum stub_data
VC __declspec() decoration level:
__declspec(uuid()), __declspec(selectany), __declspec(novtable)
DECLSPEC_UUID(), MIDL_INTERFACE()
*/
//@@@MIDL_FILE_HEADING( )

#if !defined(_M_IA64) && !defined(_M_AXP64)

#ifdef __cplusplus
extern "C" {
#endif

#include <rpc.h>
#include <rpcndr.h>

#ifdef _MIDL_USE_GUIDDEF_

#ifndef INITGUID
#define INITGUID
#include <guiddef.h>
#undef INITGUID
#else
#include <guiddef.h>
#endif

#define MIDL_DEFINE_GUID(type,name,l,w1,w2,b1,b2,b3,b4,b5,b6,b7,b8) \
    DEFINE_GUID(name,l,w1,w2,b1,b2,b3,b4,b5,b6,b7,b8)

#else // !_MIDL_USE_GUIDDEF_

#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

#define MIDL_DEFINE_GUID(type,name,l,w1,w2,b1,b2,b3,b4,b5,b6,b7,b8) \
    const type name = {l,w1,w2,{b1,b2,b3,b4,b5,b6,b7,b8}}

#endif // !_MIDL_USE_GUIDDEF_

MIDL_DEFINE_GUID(IID,
LIBID_TPCCLib,0x122A3117,0x2520,0x11D3,0xBA,0x71,0x00,0xC0,0x4F,0x
BF,0xE0,0x8B);

MIDL_DEFINE_GUID(CLSID,
CLSID_TPCC,0x122A3128,0x2520,0x11D3,0xBA,0x71,0x00,0xC0,0x4F,0xB
F,0xE0,0x8B);

MIDL_DEFINE_GUID(CLSID,
CLSID_NewOrder,0x975BAABF,0x84A7,0x11D2,0xBA,0x47,0x00,0xC0,0x4
F,0xBF,0xE0,0x8B);

MIDL_DEFINE_GUID(CLSID,
CLSID_OrderStatus,0x266836AD,0xA50D,0x11D2,0xBA,0x4E,0x00,0xC0,0x
4F,0xBF,0xE0,0x8B);

MIDL_DEFINE_GUID(CLSID,
CLSID_Payment,0xCD02F7EF,0xA4FA,0x11D2,0xBA,0x4E,0x00,0xC0,0x4F
,0xBF,0xE0,0x8B);

MIDL_DEFINE_GUID(CLSID,
CLSID_StockLevel,0x2668369E,0xA50D,0x11D2,0xBA,0x4E,0x00,0xC0,0x4
F,0xBF,0xE0,0x8B);

#undef MIDL_DEFINE_GUID

#ifdef __cplusplus
}
#endif

#endif /* !defined(_M_IA64) && !defined(_M_AXP64)*/

#pragma warning( disable: 4049 ) /* more than 64k source lines */

/* this ALWAYS GENERATED file contains the IIDs and CLSIDs */

/* link this file in with the server and any clients */

/* File created by MIDL compiler version 5.03.0280 */
/* at Thu Dec 13 23:13:14 2001
*/
/* Compiler settings for \src\tpcc_com_all.idl:
Oicf (OptLev=i2), W1, Zp8, env=Win64 (32b run,appending), ms_ext, c_ext,
robust
error checks: allocation ref bounds_check enum stub_data
VC __declspec() decoration level:

```

```

    __declspec(uuid()), __declspec(selectany), __declspec(novtable)
    DECLSPEC_UUID(), MIDL_INTERFACE()
*/
//@@MIDL_FILE_HEADING( )

#if defined(_M_IA64) || defined(_M_AXP64)

#ifdef __cplusplus
extern "C" {
#endif

#include <rpc.h>
#include <rpcndr.h>

#ifdef _MIDL_USE_GUIDDEF_

#ifndef INITGUID
#define INITGUID
#include <guiddef.h>
#undef INITGUID
#else
#include <guiddef.h>
#endif

#define MIDL_DEFINE_GUID(type,name,l,w1,w2,b1,b2,b3,b4,b5,b6,b7,b8) \
    DEFINE_GUID(name,l,w1,w2,b1,b2,b3,b4,b5,b6,b7,b8)

#else /* !_MIDL_USE_GUIDDEF_

#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

#define MIDL_DEFINE_GUID(type,name,l,w1,w2,b1,b2,b3,b4,b5,b6,b7,b8) \
    const type name = {l,w1,w2,{b1,b2,b3,b4,b5,b6,b7,b8}}

#endif /* !_MIDL_USE_GUIDDEF_

MIDL_DEFINE_GUID(IID,
LIBID_TPCClib,0x122A3117,0x2520,0x11D3,0xBA,0x71,0x00,0xC0,0x4F,0
xBF,0xE0,0x8B);

MIDL_DEFINE_GUID(CLSID,
CLSID_TPCC,0x122A3128,0x2520,0x11D3,0xBA,0x71,0x00,0xC0,0x4F,0xB
F,0xE0,0x8B);

MIDL_DEFINE_GUID(CLSID,
CLSID_NewOrder,0x975BAABF,0x84A7,0x11D2,0xBA,0x47,0x00,0xC0,0x4
F,0xBF,0xE0,0x8B);

```

```

MIDL_DEFINE_GUID(CLSID,
CLSID_OrderStatus,0x266836AD,0xA50D,0x11D2,0xBA,0x4E,0x00,0xC0,0x
4F,0xBF,0xE0,0x8B);

MIDL_DEFINE_GUID(CLSID,
CLSID_Payment,0xCD02F7EF,0xA4FA,0x11D2,0xBA,0x4E,0x00,0xC0,0x4F
,0xBF,0xE0,0x8B);

MIDL_DEFINE_GUID(CLSID,
CLSID_StockLevel,0x2668369E,0xA50D,0x11D2,0xBA,0x4E,0x00,0xC0,0x4
F,0xBF,0xE0,0x8B);

#undef MIDL_DEFINE_GUID

#ifdef __cplusplus
}
#endif

#endif /* defined(_M_IA64) || defined(_M_AXP64) */

```

## ***Tpcc\_com\_all\_resource.h***

```

//{{NO_DEPENDENCIES}}
// Microsoft Developer Studio generated include file.
// Used by tpcc_com_all.rc
//
#define IDS_PROJNAME            100
#define IDR_TPCC                101
#define IDR_NEWORDER            102
#define IDR_ORDERSTATUS        103
#define IDR_PAYMENT             104
#define IDR_STOCKLEVEL         105

// Next default values for new objects
//
#ifdef APSTUDIO_INVOKED
#ifdef 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         106
#endif
#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 'Both'
    }
}

```

### 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 'Both'
    }
}

```

### 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 'Both'
    }
}

```

### 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.dsp

```

# Microsoft Developer Studio Project File - Name="tpcc_com_ps" - Package
Owner=<<4>
# Microsoft Developer Studio Generated Build File, Format Version 6.00
# ** DO NOT EDIT **

# TARGETTYPE "Win32 (x86) Application" 0x0101

CFG=tpcc_com_ps - Win32 Debug
!MESSAGE This is not a valid makefile. To build this project using NMAKE,
!MESSAGE use the Export Makefile command and run
!MESSAGE
!MESSAGE NMAKE /f "tpcc_com_ps.mak".
!MESSAGE
!MESSAGE You can specify a configuration when running NMAKE
!MESSAGE by defining the macro CFG on the command line. For example:
!MESSAGE
!MESSAGE NMAKE /f "tpcc_com_ps.mak" CFG="tpcc_com_ps - Win32
Debug"
!MESSAGE
!MESSAGE Possible choices for configuration are:
!MESSAGE
!MESSAGE "tpcc_com_ps - Win32 Release" (based on "Win32 (x86)
Application")
!MESSAGE "tpcc_com_ps - Win32 Debug" (based on "Win32 (x86)
Application")
!MESSAGE

# Begin Project
# PROP AllowPerConfigDependencies 0
# PROP Scc_ProjName ""
# PROP Scc_LocalPath ""
CPP=cl.exe
MTL=midl.exe
RSC=rc.exe

!IF "$(CFG)" == "tpcc_com_ps - 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 ".\obj"
# PROP Ignore_Export_Lib 0
# PROP Target_Dir ""

```

```

# ADD BASE CPP /nologo /W3 /GX /O2 /D "WIN32" /D "NDEBUG" /D
" _WINDOWS" /YX /FD /c
# ADD CPP /nologo /W3 /GX /O2 /D "WIN32" /D "NDEBUG" /D
_WIN32_WINNT=0x0400 /D "REGISTER_PROXY_DLL" /FD /c
# SUBTRACT CPP /YX
# ADD BASE MTL /nologo /D "NDEBUG" /mktyplib203 /o "NUL" /win32
# ADD MTL /nologo /D "NDEBUG" /mktyplib203 /o "NUL" /win32
# ADD BASE RSC /I 0x409 /d "NDEBUG"
# ADD RSC /I 0x409 /d "NDEBUG"
BSC32=bscmake.exe
# ADD BASE BSC32 /nologo
# ADD BSC32 /nologo
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 odbccp32.lib
odbccp32.lib /nologo /subsystem:windows /machine:I386
# ADD LINK32 kernel32.lib rpcndr.lib rpcns4.lib rpert4.lib oleaut32.lib
uuid.lib /nologo /entry:"DllMain" /subsystem:windows /dll /pdb:none
/machine:I386 /def:"\src\tpcc_com_ps.def"
# Begin Custom Build - Copying tpcc_com_ps.h
InputPath=\bin\tpcc_com_ps.dll
SOURCE="$(InputPath)"

..\tpcc_com_all\src\tpcc_com_ps.h" : $(SOURCE) "$(INTDIR)" "$(OUTDIR)"
copy \src\tpcc_com_ps.h ..\tpcc_com_all\src\

# End Custom Build

!ELSEIF "$(CFG)" == "tpcc_com_ps - 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 ".\obj"
# PROP Ignore_Export_Lib 0
# PROP Target_Dir ""
# ADD BASE CPP /nologo /W3 /Gm /GX /Zi /Od /D "WIN32" /D " _DEBUG"
/D " _WINDOWS" /YX /FD /c
# ADD CPP /nologo /ZI /Od /D "WIN32" /D " _DEBUG" /D
_WIN32_WINNT=0x0400 /D "REGISTER_PROXY_DLL" /FD /c
# ADD BASE MTL /nologo /D " _DEBUG" /mktyplib203 /o "NUL" /win32
# ADD MTL /nologo /D " _DEBUG" /mktyplib203 /o "NUL" /win32
# ADD BASE RSC /I 0x409 /d " _DEBUG"
# ADD RSC /I 0x409 /d " _DEBUG"
BSC32=bscmake.exe
# ADD BASE BSC32 /nologo
# ADD BSC32 /nologo
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 odbccp32.lib
odbccp32.lib /nologo /subsystem:windows /debug /machine:I386 /pdbtype:sept
# ADD LINK32 kernel32.lib rpcndr.lib rpcns4.lib rpert4.lib oleaut32.lib
uuid.lib /nologo /entry:"DllMain" /dll /debug /machine:IX86
/def:"\src\tpcc_com_ps.def" /pdbtype:sept
# SUBTRACT LINK32 /pdb:none
# Begin Custom Build - Copying tpcc_com_ps.h
InputPath=\bin\tpcc_com_ps.dll
SOURCE="$(InputPath)"

..\tpcc_com_all\src\tpcc_com_ps.h" : $(SOURCE) "$(INTDIR)" "$(OUTDIR)"
copy \src\tpcc_com_ps.h ..\tpcc_com_all\src\

# End Custom Build

```

```

!ENDIF
# Begin Target
# Name "tpcc_com_ps - Win32 Release"
# Name "tpcc_com_ps - Win32 Debug"
# Begin Group "Source"
# PROP Default_Filter ""
# Begin Source File
SOURCE=\src\dll\data.c
# End Source File
# Begin Source File
SOURCE=\src\tpcc_com_ps.def
# PROP Exclude_From_Build 1
# End Source File
# Begin Source File
SOURCE=\src\tpcc_com_ps.idl
!IF "$(CFG)" == "tpcc_com_ps - Win32 Release"
# PROP Ignore_Default_Tool 1
# Begin Custom Build
InputPath=\src\tpcc_com_ps.idl
BuildCmds= \
midl /Oicf /h "tpcc_com_ps.h" /iid "tpcc_com_ps_i.c"
"\src\tpcc_com_ps.idl" /out ".\src"
"\src\tpcc_com_ps.h" : $(SOURCE) "$(INTDIR)" "$(OUTDIR)"
$(BuildCmds)
"\src\tpcc_com_ps_i.c" : $(SOURCE) "$(INTDIR)" "$(OUTDIR)"
$(BuildCmds)
"\src\dll\data.c" : $(SOURCE) "$(INTDIR)" "$(OUTDIR)"
$(BuildCmds)
"\src\tpcc_com_ps.p.c" : $(SOURCE) "$(INTDIR)" "$(OUTDIR)"
$(BuildCmds)
# End Custom Build
!ELSEIF "$(CFG)" == "tpcc_com_ps - Win32 Debug"
# PROP Ignore_Default_Tool 1
# Begin Custom Build
InputPath=\src\tpcc_com_ps.idl
BuildCmds= \
midl /Oicf /h "tpcc_com_ps.h" /iid "tpcc_com_ps_i.c"
"\src\tpcc_com_ps.idl" /out ".\src"
"\src\tpcc_com_ps.h" : $(SOURCE) "$(INTDIR)" "$(OUTDIR)"
$(BuildCmds)
"\src\tpcc_com_ps_i.c" : $(SOURCE) "$(INTDIR)" "$(OUTDIR)"
$(BuildCmds)
"\src\dll\data.c" : $(SOURCE) "$(INTDIR)" "$(OUTDIR)"
$(BuildCmds)
"\src\tpcc_com_ps.p.c" : $(SOURCE) "$(INTDIR)" "$(OUTDIR)"
$(BuildCmds)
# End Custom Build

```

```
!ENDIF
```

```
# End Source File  
# Begin Source File
```

```
SOURCE=.\src\tpcc_com_ps_i.c  
# End Source File  
# Begin Source File
```

```
SOURCE=.\src\tpcc_com_ps_p.c  
# End Source File  
# End Group  
# End Target  
# End Project
```

## ***Tpcc\_com\_ps.h***

```
#pragma warning( disable: 4049 ) /* more than 64k source lines */
```

```
/* this ALWAYS GENERATED file contains the definitions for the interfaces  
*/
```

```
/* File created by MIDL compiler version 5.03.0280 */  
/* at Thu Dec 13 23:13:08 2001  
*/
```

```
/* Compiler settings for \src\tpcc_com_ps.idl:  
Oicf (OptLev=i2), W1, Zp8, env=Win32 (32b run), ms_ext, c_ext  
error checks: allocation ref bounds_check enum stub_data  
VC __declspec() decoration level:  
__declspec(uuid()), __declspec(selectany), __declspec(novtable)  
DECLSPEC_UUID(), MIDL_INTERFACE()  
*/
```

```
//@@MIDL_FILE_HEADING( )
```

```
/* 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__
```

```
/* Forward Declarations */
```

```
#ifndef __ITPCC_FWD_DEFINED__  
#define __ITPCC_FWD_DEFINED__  
typedef interface ITPCC ITPCC;  
#endif /* __ITPCC_FWD_DEFINED__ */
```

```
/* header files for imported files */  
#include "oaidl.h"  
#include "ocidl.h"
```

```
#ifndef __cplusplus  
extern "C"{  
#endif
```

```
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 STDMETHODCALLTYPE NewOrder(  
/* [in] */ VARIANT txn_in,  
/* [out] */ VARIANT __RPC_FAR *txn_out) = 0;
```

```
virtual HRESULT STDMETHODCALLTYPE Payment(  
/* [in] */ VARIANT txn_in,  
/* [out] */ VARIANT __RPC_FAR *txn_out) = 0;
```

```
virtual HRESULT STDMETHODCALLTYPE Delivery(  
/* [in] */ VARIANT txn_in,  
/* [out] */ VARIANT __RPC_FAR *txn_out) = 0;
```

```
virtual HRESULT STDMETHODCALLTYPE StockLevel(  
/* [in] */ VARIANT txn_in,  
/* [out] */ VARIANT __RPC_FAR *txn_out) = 0;
```

```
virtual HRESULT STDMETHODCALLTYPE OrderStatus(  
/* [in] */ VARIANT txn_in,  
/* [out] */ VARIANT __RPC_FAR *txn_out) = 0;
```

```
virtual HRESULT STDMETHODCALLTYPE 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,
    /* [out] */ VARIANT __RPC_FAR *txn_out);

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,txn_out) \
    (This)->lpVtbl -> Delivery(This,txn_in,txn_out)

#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,
    /* [out] */ VARIANT __RPC_FAR *txn_out);

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 STDMETHODCALLTYPE 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);

#endif /* __ITPCC_INTERFACE_DEFINED__ */

/* Additional Prototypes for ALL interfaces */

unsigned long __RPC_USER VARIANT_UserSize( unsigned long
__RPC_FAR *, unsigned long , VARIANT __RPC_FAR * );
unsigned char __RPC_FAR * __RPC_USER VARIANT_UserMarshal(
unsigned long __RPC_FAR *, unsigned char __RPC_FAR *, VARIANT
__RPC_FAR * );
unsigned char __RPC_FAR * __RPC_USER
VARIANT_UserUnmarshal(unsigned long __RPC_FAR *, unsigned char
__RPC_FAR *, VARIANT __RPC_FAR * );
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 "oaidl.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,
    [out]
    VARIANT *txn_out
    );

    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

#pragma warning( disable: 4049 ) /* more than 64k source lines */

/* this ALWAYS GENERATED file contains the IIDs and CLSIDs */

/* link this file in with the server and any clients */

/* File created by MIDL compiler version 5.03.0280 */
/* at Thu Dec 13 23:13:08 2001
*/
/* Compiler settings for .\src\tpcc_com_ps.idl:
Oicf (OptLev=i2), W1, Zp8, env=Win32 (32b run), ms_ext, c_ext
error checks: allocation ref bounds_check enum stub_data
VC __declspec() decoration level:
__declspec(uuid()), __declspec(selectany), __declspec(novtable)
DECLSPEC_UUID(), MIDL_INTERFACE()
*/

```

```

//@@@MIDL_FILE_HEADING( )

#if !defined(_M_IA64) && !defined(_M_AXP64)

#ifdef __cplusplus
extern "C"{
#endif

#include <rpc.h>
#include <rpcndr.h>

#ifdef _MIDL_USE_GUIDDEF_

#ifndef INITGUID
#define INITGUID
#include <guiddef.h>
#undef INITGUID
#else
#include <guiddef.h>
#endif

#define MIDL_DEFINE_GUID(type,name,l,w1,w2,b1,b2,b3,b4,b5,b6,b7,b8) \
    DEFINE_GUID(name,l,w1,w2,b1,b2,b3,b4,b5,b6,b7,b8)

#else // !_MIDL_USE_GUIDDEF_

#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

#define MIDL_DEFINE_GUID(type,name,l,w1,w2,b1,b2,b3,b4,b5,b6,b7,b8) \
    const type name = {l,w1,w2,{b1,b2,b3,b4,b5,b6,b7,b8}}

#endif !_MIDL_USE_GUIDDEF_

MIDL_DEFINE_GUID(IID,
IID_ITPCC,0xFEEE6AA2,0x84B1,0x11d2,0xBA,0x47,0x00,0xC0,0x4F,0xBF,
0xE0,0x8B);

#undef MIDL_DEFINE_GUID

#ifdef __cplusplus
}
#endif

#endif /* !defined(_M_IA64) && !defined(_M_AXP64)*/

#pragma warning( disable: 4049 ) /* more than 64k source lines */

/* this ALWAYS GENERATED file contains the IIDs and CLSIDs */

/* link this file in with the server and any clients */

```

```

/* File created by MIDL compiler version 5.03.0280 */
/* at Thu Dec 13 23:13:08 2001
*/
/* Compiler settings for .\src\tpcc_com_ps.idl:
Oicf (OptLev=i2), W1, Zp8, env=Win64 (32b run,appending), ms_ext, c_ext,
robust
error checks: allocation ref bounds_check enum stub_data
VC __declspec() decoration level:
__declspec(uuid()), __declspec(selectany), __declspec(novtable)
DECLSPEC_UUID(), MIDL_INTERFACE()
*/
//@@@MIDL_FILE_HEADING( )

#if defined(_M_IA64) || defined(_M_AXP64)

#ifdef __cplusplus
extern "C"{
#endif

#include <rpc.h>
#include <rpcndr.h>

#ifdef _MIDL_USE_GUIDDEF_

#ifndef INITGUID
#define INITGUID
#include <guiddef.h>
#undef INITGUID
#else
#include <guiddef.h>
#endif

#define MIDL_DEFINE_GUID(type,name,l,w1,w2,b1,b2,b3,b4,b5,b6,b7,b8) \
    DEFINE_GUID(name,l,w1,w2,b1,b2,b3,b4,b5,b6,b7,b8)

#else // !_MIDL_USE_GUIDDEF_

#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

#define MIDL_DEFINE_GUID(type,name,l,w1,w2,b1,b2,b3,b4,b5,b6,b7,b8) \
    const type name = {l,w1,w2,{b1,b2,b3,b4,b5,b6,b7,b8}}

#endif !_MIDL_USE_GUIDDEF_

MIDL_DEFINE_GUID(IID,
IID_ITPCC,0xFEEE6AA2,0x84B1,0x11d2,0xBA,0x47,0x00,0xC0,0x4F,0xBF,
0xE0,0x8B);

#undef MIDL_DEFINE_GUID

```



```

#ifdef __cplusplus
}
#endif

#ifdef defined(_M_IA64) || defined(_M_AXP64)*/

Tpcc_com_ps_p.c

#pragma warning( disable: 4049 ) /* more than 64k source lines */

/* this ALWAYS GENERATED file contains the proxy stub code */

/* File created by MIDL compiler version 5.03.0280 */
/* at Thu Dec 13 23:13:08 2001
*/
/* Compiler settings for \src\tpcc_com_ps.idl:
Oicf (OptLev=i2), W1, Zp8, env=Win32 (32b run), ms_ext, c_ext
error checks: allocation ref bounds_check enum stub_data
VC __declspec() decoration level:
__declspec(uuid()), __declspec(selectany), __declspec(novtable)
DECLSPEC_UUID(), MIDL_INTERFACE()
*/
//@@@MIDL_FILE_HEADING( )

#ifdef !defined(_M_IA64) && !defined(_M_AXP64)
#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 193
#define TRANSMIT_AS_TABLE_SIZE 0
#define WIRE_MARSHAL_TABLE_SIZE 1

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,{0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00}} */

/* Object interface: IUnknown, ver. 0.0,

GUID={0x00000000,0x0000,0x0000,{0xC0,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x46}} */

/* Object interface: ITPCC, ver. 0.0,

GUID={0xFEE6AA2,0x84B1,0x11d2,{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")
static const unsigned short ITPCC_FormatStringOffsetTable[] =
{
    0,
    34,
    68,
    102,
    136,
    170
};

static const MIDL_SERVER_INFO ITPCC_ServerInfo =
{
    &Object_StubDesc,
    0,
    __MIDL_ProcFormatString.Format,
    &ITPCC_FormatStringOffsetTable[-3],
    0,
    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
};

extern const USER_MARSHAL_ROUTINE_QUADRUPLE
UserMarshalRoutines[ WIRE_MARSHAL_TABLE_SIZE ];

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,
    0x5030118, /* MIDL Version 5.3.280 */
    0,
    UserMarshalRoutines,
    0, /* notify & notify_flag routine table */
    0x1, /* MIDL flag */
    0, /* Reserved3 */
    0, /* Reserved4 */
    0 /* Reserved5 */
};

#pragma data_seg(".rdata")

static const USER_MARSHAL_ROUTINE_QUADRUPLE
UserMarshalRoutines[ WIRE_MARSHAL_TABLE_SIZE ] =
{
    {
        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_marshal] or [user_marshal] attribute.
#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, /*
        Oi2 */
        0x6c, /* Old Flags: object,
        /* 2 */ NdrFcLong( 0x0 ), /* 0 */
        /* 6 */ NdrFcShort( 0x3 ), /* 3 */
        #ifndef _ALPHA_
        #ifndef _PPC_
        #if !defined(_MIPS_)
        /* 8 */ NdrFcShort( 0x1c ), /* x86 Stack size/offset = 28 */
        #else
        NdrFcShort( 0x20 ), /* MIPS Stack
        size/offset = 32 */
        #endif
        #endif
        #else
        NdrFcShort( 0x20 ), /* 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_
        #ifndef _PPC_
        #if !defined(_MIPS_)
        /* 18 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
        #else
        NdrFcShort( 0x8 ), /* MIPS Stack
        size/offset = 8 */
        #endif
        #endif
        #else
        NdrFcShort( 0x8 ), /* PPC Stack size/offset
        = 8 */
        #endif
        #else
        NdrFcShort( 0x8 ), /* Alpha Stack
        size/offset = 8 */
        #endif
        #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_
        #ifndef _PPC_
        #if !defined(_MIPS_)
        /* 24 */ NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
        #else
        NdrFcShort( 0x18 ), /* MIPS Stack
        size/offset = 24 */

```

```

#endif
#else
= 24 */
#endif
#else
size/offset = 24 */
#endif
/* 26 */ NdrFcShort( 0x3da ), /* Type Offset=986 */

/* Return value */

/* 28 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifndef _ALPHA_
#ifndef _PPC_
#if !defined(_MIPS_)
/* 30 */ NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
#else
size/offset = 28 */
#endif
#else
= 28 */
#endif
#else
NdrFcShort( 0x1c ), /* MIPS Stack
size/offset = 28 */
#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_
#ifndef _PPC_
#if !defined(_MIPS_)
/* 42 */ NdrFcShort( 0x1c ), /* x86 Stack size/offset = 28 */
#else
size/offset = 32 */
#endif
#else
= 32 */
#endif
#else
NdrFcShort( 0x20 ), /* MIPS Stack
size/offset = 32 */
#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_
#ifndef _PPC_
#if !defined(_MIPS_)
/* 52 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
#else
size/offset = 8 */
#endif
#else
= 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_
#ifndef _PPC_
#if !defined(_MIPS_)
/* 58 */ NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
#else
size/offset = 24 */
#endif
/* 60 */ NdrFcShort( 0x3da ), /* Type Offset=986 */

/* Return value */

/* 62 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifndef _ALPHA_
#ifndef _PPC_
#if !defined(_MIPS_)
/* 64 */ NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
#else
size/offset = 28 */
#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_
#ifndef _PPC_

```

```

#if !defined(_MIPS_)
/* 76 */ NdrFcShort( 0x1c ), /* x86 Stack size/offset = 28 */
#else
NdrFcShort( 0x20 ), /* MIPS Stack
size/offset = 32 */
#endif
#else
NdrFcShort( 0x20 ), /* PPC Stack size/offset
= 32 */
#endif
#else
NdrFcShort( 0x28 ), /* Alpha Stack
size/offset = 40 */
#endif
/* 78 */ NdrFcShort( 0x0 ), /* 0 */
/* 80 */ NdrFcShort( 0x8 ), /* 8 */
/* 82 */ 0x7, /* Oi2 Flags: srv must size, clt must size, has
return, */
0x3, /* 3 */
/* Parameter txn_in */
/* 84 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by val, */
#ifndef _ALPHA_
#ifndef _PPC_
#if !defined(_MIPS_)
/* 86 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
#else
NdrFcShort( 0x8 ), /* MIPS Stack
size/offset = 8 */
#endif
#else
NdrFcShort( 0x8 ), /* PPC Stack size/offset
= 8 */
#endif
#else
NdrFcShort( 0x8 ), /* Alpha Stack
size/offset = 8 */
#endif
/* 88 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */
/* Parameter txn_out */
/* 90 */ NdrFcShort( 0x4113 ), /* Flags: must size, must free, out,
simple ref, srv alloc size=16 */
#ifndef _ALPHA_
#ifndef _PPC_
#if !defined(_MIPS_)
/* 92 */ NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
#else
NdrFcShort( 0x18 ), /* MIPS Stack
size/offset = 24 */
#endif
#else
NdrFcShort( 0x18 ), /* PPC Stack size/offset
= 24 */
#endif
#else
NdrFcShort( 0x18 ), /* Alpha Stack
size/offset = 24 */
#endif
/* 94 */ NdrFcShort( 0x3da ), /* Type Offset=986 */
/* Return value */
/* 96 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifndef _ALPHA_
#ifndef _PPC_

```

```

#if !defined(_MIPS_)
/* 98 */ NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
#else
NdrFcShort( 0x1c ), /* MIPS Stack
size/offset = 28 */
#endif
#else
NdrFcShort( 0x1c ), /* PPC Stack size/offset
= 28 */
#endif
#else
NdrFcShort( 0x20 ), /* Alpha Stack
size/offset = 32 */
#endif
/* 100 */ 0x8, /* FC_LONG */
0x0, /* 0 */
/* Procedure StockLevel */
/* 102 */ 0x33, /* FC_AUTO_HANDLE */
0x6c, /* Old Flags: object,
Oi2 */
/* 104 */ NdrFcLong( 0x0 ), /* 0 */
/* 108 */ NdrFcShort( 0x6 ), /* 6 */
#ifndef _ALPHA_
#ifndef _PPC_
#if !defined(_MIPS_)
/* 110 */ NdrFcShort( 0x1c ), /* x86 Stack size/offset = 28 */
#else
NdrFcShort( 0x20 ), /* MIPS Stack
size/offset = 32 */
#endif
#else
NdrFcShort( 0x20 ), /* PPC Stack size/offset
= 32 */
#endif
#else
NdrFcShort( 0x28 ), /* Alpha Stack
size/offset = 40 */
#endif
/* 112 */ NdrFcShort( 0x0 ), /* 0 */
/* 114 */ NdrFcShort( 0x8 ), /* 8 */
/* 116 */ 0x7, /* Oi2 Flags: srv must size, clt must size, has
return, */
0x3, /* 3 */
/* Parameter txn_in */
/* 118 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by val, */
#ifndef _ALPHA_
#ifndef _PPC_
#if !defined(_MIPS_)
/* 120 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
#else
NdrFcShort( 0x8 ), /* MIPS Stack
size/offset = 8 */
#endif
#else
NdrFcShort( 0x8 ), /* PPC Stack size/offset
= 8 */
#endif
#else
NdrFcShort( 0x8 ), /* Alpha Stack
size/offset = 8 */
#endif
/* 122 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */
/* Parameter txn_out */

```

```

/* 124 */ NdrFcShort( 0x4113 ), /* Flags: must size, must free, out,
simple ref, srv alloc size=16 */
#ifdef _ALPHA_
#ifdef _PPC_
#if !defined( _MIPS_ )
/* 126 */ NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
#else
NdrFcShort( 0x18 ), /* MIPS Stack
size/offset = 24 */
#endif
#else
NdrFcShort( 0x18 ), /* PPC Stack size/offset
= 24 */
#endif
#else
NdrFcShort( 0x18 ), /* Alpha Stack
size/offset = 24 */
#endif
/* 128 */ NdrFcShort( 0x3da ), /* Type Offset=986 */

/* Return value */

/* 130 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifdef _ALPHA_
#ifdef _PPC_
#if !defined( _MIPS_ )
/* 132 */ NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
#else
NdrFcShort( 0x1c ), /* MIPS Stack
size/offset = 28 */
#endif
#else
NdrFcShort( 0x1c ), /* PPC Stack size/offset
= 28 */
#endif
#else
NdrFcShort( 0x20 ), /* Alpha Stack
size/offset = 32 */
#endif
/* 134 */ 0x8, /* FC_LONG */
0x0, /* 0 */

/* Procedure OrderStatus */

/* 136 */ 0x33, /* FC_AUTO_HANDLE */
0x6c, /* Old Flags: object,
Oi2 */
/* 138 */ NdrFcLong( 0x0 ), /* 0 */
/* 142 */ NdrFcShort( 0x7 ), /* 7 */
#ifdef _ALPHA_
#ifdef _PPC_
#if !defined( _MIPS_ )
/* 144 */ NdrFcShort( 0x1c ), /* x86 Stack size/offset = 28 */
#else
NdrFcShort( 0x20 ), /* MIPS Stack
size/offset = 32 */
#endif
#else
NdrFcShort( 0x20 ), /* PPC Stack size/offset
= 32 */
#endif
#else
NdrFcShort( 0x28 ), /* Alpha Stack
size/offset = 40 */
#endif
/* 146 */ NdrFcShort( 0x0 ), /* 0 */
/* 148 */ NdrFcShort( 0x8 ), /* 8 */

/* 150 */ 0x7, /* Oi2 Flags: srv must size, clt must size, has
return, */
0x3, /* 3 */

/* Parameter txn_in */

/* 152 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by val, */
#ifdef _ALPHA_
#ifdef _PPC_
#if !defined( _MIPS_ )
/* 154 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
#else
NdrFcShort( 0x8 ), /* MIPS Stack
size/offset = 8 */
#endif
#else
NdrFcShort( 0x8 ), /* PPC Stack size/offset
= 8 */
#endif
#else
NdrFcShort( 0x8 ), /* Alpha Stack
size/offset = 8 */
#endif
/* 156 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */

/* Parameter txn_out */

/* 158 */ NdrFcShort( 0x4113 ), /* Flags: must size, must free, out,
simple ref, srv alloc size=16 */
#ifdef _ALPHA_
#ifdef _PPC_
#if !defined( _MIPS_ )
/* 160 */ NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
#else
NdrFcShort( 0x18 ), /* MIPS Stack
size/offset = 24 */
#endif
#else
NdrFcShort( 0x18 ), /* PPC Stack size/offset
= 24 */
#endif
#else
NdrFcShort( 0x20 ), /* Alpha Stack
size/offset = 24 */
#endif
/* 162 */ NdrFcShort( 0x3da ), /* Type Offset=986 */

/* Return value */

/* 164 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifdef _ALPHA_
#ifdef _PPC_
#if !defined( _MIPS_ )
/* 166 */ NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
#else
NdrFcShort( 0x1c ), /* MIPS Stack
size/offset = 28 */
#endif
#else
NdrFcShort( 0x1c ), /* PPC Stack size/offset
= 28 */
#endif
#else
NdrFcShort( 0x20 ), /* Alpha Stack
size/offset = 32 */
#endif
/* 168 */ 0x8, /* FC_LONG */
0x0, /* 0 */

```

```

/* Procedure CallSetComplete */
/* 170 */ 0x33, /* FC_AUTO_HANDLE */
           0x6c, /* Old Flags: object,
Oi2 */
/* 172 */ NdrFcLong( 0x0 ), /* 0 */
/* 176 */ NdrFcShort( 0x8 ), /* 8 */
#ifdef _ALPHA_
/* 178 */ NdrFcShort( 0x8 ), /* x86, MIPS, PPC Stack size/offset = 8 */
#else
           NdrFcShort( 0x10 ), /* Alpha Stack
size/offset = 16 */
#endif
/* 180 */ NdrFcShort( 0x0 ), /* 0 */
/* 182 */ NdrFcShort( 0x8 ), /* 8 */
/* 184 */ 0x4, /* Oi2 Flags: has return, */
           0x1, /* 1 */

/* Return value */

/* 186 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifdef _ALPHA_
/* 188 */ NdrFcShort( 0x4 ), /* x86, MIPS, PPC Stack size/offset = 4 */
#else
           NdrFcShort( 0x8 ), /* Alpha Stack
size/offset = 8 */
#endif
/* 190 */ 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( 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, /*
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 */
0x0, /* 0 */
/* 338 */ 0x0, /* 0 */
0x0, /* 0 */
/* 340 */ 0x0, /* 0 */
0x0, /* 0 */
/* 342 */ 0x0, /* 0 */
0x46, /* 70 */
/* 344 */
0x12, 0x10, /* FC_UP
[pointer_deref] */
/* 346 */ NdrFcShort( 0x2 ), /* Offset= 2 (348) */
/* 348 */
0x12, 0x0, /* FC_UP */
/* 350 */ NdrFcShort( 0x1fc ), /* Offset= 508 (858) */
/* 352 */
0x2a, /*
FC_ENCAPSULATED_UNION */
0x49, /* 73 */
/* 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( 0xffffffff ), /* Offset= -1 (417) */
/* 420 */
0x1b, /* FC_CARRAY */
0x3, /* 3 */
/* 422 */ NdrFcShort( 0x4 ), /* 4 */
/* 424 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
0x0, /* */
/* 426 */ NdrFcShort( 0x0 ), /* 0 */
/* 428 */
0x4b, /* FC_PP */
0x5c, /* FC_PAD */
/* 430 */
0x48, /*
FC_VARIABLE_REPEAT */
0x49, /*
FC_FIXED_OFFSET */
/* 432 */ NdrFcShort( 0x4 ), /* 4 */
/* 434 */ NdrFcShort( 0x0 ), /* 0 */
/* 436 */ NdrFcShort( 0x1 ), /* 1 */
/* 438 */ NdrFcShort( 0x0 ), /* 0 */
/* 440 */ NdrFcShort( 0x0 ), /* 0 */
/* 442 */ 0x12, 0x0, /* FC_UP */
/* 444 */ NdrFcShort( 0xffffffff6e ), /* Offset= -146 (298) */
/* 446 */
0x5b, /* FC_END */
0x8, /* FC_LONG */
/* 448 */ 0x5c, /* FC_PAD */
0x5b, /* FC_END */
/* 450 */
0x16, /* FC_PSTRUCT */
0x3, /* 3 */
/* 452 */ NdrFcShort( 0x8 ), /* 8 */
/* 454 */
0x4b, /* FC_PP */
0x5c, /* FC_PAD */
/* 456 */
0x46, /* FC_NO_REPEAT */
0x5c, /* FC_PAD */
/* 458 */ NdrFcShort( 0x4 ), /* 4 */

```

```

/* 460 */ NdrFcShort( 0x4 ), /* 4 */
/* 462 */ 0x11, 0x0, /* FC_RP */
/* 464 */ NdrFcShort( 0xfffffd4 ), /* Offset= -44 (420) */
/* 466 */
                                0x5b, /* FC_END */
                                0x8, /* FC_LONG */
/* 468 */ 0x8, /* FC_LONG */
                                0x5b, /* FC_END */
/* 470 */
                                0x21, /*
FC_BOGUS_ARRAY */
                                0x3, /* 3 */
/* 472 */ NdrFcShort( 0x0 ), /* 0 */
/* 474 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
                                0x0, /* */
/* 476 */ NdrFcShort( 0x0 ), /* 0 */
/* 478 */ NdrFcLong( 0xffffffff ), /* -1 */
/* 482 */ 0x4c, /* FC_EMBEDDED_COMPLEX */
                                0x0, /* 0 */
/* 484 */ NdrFcShort( 0xfffff50 ), /* Offset= -176 (308) */
/* 486 */ 0x5c, /* FC_PAD */
                                0x5b, /* FC_END */
/* 488 */
                                0x1a, /*
FC_BOGUS_STRUCT */
                                0x3, /* 3 */
/* 490 */ NdrFcShort( 0x8 ), /* 8 */
/* 492 */ NdrFcShort( 0x0 ), /* 0 */
/* 494 */ NdrFcShort( 0x6 ), /* Offset= 6 (500) */
/* 496 */ 0x8, /* FC_LONG */
                                0x36, /* FC_POINTER */
/* 498 */ 0x5c, /* FC_PAD */
                                0x5b, /* FC_END */
/* 500 */
                                0x11, 0x0, /* FC_RP */
/* 502 */ NdrFcShort( 0xfffffe0 ), /* Offset= -32 (470) */
/* 504 */
                                0x21, /*
FC_BOGUS_ARRAY */
                                0x3, /* 3 */
/* 506 */ NdrFcShort( 0x0 ), /* 0 */
/* 508 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
                                0x0, /* */
/* 510 */ NdrFcShort( 0x0 ), /* 0 */
/* 512 */ NdrFcLong( 0xffffffff ), /* -1 */
/* 516 */ 0x4c, /* FC_EMBEDDED_COMPLEX */
                                0x0, /* 0 */
/* 518 */ NdrFcShort( 0xfffff40 ), /* Offset= -192 (326) */
/* 520 */ 0x5c, /* FC_PAD */
                                0x5b, /* FC_END */
/* 522 */
                                0x1a, /*
FC_BOGUS_STRUCT */
                                0x3, /* 3 */
/* 524 */ NdrFcShort( 0x8 ), /* 8 */
/* 526 */ NdrFcShort( 0x0 ), /* 0 */
/* 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 */
/* 536 */ NdrFcShort( 0xfffffe0 ), /* Offset= -32 (504) */
/* 538 */
                                0x1b, /* FC_CARRY */
                                0x3, /* 3 */
/* 540 */ NdrFcShort( 0x4 ), /* 4 */
/* 542 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
                                0x0, /* */
/* 544 */ NdrFcShort( 0x0 ), /* 0 */
/* 546 */
                                0x4b, /* FC_PP */
                                0x5c, /* FC_PAD */
                                0x48, /*
FC_VARIABLE_REPEAT */
                                0x49, /*
FC_FIXED_OFFSET */
/* 550 */ NdrFcShort( 0x4 ), /* 4 */
/* 552 */ NdrFcShort( 0x0 ), /* 0 */
/* 554 */ NdrFcShort( 0x1 ), /* 1 */
/* 556 */ NdrFcShort( 0x0 ), /* 0 */
/* 558 */ NdrFcShort( 0x0 ), /* 0 */
/* 560 */ 0x12, 0x0, /* FC_UP */
/* 562 */ NdrFcShort( 0x182 ), /* Offset= 386 (948) */
/* 564 */
                                0x5b, /* FC_END */
                                0x8, /* FC_LONG */
/* 566 */ 0x5c, /* FC_PAD */
                                0x5b, /* FC_END */
/* 568 */
                                0x1a, /*
FC_BOGUS_STRUCT */
                                0x3, /* 3 */
/* 570 */ NdrFcShort( 0x8 ), /* 8 */
/* 572 */ NdrFcShort( 0x0 ), /* 0 */
/* 574 */ NdrFcShort( 0x6 ), /* Offset= 6 (580) */
/* 576 */ 0x8, /* FC_LONG */
                                0x36, /* FC_POINTER */
/* 578 */ 0x5c, /* FC_PAD */
                                0x5b, /* FC_END */
/* 580 */
                                0x11, 0x0, /* FC_RP */
/* 582 */ NdrFcShort( 0xfffffd4 ), /* Offset= -44 (538) */
/* 584 */
                                0x2f, /* FC_IP */
                                0x5a, /*
FC_CONSTANT_IID */
/* 586 */ NdrFcLong( 0x2f ), /* 47 */
/* 590 */ NdrFcShort( 0x0 ), /* 0 */
/* 592 */ NdrFcShort( 0x0 ), /* 0 */
/* 594 */ 0xc0, /* 192 */
                                0x0, /* 0 */
/* 596 */ 0x0, /* 0 */
                                0x0, /* 0 */
/* 598 */ 0x0, /* 0 */
                                0x0, /* 0 */
/* 600 */ 0x0, /* 0 */
                                0x46, /* 70 */
/* 602 */
                                0x1b, /* FC_CARRY */
                                0x0, /* 0 */
/* 604 */ NdrFcShort( 0x1 ), /* 1 */
/* 606 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
                                0x0, /* */
/* 608 */ NdrFcShort( 0x4 ), /* 4 */
/* 610 */ 0x1, /* FC_BYTE */
                                0x5b, /* FC_END */
/* 612 */
                                0x1a, /*
FC_BOGUS_STRUCT */
                                0x3, /* 3 */
/* 614 */ NdrFcShort( 0x10 ), /* 16 */

```



/* 616 */ NdrFcShort( 0x0 ), /* 0 */	0x4c, /*
/* 618 */ NdrFcShort( 0xa ), /* Offset= 10 (628) */	FC_EMBEDDED_COMPLEX /*
/* 620 */ 0x8, /* FC_LONG */	/* 692 */ 0x0, /* 0 */
/* 622 */ 0x4c, /* FC_EMBEDDED_COMPLEX */	NdrFcShort( 0xffffffffl ), /* Offset=
/* 624 */ NdrFcShort( 0xffffffff8 ), /* Offset= -40 (584) */	-15 (678) */
/* 626 */ 0x36, /* FC_POINTER */	0x5b, /* FC_END */
/* 628 */ 0x5b, /* FC_END */	0x1a, /*
/* 630 */ NdrFcShort( 0xffffffffe4 ), /* Offset= -28 (602) */	FC_BOGUS_STRUCT /*
/* 632 */ 0x12, 0x0, /* FC_UP */	0x3, /* 3 */
/* 634 */ NdrFcShort( 0xffffffffe4 ), /* Offset= -28 (602) */	/* 698 */ NdrFcShort( 0x18 ), /* 24 */
/* 636 */ 0x1b, /* FC_CARRY */	/* 700 */ NdrFcShort( 0x0 ), /* 0 */
/* 638 */ NdrFcShort( 0x0 ), /* 0 */	/* 702 */ NdrFcShort( 0xa ), /* Offset= 10 (712) */
/* 640 */ 0x4b, /* FC_PP */	/* 704 */ 0x8, /* FC_LONG */
/* 642 */ 0x5c, /* FC_PAD */	0x36, /* FC_POINTER */
FC_VARIABLE_REPEAT /*	/* 706 */ 0x4c, /* FC_EMBEDDED_COMPLEX */
FC_FIXED_OFFSET /*	0x0, /* 0 */
/* 644 */ NdrFcShort( 0x4 ), /* 4 */	/* 708 */ NdrFcShort( 0xffffffffe8 ), /* Offset= -24 (684) */
/* 646 */ NdrFcShort( 0x0 ), /* 0 */	/* 710 */ 0x5c, /* FC_PAD */
/* 648 */ NdrFcShort( 0x1 ), /* 1 */	/* 712 */ 0x5b, /* FC_END */
/* 650 */ NdrFcShort( 0x0 ), /* 0 */	/* 714 */ NdrFcShort( 0xffffffff0c ), /* Offset= -244 (470) */
/* 652 */ NdrFcShort( 0x0 ), /* 0 */	/* 716 */ 0x11, 0x0, /* FC_RP */
/* 654 */ 0x12, 0x0, /* FC_UP */	/* 718 */ NdrFcShort( 0x1 ), /* 1 */
/* 656 */ NdrFcShort( 0xffffffffd4 ), /* Offset= -44 (612) */	/* 720 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
/* 658 */ 0x5b, /* FC_END */	0x0, /* 0 */
/* 660 */ 0x5c, /* FC_PAD */	/* 722 */ NdrFcShort( 0x0 ), /* 0 */
/* 662 */ 0x5b, /* FC_END */	/* 724 */ 0x1, /* FC_BYTE */
/* 664 */ 0x1a, /*	0x5b, /* FC_END */
FC_BOGUS_STRUCT /*	/* 726 */ 0x16, /* FC_PSTRUCT */
/* 666 */ NdrFcShort( 0x8 ), /* 8 */	0x3, /* 3 */
/* 668 */ NdrFcShort( 0x0 ), /* 0 */	/* 728 */ NdrFcShort( 0x8 ), /* 8 */
/* 670 */ NdrFcShort( 0x6 ), /* Offset= 6 (674) */	/* 730 */ 0x4b, /* FC_PP */
/* 672 */ 0x8, /* FC_LONG */	0x5c, /* FC_PAD */
/* 674 */ 0x5c, /* FC_PAD */	/* 732 */ 0x46, /* FC_NO_REPEAT */
/* 676 */ 0x5b, /* FC_END */	0x5c, /* FC_PAD */
/* 678 */ 0x11, 0x0, /* FC_RP */	/* 734 */ NdrFcShort( 0x4 ), /* 4 */
/* 680 */ NdrFcShort( 0xffffffffd4 ), /* Offset= -44 (632) */	/* 736 */ NdrFcShort( 0x4 ), /* 4 */
/* 682 */ 0x1d, /* FC_SMFARRAY */	/* 738 */ 0x12, 0x0, /* FC_UP */
/* 684 */ 0x0, /* 0 */	/* 740 */ NdrFcShort( 0xffffffffe8 ), /* Offset= -24 (716) */
/* 686 */ NdrFcShort( 0x8 ), /* 8 */	/* 742 */ 0x5b, /* FC_END */
/* 688 */ 0x1, /* FC_BYTE */	0x8, /* FC_LONG */
/* 690 */ 0x6, /* FC_SHORT */	/* 744 */ 0x8, /* FC_LONG */
	0x5b, /* FC_END */
	/* 746 */ 0x11, 0x0, /* FC_RP */
	0x1b, /* FC_CARRY */
	0x1, /* 1 */
	/* 748 */ NdrFcShort( 0x2 ), /* 2 */
	/* 750 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
	0x0, /* 0 */
	/* 752 */ NdrFcShort( 0x0 ), /* 0 */
	/* 754 */ 0x6, /* FC_SHORT */
	0x5b, /* FC_END */
	/* 756 */ 0x16, /* FC_PSTRUCT */
	0x3, /* 3 */
	/* 758 */ NdrFcShort( 0x8 ), /* 8 */
	/* 760 */ 0x4b, /* FC_PP */
	0x5c, /* FC_PAD */

/* 762 */			0x46,	/* FC_NO_REPEAT */	/* 834 */	0x8,	/* FC_LONG */
			0x5c,	/* FC_PAD */		0x5b,	/* FC_END */
/* 764 */	NdrFcShort( 0x4 ),	/* 4 */			/* 836 */		
/* 766 */	NdrFcShort( 0x4 ),	/* 4 */				0x15,	/* FC_STRUCT */
/* 768 */	0x12, 0x0, /* FC_UP */					0x3,	/* 3 */
/* 770 */	NdrFcShort( 0xfffffe8 ),	/* Offset= -24 (746) */			/* 838 */	NdrFcShort( 0x8 ),	/* 8 */
/* 772 */			0x5b,	/* FC_END */	/* 840 */	0x8,	/* FC_LONG */
			0x8,	/* FC_LONG */	/* 842 */	0x5c,	/* FC_PAD */
/* 774 */	0x8,	/* FC_LONG */	0x5b,	/* FC_END */	/* 844 */	0x5b,	/* FC_END */
/* 776 */			0x1b,	/* FC_CARRAY */		0x1b,	/* FC_CARRAY */
			0x3,	/* 3 */	/* 846 */	NdrFcShort( 0x8 ),	/* 8 */
/* 778 */	NdrFcShort( 0x4 ),	/* 4 */			/* 848 */	0x7,	/* Corr desc: FC_USHORT */
/* 780 */	0x19,	/* Corr desc: field pointer, FC_ULONG */			/* 850 */	NdrFcShort( 0xffd8 ),	/* -40 */
		/* */	0x0,	/* */	/* 852 */	0x4c,	/* FC_EMBEDDED_COMPLEX */
/* 782 */	NdrFcShort( 0x0 ),	/* 0 */				0x0,	/* 0 */
/* 784 */	0x8,	/* FC_LONG */	0x5b,	/* FC_END */	/* 854 */	NdrFcShort( 0xfffffee ),	/* Offset= -18 (836) */
/* 786 */			0x16,	/* FC_PSTRUCT */	/* 856 */	0x5c,	/* FC_PAD */
			0x3,	/* 3 */		0x5b,	/* FC_END */
/* 788 */	NdrFcShort( 0x8 ),	/* 8 */			/* 858 */		0x1a,
/* 790 */			0x4b,	/* FC_PP */			/* 3 */
			0x5c,	/* FC_PAD */	/* 860 */	NdrFcShort( 0x28 ),	/* 40 */
/* 792 */			0x46,	/* FC_NO_REPEAT */	/* 862 */	NdrFcShort( 0xfffffee ),	/* Offset= -18 (844) */
			0x5c,	/* FC_PAD */	/* 864 */	NdrFcShort( 0x0 ),	/* Offset= 0 (864) */
/* 794 */	NdrFcShort( 0x4 ),	/* 4 */			/* 866 */	0x6,	/* FC_SHORT */
/* 796 */	NdrFcShort( 0x4 ),	/* 4 */				0x6,	/* FC_SHORT */
/* 798 */	0x12, 0x0, /* FC_UP */				/* 868 */	0x38,	/* FC_ALIGNM4 */
/* 800 */	NdrFcShort( 0xfffffe8 ),	/* Offset= -24 (776) */			/* 870 */	0x8,	/* FC_LONG */
/* 802 */			0x5b,	/* FC_END */		0x4c,	/* FC_SHORT */
			0x8,	/* FC_LONG */	FC_EMBEDDED_COMPLEX */		
/* 804 */	0x8,	/* FC_LONG */	0x5b,	/* FC_END */	/* 872 */	0x0,	/* 0 */
/* 806 */			0x1b,	/* FC_CARRAY */			NdrFcShort( 0xfffffd7 ),
			0x7,	/* 7 */			/* Offset=
/* 808 */	NdrFcShort( 0x8 ),	/* 8 */			/* 876 */		0x5b,
/* 810 */	0x19,	/* Corr desc: field pointer, FC_ULONG */					/* FC_END */
		/* */			/* 878 */	NdrFcShort( 0xfffffe6 ),	/* Offset= -266 (612) */
/* 812 */	NdrFcShort( 0x0 ),	/* 0 */			/* 880 */		
/* 814 */	0xb,	/* FC_HYPER */				0x12, 0x0, /* FC_UP [simple_pointer] */	/* FC_UP [simple_pointer] */
/* 816 */			0x5b,	/* FC_END */		0x5c,	/* FC_PAD */
			0x16,	/* FC_PSTRUCT */	/* 884 */		
			0x3,	/* 3 */	/* 886 */	0x6,	/* FC_SHORT */
/* 818 */	NdrFcShort( 0x8 ),	/* 8 */				0x5c,	/* FC_PAD */
/* 820 */			0x4b,	/* FC_PP */	/* 888 */		
			0x5c,	/* FC_PAD */	/* 890 */	0x8,	0x12, 0x8, /* FC_UP [simple_pointer] */
/* 822 */			0x46,	/* FC_NO_REPEAT */			/* FC_LONG */
			0x5c,	/* FC_PAD */	/* 892 */		/* FC_PAD */
/* 824 */	NdrFcShort( 0x4 ),	/* 4 */			/* 894 */	0xa,	0x12, 0x8, /* FC_UP [simple_pointer] */
/* 826 */	NdrFcShort( 0x4 ),	/* 4 */					/* FC_FLOAT */
/* 828 */	0x12, 0x0, /* FC_UP */				/* 896 */		0x5c,
/* 830 */	NdrFcShort( 0xfffffe8 ),	/* Offset= -24 (806) */			/* 898 */	0xc,	0x12, 0x8, /* FC_UP [simple_pointer] */
/* 832 */			0x5b,	/* FC_END */	/* 900 */		/* FC_DOUBLE */
						0x5c,	/* FC_PAD */
					/* 902 */	NdrFcShort( 0xffffd90 ),	/* Offset= -624 (278) */
					/* 904 */		

```

                                0x12, 0x10,          /* FC_UP
[pointer_deref] */
/* 906 */ NdrFcShort( 0xfffffd92 ), /* Offset= -622 (284) */
/* 908 */
                                0x12, 0x10,          /* FC_UP
[pointer_deref] */
/* 910 */ NdrFcShort( 0xffffda6 ), /* Offset= -602 (308) */
/* 912 */
                                0x12, 0x10,          /* FC_UP
[pointer_deref] */
/* 914 */ NdrFcShort( 0xffffdb4 ), /* Offset= -588 (326) */
/* 916 */
                                0x12, 0x10,          /* FC_UP
[pointer_deref] */
/* 918 */ NdrFcShort( 0xffffdc2 ), /* Offset= -574 (344) */
/* 920 */
                                0x12, 0x10,          /* FC_UP
[pointer_deref] */
/* 922 */ NdrFcShort( 0x2 ), /* Offset= 2 (924) */
/* 924 */
                                0x12, 0x0, /* FC_UP */
/* 926 */ NdrFcShort( 0x16 ), /* Offset= 22 (948) */
/* 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( 0xfffffc42 ), /* 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( 0xfffffc32 ), /* Offset= -974 (2) */
/* 978 */
                                0x11, 0x4, /* FC_RP [allocated_on_stack] */
/* 980 */ NdrFcShort( 0x6 ), /* Offset= 6 (986) */
/* 982 */
                                0x13, 0x0, /* FC_OP */
/* 984 */ NdrFcShort( 0xfffffde ), /* Offset= -36 (948) */
/* 986 */ 0xb4, /* FC_USER_MARSHAL */
                                0x83,        /* 131 */
/* 988 */ NdrFcShort( 0x0 ), /* 0 */
/* 990 */ NdrFcShort( 0x10 ), /* 16 */
/* 992 */ NdrFcShort( 0x0 ), /* 0 */
/* 994 */ NdrFcShort( 0xffffff4 ), /* 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 */
};
#endif /* !defined(_M_IA64) && !defined(_M_AXP64) */
#pragma warning( disable: 4049 ) /* more than 64k source lines */

```

```

/* this ALWAYS GENERATED file contains the proxy stub code */

/* File created by MIDL compiler version 5.03.0280 */
/* at Thu Dec 13 23:13:08 2001
*/
/* Compiler settings for \src\tpcc_com_ps.idl:
Oicf (OptLev=i2), W1, Zp8, env=Win64 (32b run,appending), ms_ext, c_ext,
robust
error checks: allocation ref bounds_check enum stub_data
VC __declspec() decoration level:
__declspec(uuid()), __declspec(selectany), __declspec(novtable)
DECLSPEC_UUID(), MIDL_INTERFACE()
*/
//@@@MIDL_FILE_HEADING( )

#ifdef _M_IA64 || defined(_M_AXP64)
#define USE_STUBLESS_PROXY

/* verify that the <rpcproxy.h> version is high enough to compile this file*/
#ifndef __REDQ_RPCPROXY_H_VERSION__
#define __REQUIRED_RPCPROXY_H_VERSION__ 475
#endif

#include "rpcproxy.h"
#ifndef __RPCPROXY_H_VERSION__
#error this stub requires an updated version of <rpcproxy.h>
#endif // __RPCPROXY_H_VERSION__

#include "tpcc_com_ps.h"

#define TYPE_FORMAT_STRING_SIZE 979
#define PROC_FORMAT_STRING_SIZE 253
#define TRANSMIT_AS_TABLE_SIZE 0
#define WIRE_MARSHAL_TABLE_SIZE 1

typedef struct _MIDL_TYPE_FORMAT_STRING
{
    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,{0x00,0x00,0x00,0x00,0x00,0x00,0x00,0
x00}} */

/* Object interface: IUnknown, ver. 0.0,
GUID={0x00000000,0x0000,0x0000,{0xC0,0x00,0x00,0x00,0x00,0x00,0x00,0
x46}} */

```

```

/* Object interface: ITPCC, ver. 0.0,
GUID={0xFEE6AA2,0x84B1,0x11d2,{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")
static const unsigned short ITPCC_FormatStringOffsetTable[] =
{
    0,
    44,
    88,
    132,
    176,
    220
};

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
};

```

```

extern const USER_MARSHAL_ROUTINE_QUADRUPLE
UserMarshalRoutines[ WIRE_MARSHAL_TABLE_SIZE ];

static const MIDL_STUB_DESC Object_StubDesc =
{
    0,
    NdrOleAllocate,
    NdrOleFree,
    0,
    0,
    0,
    0,
    0,
    0,
    0,
    0,
    __MIDL_TypeFormatString.Format,
    1, /* -error bounds_check flag */
    0x50002, /* Ndr library version */
    0,
    0x5030118, /* MIDL Version 5.3.280 */
    0,
    UserMarshalRoutines,
    0, /* notify & notify_flag routine table */
    0x1, /* MIDL flag */
    0, /* Reserved3 */
    0, /* Reserved4 */
    0 /* Reserved5 */
};

#pragma data_seg(".rdata")

static const USER_MARSHAL_ROUTINE_QUADRUPLE
UserMarshalRoutines[ WIRE_MARSHAL_TABLE_SIZE ] =
{
    {
        VARIANT_UserSize
        ,VARIANT_UserMarshal
        ,VARIANT_UserUnmarshal
        ,VARIANT_UserFree
    }
};

#if !defined(__RPC_WIN64__)
#error Invalid build platform for this stub.
#endif

static const MIDL_PROC_FORMAT_STRING __MIDL_ProcFormatString =
{
    {
        0,
        {
            /* Procedure NewOrder */
            FC_AUTO_HANDLE /*
                0x33, /*
                    /* Old Flags: object,
                    Oi2 */
                    /* 2 */ NdrFcLong( 0x0 ), /* 0 */
                    /* 6 */ NdrFcShort( 0x3 ), /* 3 */
                    #ifndef _ALPHA_
                    /* 8 */ NdrFcShort( 0x38 ), /* ia64 Stack size/offset = 56 */
                    #else
                    NdrFcShort( 0x30 ), /* axp64 Stack
                    size/offset = 48 */
                    #endif
                    /* 10 */ NdrFcShort( 0x0 ), /* 0 */
                */
            }
        }
    }
};

```

```

/* 12 */ NdrFcShort( 0x8 ), /* 8 */
/* 14 */ 0x47, /* Oi2 Flags: srv must size, clt must size, has
return, has ext, */
0x3, /* 3 */
/* 16 */ 0xa, /* 10 */
0x7, /* Ext Flags: new corr
desc, clt corr check, srv corr check, */
/* 18 */ NdrFcShort( 0x20 ), /* 32 */
/* 20 */ NdrFcShort( 0x20 ), /* 32 */
/* 22 */ NdrFcShort( 0x0 ), /* 0 */
/* 24 */ NdrFcShort( 0x0 ), /* 0 */
/* Parameter txn_in */
/* 26 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by val, */
#ifdef _ALPHA_
/* 28 */ NdrFcShort( 0x10 ), /* ia64 Stack size/offset = 16 */
#else
NdrFcShort( 0x8 ), /* axp64 Stack
size/offset = 8 */
#endif
/* 30 */ NdrFcShort( 0x3b6 ), /* Type Offset=950 */
/* Parameter txn_out */
/* 32 */ NdrFcShort( 0x6113 ), /* Flags: must size, must free, out,
simple ref, srv alloc size=24 */
#ifdef _ALPHA_
/* 34 */ NdrFcShort( 0x28 ), /* ia64 Stack size/offset = 40 */
#else
NdrFcShort( 0x20 ), /* axp64 Stack
size/offset = 32 */
#endif
/* 36 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */
/* Return value */
/* 38 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifdef _ALPHA_
/* 40 */ NdrFcShort( 0x30 ), /* ia64 Stack size/offset = 48 */
#else
NdrFcShort( 0x28 ), /* axp64 Stack
size/offset = 40 */
#endif
/* 42 */ 0x8, /* FC_LONG */
0x0, /* 0 */
/* Procedure Payment */
/* 44 */ 0x33, /* FC_AUTO_HANDLE */
0x6c, /* Old Flags: object,
Oi2 */
/* 46 */ NdrFcLong( 0x0 ), /* 0 */
/* 50 */ NdrFcShort( 0x4 ), /* 4 */
#ifdef _ALPHA_
/* 52 */ NdrFcShort( 0x38 ), /* ia64 Stack size/offset = 56 */
#else
NdrFcShort( 0x30 ), /* axp64 Stack
size/offset = 48 */
#endif
/* 54 */ NdrFcShort( 0x0 ), /* 0 */
/* 56 */ NdrFcShort( 0x8 ), /* 8 */
/* 58 */ 0x47, /* Oi2 Flags: srv must size, clt must size, has
return, has ext, */
0x3, /* 3 */
/* 60 */ 0xa, /* 10 */
0x7, /* Ext Flags: new corr
desc, clt corr check, srv corr check, */

```

```

/* 62 */ NdrFcShort( 0x20 ), /* 32 */
/* 64 */ NdrFcShort( 0x20 ), /* 32 */
/* 66 */ NdrFcShort( 0x0 ), /* 0 */
/* 68 */ NdrFcShort( 0x0 ), /* 0 */

/* Parameter txn_in */

/* 70 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by val, */
#ifdef _ALPHA_
/* 72 */ NdrFcShort( 0x10 ), /* ia64 Stack size/offset = 16 */
#else
NdrFcShort( 0x8 ), /* axp64 Stack
size/offset = 8 */
#endif
/* 74 */ NdrFcShort( 0x3b6 ), /* Type Offset=950 */

/* Parameter txn_out */

/* 76 */ NdrFcShort( 0x6113 ), /* Flags: must size, must free, out,
simple ref, srv alloc size=24 */
#ifdef _ALPHA_
/* 78 */ NdrFcShort( 0x28 ), /* ia64 Stack size/offset = 40 */
#else
NdrFcShort( 0x20 ), /* axp64 Stack
size/offset = 32 */
#endif
/* 80 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */

/* Return value */

/* 82 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifdef _ALPHA_
/* 84 */ NdrFcShort( 0x30 ), /* ia64 Stack size/offset = 48 */
#else
NdrFcShort( 0x28 ), /* axp64 Stack
size/offset = 40 */
#endif
/* 86 */ 0x8, /* FC_LONG */
0x0, /* 0 */

/* Procedure Delivery */

/* 88 */ 0x33, /* FC_AUTO_HANDLE */
0x6c, /* Old Flags: object,
Oi2 */
/* 90 */ NdrFcLong( 0x0 ), /* 0 */
/* 94 */ NdrFcShort( 0x5 ), /* 5 */
#ifdef _ALPHA_
/* 96 */ NdrFcShort( 0x38 ), /* ia64 Stack size/offset = 56 */
#else
NdrFcShort( 0x30 ), /* axp64 Stack
size/offset = 48 */
#endif
/* 98 */ NdrFcShort( 0x0 ), /* 0 */
/* 100 */ NdrFcShort( 0x8 ), /* 8 */
/* 102 */ 0x47, /* Oi2 Flags: srv must size, clt must size, has
return, has ext, */
0x3, /* 3 */
/* 104 */ 0xa, /* 10 */
0x7, /* Ext Flags: new corr
desc, clt corr check, srv corr check, */
/* 106 */ NdrFcShort( 0x20 ), /* 32 */
/* 108 */ NdrFcShort( 0x20 ), /* 32 */
/* 110 */ NdrFcShort( 0x0 ), /* 0 */
/* 112 */ NdrFcShort( 0x0 ), /* 0 */

/* Parameter txn_in */

/* 114 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by val, */
#ifdef _ALPHA_
/* 116 */ NdrFcShort( 0x10 ), /* ia64 Stack size/offset = 16 */
#else
NdrFcShort( 0x8 ), /* axp64 Stack
size/offset = 8 */
#endif
/* 118 */ NdrFcShort( 0x3b6 ), /* Type Offset=950 */

/* Parameter txn_out */

/* 120 */ NdrFcShort( 0x6113 ), /* Flags: must size, must free, out,
simple ref, srv alloc size=24 */
#ifdef _ALPHA_
/* 122 */ NdrFcShort( 0x28 ), /* ia64 Stack size/offset = 40 */
#else
NdrFcShort( 0x20 ), /* axp64 Stack
size/offset = 32 */
#endif
/* 124 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */

/* Return value */

/* 126 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifdef _ALPHA_
/* 128 */ NdrFcShort( 0x30 ), /* ia64 Stack size/offset = 48 */
#else
NdrFcShort( 0x28 ), /* axp64 Stack
size/offset = 40 */
#endif
/* 130 */ 0x8, /* FC_LONG */
0x0, /* 0 */

/* Procedure StockLevel */

/* 132 */ 0x33, /* FC_AUTO_HANDLE */
0x6c, /* Old Flags: object,
Oi2 */
/* 134 */ NdrFcLong( 0x0 ), /* 0 */
/* 138 */ NdrFcShort( 0x6 ), /* 6 */
#ifdef _ALPHA_
/* 140 */ NdrFcShort( 0x38 ), /* ia64 Stack size/offset = 56 */
#else
NdrFcShort( 0x30 ), /* axp64 Stack
size/offset = 48 */
#endif
/* 142 */ NdrFcShort( 0x0 ), /* 0 */
/* 144 */ NdrFcShort( 0x8 ), /* 8 */
/* 146 */ 0x47, /* Oi2 Flags: srv must size, clt must size, has
return, has ext, */
0x3, /* 3 */
/* 148 */ 0xa, /* 10 */
0x7, /* Ext Flags: new corr
desc, clt corr check, srv corr check, */
/* 150 */ NdrFcShort( 0x20 ), /* 32 */
/* 152 */ NdrFcShort( 0x20 ), /* 32 */
/* 154 */ NdrFcShort( 0x0 ), /* 0 */
/* 156 */ NdrFcShort( 0x0 ), /* 0 */

/* Parameter txn_in */

/* 158 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by val, */
#ifdef _ALPHA_
/* 160 */ NdrFcShort( 0x10 ), /* ia64 Stack size/offset = 16 */
#else
NdrFcShort( 0x8 ), /* axp64 Stack
size/offset = 8 */
#endif

```

```

/* 162 */ NdrFcShort( 0x3b6 ), /* Type Offset=950 */
        /* Parameter txn_out */

/* 164 */ NdrFcShort( 0x6113 ), /* Flags: must size, must free, out,
simple ref, srv alloc size=24 */
#ifdef _ALPHA_
/* 166 */ NdrFcShort( 0x28 ), /* ia64 Stack size/offset = 40 */
#else
        NdrFcShort( 0x20 ), /* axp64 Stack
size/offset = 32 */
#endif
/* 168 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */

        /* Return value */

/* 170 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifdef _ALPHA_
/* 172 */ NdrFcShort( 0x30 ), /* ia64 Stack size/offset = 48 */
#else
        NdrFcShort( 0x28 ), /* axp64 Stack
size/offset = 40 */
#endif
/* 174 */ 0x8, /* FC_LONG */
        0x0, /* 0 */

        /* Procedure OrderStatus */

/* 176 */ 0x33, /* FC_AUTO_HANDLE */
        0x6c, /* Old Flags: object,
Oi2 */
/* 178 */ NdrFcLong( 0x0 ), /* 0 */
/* 182 */ NdrFcShort( 0x7 ), /* 7 */
#ifdef _ALPHA_
/* 184 */ NdrFcShort( 0x38 ), /* ia64 Stack size/offset = 56 */
#else
        NdrFcShort( 0x30 ), /* axp64 Stack
size/offset = 48 */
#endif
/* 186 */ NdrFcShort( 0x0 ), /* 0 */
/* 188 */ NdrFcShort( 0x8 ), /* 8 */
/* 190 */ 0x47, /* Oi2 Flags: srv must size, clt must size, has
return, has ext, */
        0x3, /* 3 */
/* 192 */ 0xa, /* 10 */
        0x7, /* Ext Flags: new corr
desc, clt corr check, srv corr check, */
/* 194 */ NdrFcShort( 0x20 ), /* 32 */
/* 196 */ NdrFcShort( 0x20 ), /* 32 */
/* 198 */ NdrFcShort( 0x0 ), /* 0 */
/* 200 */ NdrFcShort( 0x0 ), /* 0 */

        /* Parameter txn_in */

/* 202 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by val, */
#ifdef _ALPHA_
/* 204 */ NdrFcShort( 0x10 ), /* ia64 Stack size/offset = 16 */
#else
        NdrFcShort( 0x8 ), /* axp64 Stack
size/offset = 8 */
#endif
/* 206 */ NdrFcShort( 0x3b6 ), /* Type Offset=950 */

        /* Parameter txn_out */

/* 208 */ NdrFcShort( 0x6113 ), /* Flags: must size, must free, out,
simple ref, srv alloc size=24 */
#ifdef _ALPHA_
/* 210 */ NdrFcShort( 0x28 ), /* ia64 Stack size/offset = 40 */
#else
        NdrFcShort( 0x20 ), /* axp64 Stack
size/offset = 32 */
#endif
/* 212 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */

        /* Return value */

/* 214 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifdef _ALPHA_
/* 216 */ NdrFcShort( 0x30 ), /* ia64 Stack size/offset = 48 */
#else
        NdrFcShort( 0x28 ), /* axp64 Stack
size/offset = 40 */
#endif
/* 218 */ 0x8, /* FC_LONG */
        0x0, /* 0 */

        /* Procedure CallSetComplete */

/* 220 */ 0x33, /* FC_AUTO_HANDLE */
        0x6c, /* Old Flags: object,
Oi2 */
/* 222 */ NdrFcLong( 0x0 ), /* 0 */
/* 226 */ NdrFcShort( 0x8 ), /* 8 */
/* 228 */ NdrFcShort( 0x10 ), /* ia64, axp64 Stack size/offset = 16 */
/* 230 */ NdrFcShort( 0x0 ), /* 0 */
/* 232 */ NdrFcShort( 0x8 ), /* 8 */
/* 234 */ 0x44, /* Oi2 Flags: has return, has ext, */
        0x1, /* 1 */
/* 236 */ 0xa, /* 10 */
        0x1, /* Ext Flags: new corr
desc, */
/* 238 */ NdrFcShort( 0x0 ), /* 0 */
/* 240 */ NdrFcShort( 0x0 ), /* 0 */
/* 242 */ NdrFcShort( 0x0 ), /* 0 */
/* 244 */ NdrFcShort( 0x0 ), /* 0 */

        /* Return value */

/* 246 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
/* 248 */ NdrFcShort( 0x8 ), /* ia64, axp64 Stack size/offset = 8 */
/* 250 */ 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( 0x39e ), /* Offset= 926 (930) */
/* 6 */
        0x2b, /*
FC_NON_ENCAPSULATED_UNION */
        0x9, /* FC_ULONG */
/* 8 */ 0x7, /* Corr desc: FC_USHORT */
        0x0, /* */
/* 10 */ NdrFcShort( 0xffff8 ), /* -8 */
/* 12 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 14 */ NdrFcShort( 0x2 ), /* Offset= 2 (16) */
/* 16 */ NdrFcShort( 0x10 ), /* 16 */

```

```

/* 18 */ NdrFcShort( 0x2b ), /* 43 */
/* 20 */ NdrFcLong( 0x3 ), /* 3 */
/* 24 */ NdrFcShort( 0x8008 ), /* Simple arm type: FC_LONG */
/* 26 */ NdrFcLong( 0x11 ), /* 17 */
/* 30 */ NdrFcShort( 0x8001 ), /* Simple arm type: FC_BYTE */
/* 32 */ NdrFcLong( 0x2 ), /* 2 */
/* 36 */ NdrFcShort( 0x8006 ), /* Simple arm type: FC_SHORT */
/* 38 */ NdrFcLong( 0x4 ), /* 4 */
/* 42 */ NdrFcShort( 0x800a ), /* Simple arm type: FC_FLOAT */
/* 44 */ NdrFcLong( 0x5 ), /* 5 */
/* 48 */ NdrFcShort( 0x800c ), /* Simple arm type: FC_DOUBLE */
/*
/* 50 */ NdrFcLong( 0xb ), /* 11 */
/* 54 */ NdrFcShort( 0x8006 ), /* Simple arm type: FC_SHORT */
/* 56 */ NdrFcLong( 0xa ), /* 10 */
/* 60 */ NdrFcShort( 0x8008 ), /* Simple arm type: FC_LONG */
/* 62 */ NdrFcLong( 0x6 ), /* 6 */
/* 66 */ NdrFcShort( 0xd6 ), /* Offset= 214 (280) */
/* 68 */ NdrFcLong( 0x7 ), /* 7 */
/* 72 */ NdrFcShort( 0x800c ), /* Simple arm type: FC_DOUBLE */
/*
/* 74 */ NdrFcLong( 0x8 ), /* 8 */
/* 78 */ NdrFcShort( 0xd0 ), /* Offset= 208 (286) */
/* 80 */ NdrFcLong( 0xd ), /* 13 */
/* 84 */ NdrFcShort( 0xe4 ), /* Offset= 228 (312) */
/* 86 */ NdrFcLong( 0x9 ), /* 9 */
/* 90 */ NdrFcShort( 0xf0 ), /* Offset= 240 (330) */
/* 92 */ NdrFcLong( 0x2000 ), /* 8192 */
/* 96 */ NdrFcShort( 0xfc ), /* Offset= 252 (348) */
/* 98 */ NdrFcLong( 0x24 ), /* 36 */
/* 102 */ NdrFcShort( 0x2f4 ), /* Offset= 756 (858) */
/* 104 */ NdrFcLong( 0x4024 ), /* 16420 */
/* 108 */ NdrFcShort( 0x2ee ), /* Offset= 750 (858) */
/* 110 */ NdrFcLong( 0x4011 ), /* 16401 */
/* 114 */ NdrFcShort( 0x2ec ), /* Offset= 748 (862) */
/* 116 */ NdrFcLong( 0x4002 ), /* 16386 */
/* 120 */ NdrFcShort( 0x2ea ), /* Offset= 746 (866) */
/* 122 */ NdrFcLong( 0x4003 ), /* 16387 */
/* 126 */ NdrFcShort( 0x2e8 ), /* Offset= 744 (870) */
/* 128 */ NdrFcLong( 0x4004 ), /* 16388 */
/* 132 */ NdrFcShort( 0x2e6 ), /* Offset= 742 (874) */
/* 134 */ NdrFcLong( 0x4005 ), /* 16389 */
/* 138 */ NdrFcShort( 0x2e4 ), /* Offset= 740 (878) */
/* 140 */ NdrFcLong( 0x400b ), /* 16395 */
/* 144 */ NdrFcShort( 0x2d2 ), /* Offset= 722 (866) */
/* 146 */ NdrFcLong( 0x400a ), /* 16394 */
/* 150 */ NdrFcShort( 0x2d0 ), /* Offset= 720 (870) */
/* 152 */ NdrFcLong( 0x4006 ), /* 16390 */
/* 156 */ NdrFcShort( 0x2d6 ), /* Offset= 726 (882) */
/* 158 */ NdrFcLong( 0x4007 ), /* 16391 */
/* 162 */ NdrFcShort( 0x2cc ), /* Offset= 716 (878) */
/* 164 */ NdrFcLong( 0x4008 ), /* 16392 */
/* 168 */ NdrFcShort( 0x2ce ), /* Offset= 718 (886) */
/* 170 */ NdrFcLong( 0x400d ), /* 16397 */
/* 174 */ NdrFcShort( 0x2cc ), /* Offset= 716 (890) */
/* 176 */ NdrFcLong( 0x4009 ), /* 16393 */
/* 180 */ NdrFcShort( 0x2ca ), /* Offset= 714 (894) */
/* 182 */ NdrFcLong( 0x6000 ), /* 24576 */
/* 186 */ NdrFcShort( 0x2c8 ), /* Offset= 712 (898) */
/* 188 */ NdrFcLong( 0x400c ), /* 16396 */
/* 192 */ NdrFcShort( 0x2c6 ), /* Offset= 710 (902) */
/* 194 */ NdrFcLong( 0x10 ), /* 16 */
/* 198 */ NdrFcShort( 0x8002 ), /* Simple arm type: FC_CHAR */
/* 200 */ NdrFcLong( 0x12 ), /* 18 */
/* 204 */ NdrFcShort( 0x8006 ), /* Simple arm type: FC_SHORT */
/* 206 */ NdrFcLong( 0x13 ), /* 19 */
/* 210 */ NdrFcShort( 0x8008 ), /* Simple arm type: FC_LONG */
/* 212 */ NdrFcLong( 0x16 ), /* 22 */
/* 216 */ NdrFcShort( 0x8008 ), /* Simple arm type: FC_LONG */
/* 218 */ NdrFcLong( 0x17 ), /* 23 */
/* 222 */ NdrFcShort( 0x8008 ), /* Simple arm type: FC_LONG */
/* 224 */ NdrFcLong( 0xe ), /* 14 */
/* 228 */ NdrFcShort( 0x2aa ), /* Offset= 682 (910) */
/* 230 */ NdrFcLong( 0x400e ), /* 16398 */
/* 234 */ NdrFcShort( 0x2b0 ), /* Offset= 688 (922) */
/* 236 */ NdrFcLong( 0x4010 ), /* 16400 */
/* 240 */ NdrFcShort( 0x2ae ), /* Offset= 686 (926) */
/* 242 */ NdrFcLong( 0x4012 ), /* 16402 */
/* 246 */ NdrFcShort( 0x26c ), /* Offset= 620 (866) */
/* 248 */ NdrFcLong( 0x4013 ), /* 16403 */
/* 252 */ NdrFcShort( 0x26a ), /* Offset= 618 (870) */
/* 254 */ NdrFcLong( 0x4016 ), /* 16406 */
/* 258 */ NdrFcShort( 0x264 ), /* Offset= 612 (870) */
/* 260 */ NdrFcLong( 0x4017 ), /* 16407 */
/* 264 */ NdrFcShort( 0x25e ), /* Offset= 606 (870) */
/* 266 */ NdrFcLong( 0x0 ), /* 0 */
/* 270 */ NdrFcShort( 0x0 ), /* Offset= 0 (270) */
/* 272 */ NdrFcLong( 0x1 ), /* 1 */
/* 276 */ NdrFcShort( 0x0 ), /* Offset= 0 (276) */
/* 278 */ NdrFcShort( 0xffffffff ), /* Offset= -1 (277) */
/* 280 */
0x15, /* FC_STRUCT */
0x7, /* 7 */
/* 282 */ NdrFcShort( 0x8 ), /* 8 */
/* 284 */ 0xb, /* FC_HYPER */
0x5b, /* FC_END */
/* 286 */
0x12, 0x0, /* FC_UP */
/* 288 */ NdrFcShort( 0xe ), /* Offset= 14 (302) */
/* 290 */
0x1b, /* FC_CARRAY */
0x1, /* 1 */
/* 292 */ NdrFcShort( 0x2 ), /* 2 */
/* 294 */ 0x9, /* Corr desc: FC_ULONG */
0x0, /* */
/* 296 */ NdrFcShort( 0xfffc ), /* -4 */
/* 298 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 300 */ 0x6, /* FC_SHORT */
0x5b, /* FC_END */
/* 302 */
0x17, /* FC_CSTRUCT */
0x3, /* 3 */
/* 304 */ NdrFcShort( 0x8 ), /* 8 */
/* 306 */ NdrFcShort( 0xffffffff0 ), /* Offset= -16 (290) */
/* 308 */ 0x8, /* FC_LONG */
0x8, /* FC_LONG */
/* 310 */ 0x5c, /* FC_PAD */
0x5b, /* FC_END */
/* 312 */
0x2f, /* FC_IP */
0x5a, /*
FC_CONSTANT_IID */
/* 314 */ NdrFcLong( 0x0 ), /* 0 */
/* 318 */ NdrFcShort( 0x0 ), /* 0 */
/* 320 */ NdrFcShort( 0x0 ), /* 0 */
/* 322 */ 0xc0, /* 192 */
0x0, /* 0 */
/* 324 */ 0x0, /* 0 */
0x0, /* 0 */
/* 326 */ 0x0, /* 0 */
0x0, /* 0 */
/* 328 */ 0x0, /* 0 */
0x46, /* 70 */
/* 330 */
0x2f, /* FC_IP */

```



```

0x5a, /*
FC_CONSTANT_IID */
/* 332 */ NdrFcLong( 0x20400 ), /* 132096 */
/* 336 */ NdrFcShort( 0x0 ), /* 0 */
/* 338 */ NdrFcShort( 0x0 ), /* 0 */
/* 340 */ 0xc0, /* 192 */
/* 342 */ 0x0, /* 0 */
/* 344 */ 0x0, /* 0 */
/* 346 */ 0x0, /* 0 */
/* 348 */ 0x46, /* 70 */
0x12, 0x10, /* FC_UP
[pointer_deref] */
/* 350 */ NdrFcShort( 0x2 ), /* Offset= 2 (352) */
/* 352 */ 0x12, 0x0, /* FC_UP */
/* 354 */ NdrFcShort( 0x1e6 ), /* Offset= 486 (840) */
/* 356 */ 0x2a, /*
FC_ENCAPSULATED_UNION */
/* 358 */ NdrFcShort( 0x20 ), /* 32 */
/* 360 */ NdrFcShort( 0xa ), /* 10 */
/* 362 */ NdrFcLong( 0x8 ), /* 8 */
/* 366 */ NdrFcShort( 0x50 ), /* Offset= 80 (446) */
/* 368 */ NdrFcLong( 0xd ), /* 13 */
/* 372 */ NdrFcShort( 0x70 ), /* Offset= 112 (484) */
/* 374 */ NdrFcLong( 0x9 ), /* 9 */
/* 378 */ NdrFcShort( 0x90 ), /* Offset= 144 (522) */
/* 380 */ NdrFcLong( 0xc ), /* 12 */
/* 384 */ NdrFcShort( 0xb0 ), /* Offset= 176 (560) */
/* 386 */ NdrFcLong( 0x24 ), /* 36 */
/* 390 */ NdrFcShort( 0x104 ), /* Offset= 260 (650) */
/* 392 */ NdrFcLong( 0x800d ), /* 32781 */
/* 396 */ NdrFcShort( 0x120 ), /* Offset= 288 (684) */
/* 398 */ NdrFcLong( 0x10 ), /* 16 */
/* 402 */ NdrFcShort( 0x13a ), /* Offset= 314 (716) */
/* 404 */ NdrFcLong( 0x2 ), /* 2 */
/* 408 */ NdrFcShort( 0x150 ), /* Offset= 336 (744) */
/* 410 */ NdrFcLong( 0x3 ), /* 3 */
/* 414 */ NdrFcShort( 0x166 ), /* Offset= 358 (772) */
/* 416 */ NdrFcLong( 0x14 ), /* 20 */
/* 420 */ NdrFcShort( 0x17c ), /* Offset= 380 (800) */
/* 422 */ NdrFcShort( 0xffffffff ), /* Offset= -1 (421) */
/* 424 */ 0x21, /*
FC_BOGUS_ARRAY */
/* 426 */ NdrFcShort( 0x0 ), /* 0 */
/* 428 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
/* 430 */ 0x0, /* */
/* 432 */ NdrFcShort( 0x0 ), /* 0 */
/* 434 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 436 */ NdrFcLong( 0xffffffff ), /* -1 */
/* 438 */ NdrFcShort( 0x0 ), /* Corr flags: */
/* 440 */ 0x12, 0x0, /* FC_UP */
/* 442 */ NdrFcShort( 0xffffffff74 ), /* Offset= -140 (302) */
/* 444 */ 0x5c, /* FC_PAD */
/* 446 */ 0x5b, /* FC_END */
0x1a, /*
FC_BOGUS_STRUCT */
/* 448 */ NdrFcShort( 0x10 ), /* 16 */
/* 450 */ NdrFcShort( 0x0 ), /* 0 */
/* 452 */ NdrFcShort( 0x6 ), /* Offset= 6 (458) */
/* 454 */ 0x8, /* FC_LONG */
/* 456 */ 0x36, /* FC_POINTER */
/* 458 */ 0x5b, /* FC_END */
0x11, 0x0, /* FC_RP */
/* 460 */ NdrFcShort( 0xffffffffdc ), /* Offset= -36 (424) */
/* 462 */ 0x21, /*
FC_BOGUS_ARRAY */
/* 464 */ NdrFcShort( 0x0 ), /* 0 */
/* 466 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
/* 468 */ 0x0, /* */
/* 470 */ NdrFcShort( 0x0 ), /* 0 */
/* 472 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 474 */ NdrFcLong( 0xffffffff ), /* -1 */
/* 476 */ NdrFcShort( 0x0 ), /* Corr flags: */
/* 478 */ 0x4c, /* FC_EMBEDDED_COMPLEX */
/* 480 */ 0x0, /* 0 */
/* 482 */ NdrFcShort( 0xffffffff58 ), /* Offset= -168 (312) */
/* 484 */ 0x5c, /* FC_PAD */
/* 486 */ 0x5b, /* FC_END */
0x1a, /*
FC_BOGUS_STRUCT */
/* 488 */ NdrFcShort( 0x10 ), /* 16 */
/* 490 */ NdrFcShort( 0x0 ), /* 0 */
/* 492 */ NdrFcShort( 0x6 ), /* Offset= 6 (496) */
/* 494 */ 0x8, /* FC_LONG */
/* 496 */ 0x36, /* FC_POINTER */
/* 498 */ 0x5b, /* FC_END */
0x11, 0x0, /* FC_RP */
/* 500 */ NdrFcShort( 0xffffffffdc ), /* Offset= -36 (462) */
/* 502 */ 0x21, /*
FC_BOGUS_ARRAY */
/* 504 */ NdrFcShort( 0x0 ), /* 0 */
/* 506 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
/* 508 */ 0x0, /* */
/* 510 */ NdrFcShort( 0x0 ), /* 0 */
/* 512 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 514 */ NdrFcLong( 0xffffffff ), /* -1 */
/* 516 */ NdrFcShort( 0x0 ), /* Corr flags: */
/* 518 */ 0x4c, /* FC_EMBEDDED_COMPLEX */
/* 520 */ 0x0, /* 0 */
/* 522 */ NdrFcShort( 0xffffffff44 ), /* Offset= -188 (330) */
/* 524 */ 0x5c, /* FC_PAD */
/* 526 */ 0x5b, /* FC_END */
0x1a, /*
FC_BOGUS_STRUCT */
/* 528 */ NdrFcShort( 0x10 ), /* 16 */
/* 530 */ NdrFcShort( 0x0 ), /* 0 */
/* 532 */ NdrFcShort( 0x6 ), /* Offset= 6 (534) */
/* 534 */ 0x8, /* FC_LONG */
/* 536 */ 0x36, /* FC_POINTER */
/* 538 */ 0x5b, /* FC_END */
0x11, 0x0, /* FC_RP */

```

```

/* 536 */ NdrFcShort( 0xfffffde ), /* Offset= -36 (500) */
/* 538 */
FC_BOGUS_ARRAY */
0x21, /*
0x3, /* 3 */
/* 540 */ NdrFcShort( 0x0 ), /* 0 */
/* 542 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
0x0, /* */
/* 544 */ NdrFcShort( 0x0 ), /* 0 */
/* 546 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 548 */ NdrFcLong( 0xffffffff ), /* -1 */
/* 552 */ NdrFcShort( 0x0 ), /* Corr flags: */
/* 554 */
0x12, 0x0, /* FC_UP */
/* 556 */ NdrFcShort( 0x176 ), /* Offset= 374 (930) */
/* 558 */ 0x5c, /* FC_PAD */
0x5b, /* FC_END */
/* 560 */
0x1a, /*
FC_BOGUS_STRUCT */
0x3, /* 3 */
/* 562 */ NdrFcShort( 0x10 ), /* 16 */
/* 564 */ NdrFcShort( 0x0 ), /* 0 */
/* 566 */ NdrFcShort( 0x6 ), /* Offset= 6 (572) */
/* 568 */ 0x8, /* FC_LONG */
0x39, /* FC_ALIGNM8 */
/* 570 */ 0x36, /* FC_POINTER */
0x5b, /* FC_END */
/* 572 */
0x11, 0x0, /* FC_RP */
/* 574 */ NdrFcShort( 0xfffffde ), /* Offset= -36 (538) */
/* 576 */
0x2f, /* FC_IP */
0x5a, /*
FC_CONSTANT_IID */
/* 578 */ NdrFcLong( 0x2f ), /* 47 */
/* 582 */ NdrFcShort( 0x0 ), /* 0 */
/* 584 */ NdrFcShort( 0x0 ), /* 0 */
/* 586 */ 0xc0, /* 192 */
0x0, /* 0 */
/* 588 */ 0x0, /* 0 */
0x0, /* 0 */
/* 590 */ 0x0, /* 0 */
0x0, /* 0 */
/* 592 */ 0x0, /* 0 */
0x46, /* 70 */
/* 594 */
0x1b, /* FC_CARRAY */
0x0, /* 0 */
/* 596 */ NdrFcShort( 0x1 ), /* 1 */
/* 598 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
0x0, /* */
/* 600 */ NdrFcShort( 0x4 ), /* 4 */
/* 602 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 604 */ 0x1, /* FC_BYTE */
0x5b, /* FC_END */
/* 606 */
0x1a, /*
FC_BOGUS_STRUCT */
0x3, /* 3 */
/* 608 */ NdrFcShort( 0x18 ), /* 24 */
/* 610 */ NdrFcShort( 0x0 ), /* 0 */
/* 612 */ NdrFcShort( 0xc ), /* Offset= 12 (624) */
/* 614 */ 0x8, /* FC_LONG */
0x8, /* FC_LONG */
/* 616 */ 0x4c, /* FC_EMBEDDED_COMPLEX */
0x0, /* 0 */
/* 618 */ NdrFcShort( 0xfffffd6 ), /* Offset= -42 (576) */
/* 620 */ 0x39, /* FC_ALIGNM8 */
0x36, /* FC_POINTER */
/* 622 */ 0x5c, /* FC_PAD */
0x5b, /* FC_END */
/* 624 */
0x12, 0x0, /* FC_UP */
/* 626 */ NdrFcShort( 0xfffffe0 ), /* Offset= -32 (594) */
/* 628 */
0x21, /*
FC_BOGUS_ARRAY */
0x3, /* 3 */
/* 630 */ NdrFcShort( 0x0 ), /* 0 */
/* 632 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
0x0, /* */
/* 634 */ NdrFcShort( 0x0 ), /* 0 */
/* 636 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 638 */ NdrFcLong( 0xffffffff ), /* -1 */
/* 642 */ NdrFcShort( 0x0 ), /* Corr flags: */
/* 644 */
0x12, 0x0, /* FC_UP */
/* 646 */ NdrFcShort( 0xfffffd8 ), /* Offset= -40 (606) */
/* 648 */ 0x5c, /* FC_PAD */
0x5b, /* FC_END */
/* 650 */
0x1a, /*
FC_BOGUS_STRUCT */
0x3, /* 3 */
/* 652 */ NdrFcShort( 0x10 ), /* 16 */
/* 654 */ NdrFcShort( 0x0 ), /* 0 */
/* 656 */ NdrFcShort( 0x6 ), /* Offset= 6 (662) */
/* 658 */ 0x8, /* FC_LONG */
0x39, /* FC_ALIGNM8 */
/* 660 */ 0x36, /* FC_POINTER */
0x5b, /* FC_END */
/* 662 */
0x11, 0x0, /* FC_RP */
/* 664 */ NdrFcShort( 0xfffffde ), /* Offset= -36 (628) */
/* 666 */
0x1d, /* FC_SMFARRAY */
0x0, /* 0 */
/* 668 */ NdrFcShort( 0x8 ), /* 8 */
/* 670 */ 0x1, /* FC_BYTE */
0x5b, /* FC_END */
/* 672 */
0x15, /* FC_STRUCT */
0x3, /* 3 */
/* 674 */ NdrFcShort( 0x10 ), /* 16 */
/* 676 */ 0x8, /* FC_LONG */
0x6, /* FC_SHORT */
/* 678 */ 0x6, /* FC_SHORT */
0x4c, /*
FC_EMBEDDED_COMPLEX */
/* 680 */ 0x0, /* 0 */
NdrFcShort( 0xfffff1 ), /* Offset=
-15 (666) */
0x5b, /* FC_END */
/* 684 */
0x1a, /*
FC_BOGUS_STRUCT */
0x3, /* 3 */
/* 686 */ NdrFcShort( 0x20 ), /* 32 */
/* 688 */ NdrFcShort( 0x0 ), /* 0 */
/* 690 */ NdrFcShort( 0xa ), /* Offset= 10 (700) */
/* 692 */ 0x8, /* FC_LONG */
0x39, /* FC_ALIGNM8 */
/* 694 */ 0x36, /* FC_POINTER */
0x4c, /*
FC_EMBEDDED_COMPLEX */

```

```

/* 696 */ 0x0, /* 0 */
-25 (672) */ NdrFcShort( 0xffffffe7 ), /* Offset=
/* 700 */ 0x5b, /* FC_END */
0x11, 0x0, /* FC_RP */
/* 702 */ NdrFcShort( 0xfffff10 ), /* Offset=-240 (462) */
/* 704 */ 0x1b, /* FC_CARRAY */
0x0, /* 0 */
/* 706 */ NdrFcShort( 0x1 ), /* 1 */
/* 708 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
0x0, /* */
/* 710 */ NdrFcShort( 0x0 ), /* 0 */
/* 712 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 714 */ 0x1, /* FC_BYTE */
0x5b, /* FC_END */
/* 716 */ 0x1a, /*
FC_BOGUS_STRUCT */
0x3, /* 3 */
/* 718 */ NdrFcShort( 0x10 ), /* 16 */
/* 720 */ NdrFcShort( 0x0 ), /* 0 */
/* 722 */ NdrFcShort( 0x6 ), /* Offset= 6 (728) */
/* 724 */ 0x8, /* FC_LONG */
0x39, /* FC_ALIGNM8 */
/* 726 */ 0x36, /* FC_POINTER */
0x5b, /* FC_END */
/* 728 */ 0x12, 0x0, /* FC_UP */
/* 730 */ NdrFcShort( 0xfffffe6 ), /* Offset=-26 (704) */
/* 732 */ 0x1b, /* FC_CARRAY */
0x1, /* 1 */
/* 734 */ NdrFcShort( 0x2 ), /* 2 */
/* 736 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
0x0, /* */
/* 738 */ NdrFcShort( 0x0 ), /* 0 */
/* 740 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 742 */ 0x6, /* FC_SHORT */
0x5b, /* FC_END */
/* 744 */ 0x1a, /*
FC_BOGUS_STRUCT */
0x3, /* 3 */
/* 746 */ NdrFcShort( 0x10 ), /* 16 */
/* 748 */ NdrFcShort( 0x0 ), /* 0 */
/* 750 */ NdrFcShort( 0x6 ), /* Offset= 6 (756) */
/* 752 */ 0x8, /* FC_LONG */
0x39, /* FC_ALIGNM8 */
/* 754 */ 0x36, /* FC_POINTER */
0x5b, /* FC_END */
/* 756 */ 0x12, 0x0, /* FC_UP */
/* 758 */ NdrFcShort( 0xfffffe6 ), /* Offset=-26 (732) */
/* 760 */ 0x1b, /* FC_CARRAY */
0x3, /* 3 */
/* 762 */ NdrFcShort( 0x4 ), /* 4 */
/* 764 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
0x0, /* */
/* 766 */ NdrFcShort( 0x0 ), /* 0 */
/* 768 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 770 */ 0x8, /* FC_LONG */
0x5b, /* FC_END */
/* 772 */ 0x1a, /*
FC_BOGUS_STRUCT */
0x3, /* 3 */
/* 774 */ NdrFcShort( 0x10 ), /* 16 */
/* 776 */ NdrFcShort( 0x0 ), /* 0 */
/* 778 */ NdrFcShort( 0x6 ), /* Offset= 6 (784) */
/* 780 */ 0x8, /* FC_LONG */
0x39, /* FC_ALIGNM8 */
/* 782 */ 0x36, /* FC_POINTER */
0x5b, /* FC_END */
/* 784 */ 0x12, 0x0, /* FC_UP */
/* 786 */ NdrFcShort( 0xfffffe6 ), /* Offset=-26 (760) */
/* 788 */ 0x1b, /* FC_CARRAY */
0x7, /* 7 */
/* 790 */ NdrFcShort( 0x8 ), /* 8 */
/* 792 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
0x0, /* */
/* 794 */ NdrFcShort( 0x0 ), /* 0 */
/* 796 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 798 */ 0xb, /* FC_HYPER */
0x5b, /* FC_END */
/* 800 */ 0x1a, /*
FC_BOGUS_STRUCT */
0x3, /* 3 */
/* 802 */ NdrFcShort( 0x10 ), /* 16 */
/* 804 */ NdrFcShort( 0x0 ), /* 0 */
/* 806 */ NdrFcShort( 0x6 ), /* Offset= 6 (812) */
/* 808 */ 0x8, /* FC_LONG */
0x39, /* FC_ALIGNM8 */
/* 810 */ 0x36, /* FC_POINTER */
0x5b, /* FC_END */
/* 812 */ 0x12, 0x0, /* FC_UP */
/* 814 */ NdrFcShort( 0xfffffe6 ), /* Offset=-26 (788) */
/* 816 */ 0x15, /* FC_STRUCT */
0x3, /* 3 */
/* 818 */ NdrFcShort( 0x8 ), /* 8 */
/* 820 */ 0x8, /* FC_LONG */
0x8, /* FC_LONG */
/* 822 */ 0x5c, /* FC_PAD */
0x5b, /* FC_END */
/* 824 */ 0x1b, /* FC_CARRAY */
0x3, /* 3 */
/* 826 */ NdrFcShort( 0x8 ), /* 8 */
/* 828 */ 0x7, /* Corr desc: FC_USHORT */
0x0, /* */
/* 830 */ NdrFcShort( 0xffc8 ), /* -56 */
/* 832 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 834 */ 0x4c, /* FC_EMBEDDED_COMPLEX */
0x0, /* 0 */
/* 836 */ NdrFcShort( 0xfffffec ), /* Offset=-20 (816) */
/* 838 */ 0x5c, /* FC_PAD */
0x5b, /* FC_END */
/* 840 */ 0x1a, /*
FC_BOGUS_STRUCT */
0x3, /* 3 */
/* 842 */ NdrFcShort( 0x38 ), /* 56 */
/* 844 */ NdrFcShort( 0xfffffec ), /* Offset=-20 (824) */
/* 846 */ NdrFcShort( 0x0 ), /* Offset= 0 (846) */
/* 848 */ 0x6, /* FC_SHORT */
0x6, /* FC_SHORT */
/* 850 */ 0x38, /* FC_ALIGNM4 */
0x8, /* FC_LONG */
/* 852 */ 0x8, /* FC_LONG */

```

```

FC_EMBEDDED_COMPLEX /*
/* 854 */ 0x4, /* 4 */
NdrFcShort( 0xfffffe0d ), /* Offset=
-499 (356) */
/* 858 */
0x5b, /* FC_END */
/* 860 */ NdrFcShort( 0xfffff02 ), /* Offset= -254 (606) */
/* 862 */
0x12, 0x8, /* FC_UP [simple_pointer] */
/* FC_BYTE */
/* 864 */ 0x1,
0x5c, /* FC_PAD */
/* 866 */
0x12, 0x8, /* FC_UP [simple_pointer] */
/* FC_SHORT */
/* 868 */ 0x6,
0x5c, /* FC_PAD */
/* 870 */
0x12, 0x8, /* FC_UP [simple_pointer] */
/* FC_LONG */
/* 872 */ 0x8,
0x5c, /* FC_PAD */
/* 874 */
0x12, 0x8, /* FC_UP [simple_pointer] */
/* FC_FLOAT */
/* 876 */ 0xa,
0x5c, /* FC_PAD */
/* 878 */
0x12, 0x8, /* FC_UP [simple_pointer] */
/* FC_DOUBLE */
/* 880 */ 0xc,
0x5c, /* FC_PAD */
/* 882 */
0x12, 0x0, /* FC_UP */
/* 884 */ NdrFcShort( 0xffffda4 ), /* Offset= -604 (280) */
/* 886 */
0x12, 0x10, /* FC_UP
[pointer_deref] */
/* 888 */ NdrFcShort( 0xffffda6 ), /* Offset= -602 (286) */
/* 890 */
0x12, 0x10, /* FC_UP
[pointer_deref] */
/* 892 */ NdrFcShort( 0xffffdbc ), /* Offset= -580 (312) */
/* 894 */
0x12, 0x10, /* FC_UP
[pointer_deref] */
/* 896 */ NdrFcShort( 0xffffdca ), /* Offset= -566 (330) */
/* 898 */
0x12, 0x10, /* FC_UP
[pointer_deref] */
/* 900 */ NdrFcShort( 0xffffdd8 ), /* Offset= -552 (348) */
/* 902 */
0x12, 0x10, /* FC_UP
[pointer_deref] */
/* 904 */ NdrFcShort( 0x2 ), /* Offset= 2 (906) */
/* 906 */
0x12, 0x0, /* FC_UP */
/* 908 */ NdrFcShort( 0x16 ), /* Offset= 22 (930) */
/* 910 */
0x15, /* FC_STRUCT */
0x7, /* 7 */
/* 912 */ NdrFcShort( 0x10 ), /* 16 */
/* 914 */ 0x6, /* FC_SHORT */
0x1, /* FC_BYTE */
/* 916 */ 0x1, /* FC_BYTE */
0x38, /* FC_ALIGNM4 */
/* 918 */ 0x8, /* FC_LONG */
0x39, /* FC_ALIGNM8 */
/* 920 */ 0xb, /* FC_HYPER */
0x5b, /* FC_END */
/* 922 */
0x12, 0x0, /* FC_UP */
/* 924 */ NdrFcShort( 0xfffff2 ), /* Offset= -14 (910) */
/* 926 */
0x12, 0x8, /* FC_UP [simple_pointer] */
/* FC_CHAR */
/* 928 */ 0x2,
0x5c, /* FC_PAD */
/* 930 */
0x1a, /*
FC_BOGUS_STRUCT */
0x7, /* 7 */
/* 932 */ NdrFcShort( 0x20 ), /* 32 */
/* 934 */ NdrFcShort( 0x0 ), /* 0 */
/* 936 */ NdrFcShort( 0x0 ), /* Offset= 0 (936) */
/* 938 */ 0x8, /* FC_LONG */
0x8, /* FC_LONG */
/* 940 */ 0x6, /* FC_SHORT */
0x6, /* FC_SHORT */
/* 942 */ 0x6, /* FC_SHORT */
0x6, /* FC_SHORT */
/* 944 */ 0x4c, /* FC_EMBEDDED_COMPLEX */
0x0, /* 0 */
/* 946 */ NdrFcShort( 0xffffc54 ), /* Offset= -940 (6) */
/* 948 */ 0x5c, /* FC_PAD */
0x5b, /* FC_END */
/* 950 */ 0xb4, /* FC_USER_MARSHAL */
0x83, /* 131 */
/* 952 */ NdrFcShort( 0x0 ), /* 0 */
/* 954 */ NdrFcShort( 0x18 ), /* 24 */
/* 956 */ NdrFcShort( 0x0 ), /* 0 */
/* 958 */ NdrFcShort( 0xffffc44 ), /* Offset= -956 (2) */
/* 960 */
0x11, 0x4, /* FC_RP [allocated_on_stack] */
/* 962 */ NdrFcShort( 0x6 ), /* Offset= 6 (968) */
/* 964 */
0x13, 0x0, /* FC_OP */
/* 966 */ NdrFcShort( 0xfffffdc ), /* Offset= -36 (930) */
/* 968 */ 0xb4, /* FC_USER_MARSHAL */
0x83, /* 131 */
/* 970 */ NdrFcShort( 0x0 ), /* 0 */
/* 972 */ NdrFcShort( 0x18 ), /* 24 */
/* 974 */ NdrFcShort( 0x0 ), /* 0 */
/* 976 */ NdrFcShort( 0xffffff4 ), /* Offset= -12 (964) */
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 */
};

#endif /* defined(_M_IA64) || defined(_M_AXP64)*/

```

## ***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 'Both'
            }
        }
    }
}

```

## ***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:
* 4.20.000 - 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 <sqlext.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";

static SQLHENV henv = SQL_NULL_HENV;
// ODBC environment handle

BOOL WINAPIENTRY 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;
                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,
        "No 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_erno == 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 for login
    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);

    {
        char buffer[128];

        // set some options affecting connection behavior
strcpy(buffer, "set nocount on set XACT_ABORT 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 && !INativeError !=
0)
            pODBCErr->m_NativeError = INativeError;

        // 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 (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 );
    }

    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_SSHORT, SQL_SMALLINT, 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_TRANS,
iTryCount);
}

void CTPCC_ODBC::InitNewOrderParams()
{
    if ( SQLAllocHandle(SQL_HANDLE_STMT, m_hdbc,
&m_hstmtNewOrder) != SQL_SUCCESS
        || SQLAllocHandle(SQL_HANDLE_DESC, m_hdbc,
&m_descNewOrderCols1) != SQL_SUCCESS
        || SQLAllocHandle(SQL_HANDLE_DESC, m_hdbc,
&m_descNewOrderCols2) != SQL_SUCCESS
    )
        ThrowError(CODBCERR::eAllocHandle);

    m_hstmt = m_hstmtNewOrder;

    if ( SQLSetStmtAttrW(m_hstmt, SQL_ATTR_APP_ROW_DESC,
m_descNewOrderCols1, SQL_IS_POINTER ) != SQL_SUCCESS )
        ThrowError(CODBCERR::eSetStmtAttr);

    int i = 0;
    if ( SQLBindParameter(m_hstmt, ++i, SQL_PARAM_INPUT,
SQL_C_SSHORT, SQL_SMALLINT, 0, 0, &m_txn.NewOrder.w_id, 0,
NULL) != SQL_SUCCESS
        || 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_SSHORT, SQL_SMALLINT, 0, 0,
&m_txn.NewOrder.OL[j].ol_supply_w_id, 0, NULL) != SQL_SUCCESS
            || SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT, SQL_C_SSHORT, SQL_SMALLINT, 0, 0,
&m_txn.NewOrder.OL[j].ol_quantity, 0, NULL) != SQL_SUCCESS
        )
            ThrowError(CODBCERR::eBindParam);
    }

    // set the bind offset pointer
    if ( SQLSetStmtAttrW(m_hstmt,
SQL_ATTR_ROW_BIND_OFFSET_PTR, &m_BindOffset,
SQL_IS_POINTER ) != SQL_SUCCESS )
        ThrowError(CODBCERR::eSetStmtAttr);

    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);

    // 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()
{
    int
    RETCODE
    int
    i;
    rc;
    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
            break;
        }
    }

    while (TRUE)
    {
        try
        {
            m_BindOffset = 0;
            rc = SQLExecDirectW(m_hstmt,
(SQLWCHAR*)szSqlTemplate, SQL_NTS);
            if (rc != SQL_SUCCESS && rc !=
SQL_SUCCESS_WITH_INFO)
                ThrowError(CODBCERR::eExecDirect);

            // Get order line results
            m_txn.NewOrder.total_amount = 0;
            for ( i = 0; i < m_txn.NewOrder.o_ol_cnt; i++)
                {
                    // set the bind offset value...
                    m_BindOffset = i *
                    sizeof(m_txn.NewOrder.OL[0]);

                    if ( SQLFetch(m_hstmt) ==
                    SQL_ERROR)
                        ThrowError(CODBCERR::eFetch);

                    // 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_TRANS,
            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_SSHORT, SQL_SMALLINT, 0, 0, &m_txn.Payment.w_id, 0, NULL)
!= SQL_SUCCESS
        || SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT, SQL_C_SSHORT, SQL_SMALLINT, 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),
SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i, SQL_C_CHAR,
&m_txn.Payment.w_street_2, sizeof(m_txn.Payment.w_street_2),
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),
SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i, SQL_C_CHAR,
&m_txn.Payment.d_street_2, sizeof(m_txn.Payment.d_street_2),
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),
SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i, SQL_C_CHAR,
&m_txn.Payment.c_street_1, sizeof(m_txn.Payment.c_street_1),
SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i, SQL_C_CHAR,
&m_txn.Payment.c_street_2, sizeof(m_txn.Payment.c_street_2),
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),
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()
{
    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)

```

```

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_TRANS,
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_SSHORT, SQL_SMALLINT, 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_SSHORT,
&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*)L" {call tpcc_orderstatus(?,?,?)", SQL_NTS);
                if ( ((rc == SQL_SUCCESS_WITH_INFO)
&& ( m_RowsFetched != 0)) || (rc == SQL_ERROR) )
                    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_TRANS,
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_SSHORT, SQL_SMALLINT, 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_TRANS,
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;

    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 PORDER_STATUS_DATA
BuffAddr_OrderStatus() { return &m_txn.OrderStatus; };

        void NewOrder          ();
        void Payment           ();
        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.
4.20.000
*
*      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 structure templates.
*
*      Change history:
*
*      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 dblink, 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
    short 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
    short 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
    short w_id;
    short d_id;
    long c_id;
    short c_d_id;
    short 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;
    short ol_supply_w_id;
    short ol_quantity;
    double ol_amount;
    TIMESTAMP_STRUCT ol_delivery_d;
} OL_ORDER_STATUS_DATA;

```

```

typedef struct
{
    // input params
    short    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
    short    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
    short         w_id; //delivery
    warehouse
    short         o_carrier_id; //carrier id
} DELIVERY_TRANSACTION;

```

```

typedef struct
{
    // input params
    short    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 Payment() = 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 (ms)
    int    DeltaT2; // keying time (ms)
    int    DeltaT3; // think time (ms)
    int    DeltaT4; // response time (ms)
    int    RTDelay; // response time delay
(ms)
    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 chosen for
this particular
    BYTE    TxnStatus; // completion status for
txn to indicate errors
    BYTE    reserved; // for word alignment
    TXN_DETAILS    TxnDetails; //
} TXN_RECORD_TPCC, *PTXN_RECORD_TPCC;

// TPC-C Deferred Delivery Txn Record Layout:
//
//Incorporating delivery transaction information into the above
//structure would increase the size of TXN_DETAILS from 8 to 42
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.
//
struct
//
{
JULIAN_TIME TS;
// timestamp of record
int
iPos; // byte position in file
//
}
RecMap[RecMapSize];
#define RecMapSize 200
} TXN_LOG_HEADER, *PTXN_LOG_HEADER;

/* Header of the sorted pointers blocks in Temp file (in merging). */
typedef struct BLOCK_HEADER {
long BlockPos;
__int64 CurPos;
DWORD BytesRead;
int nRecords;
BYTE *offset; /* offset of pointers to records in
the log file */
} BLOCK_HEADER, *PBLOCK_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

FILE            *tmpFile;
//temp file for merging sorted pieces
PBLOCK_HEADER  tmpHeaders;
//sorted pointers block header
BYTE           **recPointers;
//record pointer buffers for each sorted block
PTXN_RECORD_HEADER *recBuffers;
//record buffers for each sorted block
int            *PointersRead;
//# of pointers processed in each block
BOOL          *BlockAvailable;
//whether to check a particular block for jmin

int            nBlocks;
int            jmin;
//index (block-wise) of the lowest timestamp record
int            iAvgRecordLen;
//average record length

int            iSortedReturnedCount;
//keeps track of the # of sorted records returned through GetSortedRecord()

int Write(BYTE *ptr, DWORD Size);
static void LogFileIO(CTxnLog *);

void LoadBuffers(int j);
//used in sort/merge to load record buffers

public:
CTxnLog::CTxnLog(LPCTSTR szFileName, DWORD
dwOpts);
~CTxnLog(void);

int WriteToLog(PTXN_RECORD_TPCC pTxnRcrd);
int WriteToLog(PTXN_RECORD_TPCC_DELIV_DEF
pTxnRcrd);
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_TIME SeekTimeT0, BOOL bSkipCtrlRecs =
FALSE);

int Sort(void);
PTXN_RECORD_HEADER GetSortedRecord();

```

```

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);
};
};

```

### webclnt.dsp

```

# Microsoft Developer Studio Project File - Name="webclnt" - Package
Owner=<4>
# Microsoft Developer Studio Generated Build File, Format Version 5.00
# ** DO NOT EDIT **

# TARGETTYPE "Win32 (x86) Application" 0x0101

CFG=webclnt - Win32 Release
!MESSAGE This is not a valid makefile. To build this project using NMAKE,
!MESSAGE use the Export Makefile command and run
!MESSAGE
!MESSAGE NMAKE /f "Webclnt.mak".
!MESSAGE

```

```

!MESSAGE You can specify a configuration when running NMAKE
!MESSAGE by defining the macro CFG on the command line. For example:
!MESSAGE
!MESSAGE NMAKE /f "WebcInt.mak" CFG="webcInt - Win32 Release"
!MESSAGE
!MESSAGE Possible choices for configuration are:
!MESSAGE
!MESSAGE "webcInt - Win32 Release" (based on "Win32 (x86) Application")
!MESSAGE "webcInt - Win32 Debug" (based on "Win32 (x86) Application")
!MESSAGE

```

```

# Begin Project
# PROP Scc_ProjName ""
# PROP Scc_LocalPath ""
CPP=cl.exe
MTL=midl.exe
RSC=rc.exe

```

```

!IF "$(CFG)" == "webcInt - 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 ".\Release"
# PROP Intermediate_Dir ".\Release"
# PROP Target_Dir ""
# ADD BASE CPP /nologo /W3 /GX /O2 /D "WIN32" /D "NDEBUG" /D
" _WINDOWS" /YX /c
# ADD CPP /nologo /W3 /GX /O2 /D "WIN32" /D "NDEBUG" /D
" _WINDOWS" /YX /FD /c
# ADD BASE MTL /nologo /D "NDEBUG" /win32
# ADD MTL /nologo /D "NDEBUG" /mktyplib203 /win32
# ADD BASE RSC /I 0x409 /d "NDEBUG"
# ADD RSC /I 0x409 /d "NDEBUG"
BSC32=bscmake.exe
# ADD BASE BSC32 /nologo
# ADD BSC32 /nologo
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 odbccp32.lib
odbccp32.lib /nologo /subsystem:windows /machine:I386
# ADD LINK32 kernel32.lib user32.lib gdi32.lib winspool.lib comdlg32.lib
advapi32.lib shell32.lib ole32.lib oleaut32.lib uuid.lib odbccp32.lib
/nologo /subsystem:windows /machine:I386

```

```

!ELSEIF "$(CFG)" == "webcInt - 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 ".\Debug"
# PROP Intermediate_Dir ".\Debug"
# PROP Target_Dir ""
# ADD BASE CPP /nologo /W3 /Gm /GX /Zi /Od /D "WIN32" /D "_DEBUG"
/D "_WINDOWS" /YX /c
# ADD CPP /nologo /W3 /Gm /GX /Zi /Od /D "WIN32" /D "_DEBUG" /D
" _WINDOWS" /YX /FD /c
# ADD BASE MTL /nologo /D "_DEBUG" /win32
# ADD MTL /nologo /D "_DEBUG" /mktyplib203 /win32
# ADD BASE RSC /I 0x409 /d "_DEBUG"

```

```

# ADD RSC /I 0x409 /d "_DEBUG"
BSC32=bscmake.exe
# ADD BASE BSC32 /nologo
# ADD BSC32 /nologo
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 odbccp32.lib
odbccp32.lib /nologo /subsystem:windows /debug /machine:I386
# ADD LINK32 kernel32.lib user32.lib gdi32.lib winspool.lib comdlg32.lib
advapi32.lib shell32.lib ole32.lib oleaut32.lib uuid.lib odbccp32.lib
/nologo /subsystem:windows /debug /machine:I386

```

```

!ENDIF

```

```

# Begin Target

# Name "webcInt - Win32 Release"
# Name "webcInt - Win32 Debug"
# End Target
# End Project

```

## Stored Procedures

### *neword.sql*

```

-- File: NEWORD.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.42
-- Copyright Microsoft, 2002
-- Purpose: Creates new order transaction stored procedure
--
-- Interface Level: 4.10.000

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,
@s_w_id1 int = 0, @ol_qty1 smallint = 0,
@i_id2 int = 0,
@s_w_id2 int = 0, @ol_qty2 smallint = 0,
@i_id3 int = 0,
@s_w_id3 int = 0, @ol_qty3 smallint = 0,
@i_id4 int = 0,
@s_w_id4 int = 0, @ol_qty4 smallint = 0,
@i_id5 int = 0,
@s_w_id5 int = 0, @ol_qty5 smallint = 0,
@i_id6 int = 0,
@s_w_id6 int = 0, @ol_qty6 smallint = 0,
@i_id7 int = 0,
@s_w_id7 int = 0, @ol_qty7 smallint = 0,
@i_id8 int = 0,
@s_w_id8 int = 0, @ol_qty8 smallint = 0,
@i_id9 int = 0,
@s_w_id9 int = 0, @ol_qty9 smallint = 0,
@i_id10 int = 0,
@s_w_id10 int = 0, @ol_qty10 smallint = 0,

```

```

@s_w_id11 int = 0, @ol_qty11 smallint = 0,
@s_w_id12 int = 0, @ol_qty12 smallint = 0,
@s_w_id13 int = 0, @ol_qty13 smallint = 0,
@s_w_id14 int = 0, @ol_qty14 smallint = 0,
@s_w_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,
    @o_entry_d = getdate(),
    @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

```

```
@i_id11 int = 0,
```

```
@i_id12 int = 0,
```

```
@i_id13 int = 0,
```

```
@i_id14 int = 0,
```

```
@i_id15 int = 0,
```

```

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,

```

```

@li_qty = case @li_no
when 1 then @ol_qty1
when 2 then @ol_qty2
when 3 then @ol_qty3
when 4 then @ol_qty4
when 5 then @ol_qty5
when 6 then @ol_qty6
when 7 then @ol_qty7
when 8 then @ol_qty8
when 9 then @ol_qty9
when 10 then @ol_qty10
when 11 then @ol_qty11
when 12 then @ol_qty12
when 13 then @ol_qty13
when 14 then @ol_qty14
when 15 then @ol_qty15
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 + @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

```

```

s_dist_01          when 1 then
s_dist_02          when 2 then
s_dist_03          when 3 then
s_dist_04          when 4 then
s_dist_05          when 5 then
s_dist_06          when 6 then
s_dist_07          when 7 then
s_dist_08          when 8 then
s_dist_09          when 9 then
s_dist_10          when 10 then
                    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,
                                                    1899',
                                                    @li_qty,
                                                    @i_price
                                                    * @li_qty,
                                                    @s_dist)

-- send line-item data to client
                    select @i_name,
                           @s_quantity,
                           b_g = 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
                    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

```

**payment.sql**

```

-- File: PAYMENT.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.42
-- Copyright Microsoft, 2002
-- Purpose: Creates payment transaction stored procedure
--
-- Interface Level: 4.10.000

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      @cnt = count(*)
                from        customer (repeatable read)
                where       c_last = @c_last and
                           c_w_id = @c_w_id and
                           c_d_id = @c_d_id

                select      @val = (@cnt + 1) / 2
                set         rowcount @val

                select      @c_id = c_id
                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_last, c_first

                set         rowcount 0

        end

-- get customer info and update balances

                update      customer
                set         @c_balance = c_balance + @h_amount,
                           c_payment_cnt = c_payment_cnt + 1,
                           c_ytd_payment = c_ytd_payment + @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

```

## ordstat.sql

```

-- File:   ORDSTAT.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.42
-- Copyright Microsoft, 2002
-- Purpose: Creates order status transaction stored procedure
--
-- Interface Level: 4.10.000

```

```

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 (repeatable read)
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 (repeatable read)
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 (repeatable read)
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

## delivery.sql

```

-- File: DELIVERY.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.42
-- Copyright Microsoft, 2002
-- Purpose: Creates delivery transaction stored procedure
--
-- Interface Level: 4.10.000

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

select @d_id = 0

begin tran d

    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,

where @c_id = o_c_id
      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 = getdate(),
  @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

## Stocklev.sql

```

-- File: STOCKLEV.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.42
-- Copyright Microsoft, 2002
-- Purpose: Creates stock level transaction stored procedure
--
-- Interface Level: 4.10.000

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

go

```

## version.sql

```

-- File: VERSION.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.41
-- Copyright Microsoft, 2001
-- Purpose: Returns version level of TPC-C stored procs
-- Note: Always update the return value of this proc for
-- any interface changes or 'must have' bug fixes.
--
-- The value returned by this SP defines the 'interface level',
-- which must match between the stored procs and the client code.
-- The interface level may be down rev from the current kit. This
-- indicates that the interface hasn't changed since that version.

use tpcc
go

```

```
if exists ( select name from sysobjects where name = 'tpcc_version' )
    drop procedure tpcc_version
go

create proc tpcc_version
as
declare    @version char(8)

begin
    select @version = '4.10.000'
    select @version as 'Version'
end

go
```

# Appendix B: Database Design

## Database Build

### createdb.sql

```
-- File:   CREATEDB.SQL
--        Microsoft TPC-C Benchmark Kit Ver. 4.41
--        Copyright Microsoft, 2001
-- Purpose: Creates tpcc database and backup files

use master
go

--        Create temporary table for timing

if exists ( select name from sysobjects where name = 'tpcc_timer' )
    drop table tpcc_timer
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                = MSSQL_tpcc_root,
    FILENAME             = 'F:\MSSQL_tpcc_root.mdf',
    SIZE                 = 8MB,
    FILEGROWTH           = 0),
FILEGROUP    MSSQL_misc_fg
(
    NAME                = MSSQL_misc7,
    FILENAME             = 'C:\mp\m7\',
    SIZE                 = 24500MB,
    FILEGROWTH           = 0),
(
    NAME                = MSSQL_misc8,
    FILENAME             = 'C:\mp\m8\',
    SIZE                 = 24500MB,
    FILEGROWTH           = 0),
(
    NAME                = MSSQL_misc9,
    FILENAME             = 'C:\mp\m9\',
    SIZE                 = 24500MB,
    FILEGROWTH           = 0),
(
    NAME                = MSSQL_misc10,
    FILENAME             = 'C:\mp\m10\',
    SIZE                 = 24500MB,
    FILEGROWTH           = 0),
(
    NAME                = MSSQL_misc11,
    FILENAME             = 'C:\mp\m11\',
    SIZE                 = 24500MB,
    FILEGROWTH           = 0),
(
    NAME                = MSSQL_misc12,
```

```
FILENAME             = 'C:\mp\m12\',
SIZE                 = 24500MB,
FILEGROWTH           = 0),
(
    NAME                = MSSQL_misc13,
    FILENAME             = 'C:\mp\m13\',
    SIZE                 = 24500MB,
    FILEGROWTH           = 0),
(
    NAME                = MSSQL_misc14,
    FILENAME             = 'C:\mp\m14\',
    SIZE                 = 24500MB,
    FILEGROWTH           = 0),
(
    NAME                = MSSQL_misc15,
    FILENAME             = 'C:\mp\m15\',
    SIZE                 = 24500MB,
    FILEGROWTH           = 0),
(
    NAME                = MSSQL_misc16,
    FILENAME             = 'C:\mp\m16\',
    SIZE                 = 24500MB,
    FILEGROWTH           = 0),
(
    NAME                = MSSQL_misc17,
    FILENAME             = 'C:\mp\m17\',
    SIZE                 = 24500MB,
    FILEGROWTH           = 0),
(
    NAME                = MSSQL_misc18,
    FILENAME             = 'C:\mp\m18\',
    SIZE                 = 24500MB,
    FILEGROWTH           = 0),
(
    NAME                = MSSQL_misc19,
    FILENAME             = 'C:\mp\m19\',
    SIZE                 = 24500MB,
    FILEGROWTH           = 0),
(
    NAME                = MSSQL_misc20,
    FILENAME             = 'C:\mp\m20\',
    SIZE                 = 24500MB,
    FILEGROWTH           = 0),
(
    NAME                = MSSQL_misc21,
    FILENAME             = 'C:\mp\m21\',
    SIZE                 = 24500MB,
    FILEGROWTH           = 0),
(
    NAME                = MSSQL_misc22,
    FILENAME             = 'C:\mp\m22\',
    SIZE                 = 24500MB,
    FILEGROWTH           = 0),
(
    NAME                = MSSQL_misc23,
    FILENAME             = 'C:\mp\m23\',
    SIZE                 = 24500MB,
    FILEGROWTH           = 0),
(
    NAME                = MSSQL_misc24,
    FILENAME             = 'C:\mp\m24\',
    SIZE                 = 24500MB,
    FILEGROWTH           = 0),
(
    NAME                = MSSQL_misc25,
    FILENAME             = 'C:\mp\m25\',
    SIZE                 = 24500MB,
    FILEGROWTH           = 0),
(
    NAME                = MSSQL_misc26,
    FILENAME             = 'C:\mp\m26\',
    SIZE                 = 24500MB,
    FILEGROWTH           = 0),
FILEGROUP    MSSQL_cs_fg
(
    NAME                = MSSQL_cs7,
    FILENAME             = 'c:\mp\c7\',
    SIZE                 = 44500MB,
    FILEGROWTH           = 0),
(
    NAME                = MSSQL_cs8,
    FILENAME             = 'c:\mp\c8\',
    SIZE                 = 44500MB,
```

```

FILEGROWTH = 0),
( NAME = MSSQL_cs9,
  FILENAME = 'c:\mp\c9\',
  SIZE = 44500MB,
  FILEGROWTH = 0),
( NAME = MSSQL_cs10,
  FILENAME = 'c:\mp\c10\',
  SIZE = 44500MB,
  FILEGROWTH = 0),
( NAME = MSSQL_cs11,
  FILENAME = 'c:\mp\c11\',
  SIZE = 44500MB,
  FILEGROWTH = 0),
( NAME = MSSQL_cs12,
  FILENAME = 'c:\mp\c12\',
  SIZE = 44500MB,
  FILEGROWTH = 0),
( NAME = MSSQL_cs13,
  FILENAME = 'c:\mp\c13\',
  SIZE = 44500MB,
  FILEGROWTH = 0),
( NAME = MSSQL_cs14,
  FILENAME = 'c:\mp\c14\',
  SIZE = 44500MB,
  FILEGROWTH = 0),
( NAME = MSSQL_cs15,
  FILENAME = 'c:\mp\c15\',
  SIZE = 44500MB,
  FILEGROWTH = 0),
( NAME = MSSQL_cs16,
  FILENAME = 'c:\mp\c16\',
  SIZE = 44500MB,
  FILEGROWTH = 0),
( NAME = MSSQL_cs17,
  FILENAME = 'c:\mp\c17\',
  SIZE = 44500MB,
  FILEGROWTH = 0),
( NAME = MSSQL_cs18,
  FILENAME = 'c:\mp\c18\',
  SIZE = 44500MB,
  FILEGROWTH = 0),
( NAME = MSSQL_cs19,
  FILENAME = 'c:\mp\c19\',
  SIZE = 44500MB,
  FILEGROWTH = 0),
( NAME = MSSQL_cs20,
  FILENAME = 'c:\mp\c20\',
  SIZE = 44500MB,
  FILEGROWTH = 0),
( NAME = MSSQL_cs21,
  FILENAME = 'c:\mp\c21\',
  SIZE = 44500MB,
  FILEGROWTH = 0),
( NAME = MSSQL_cs22,
  FILENAME = 'c:\mp\c22\',
  SIZE = 44500MB,
  FILEGROWTH = 0),
( NAME = MSSQL_cs23,
  FILENAME = 'c:\mp\c23\',
  SIZE = 44500MB,
  FILEGROWTH = 0),
( NAME = MSSQL_cs24,
  FILENAME = 'c:\mp\c24\',
  SIZE = 44500MB,
  FILEGROWTH = 0),
( NAME = MSSQL_cs25,
  FILENAME = 'c:\mp\c25\',
  SIZE = 44500MB,

```

```

FILEGROWTH = 0),
( NAME = MSSQL_cs26,
  FILENAME = 'c:\mp\c26\',
  SIZE = 44500MB,
  FILEGROWTH = 0)

LOG ON
( NAME = MSSQL_tpcc_log,
  FILENAME = 'E:',
  SIZE = 417000MB,
  FILEGROWTH = 0)

COLLATE Latin1_General_BIN
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

```

### **dbop1.sql**

```

-- File: DBOPT1.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.41
-- Copyright Microsoft, 2001
-- 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.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.41
-- Copyright Microsoft, 2001
-- Purpose: Resets database options after data load

```

```

exec sp_dboption tpcc,'select into/bulkcopy',false
exec sp_dboption tpcc,'trunc. log on chkpt.',false
exec sp_dboption tpcc,'torn page detection',false
GO

```

```

USE tpcc
GO

CHECKPOINT

```

```

GO

sp_configure 'allow updates',1
GO

RECONFIGURE WITH OVERRIDE
GO

DECLARE          @msg          varchar(50)

--
--
-- OPTIONS FOR SQL SERVER 2000 --
-- Set option values for user-defined indexes --
--
--

SET          @msg          = ''
PRINT        @msg
SET          @msg          = 'Setting SQL Server indexoptions'
PRINT        @msg
SET          @msg          = ''
PRINT        @msg

EXEC sp_indexoption          'customer', 'DisallowPageLocks', TRUE
EXEC sp_indexoption          'district', 'DisallowPageLocks', TRUE
EXEC sp_indexoption          'warehouse',          'DisallowPageLocks',
TRUE
EXEC sp_indexoption          'stock',          'DisallowPageLocks', TRUE
EXEC sp_indexoption          'order_line',          'DisallowRowLocks',
TRUE
EXEC sp_indexoption          'orders',          'DisallowRowLocks', TRUE
EXEC sp_indexoption          'new_order',          'DisallowRowLocks',
TRUE
EXEC sp_indexoption          'item',          'DisallowRowLocks',
TRUE
EXEC sp_indexoption          '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('warehouse')          = id OR
      object_id('district')          = id OR
      object_id('customer')          = id OR
      object_id('stock')          = id OR
      object_id('orders')          = id OR
      object_id('order_line')= id OR
      object_id('history')          = id OR
      object_id('new_order')          = id OR
      object_id('item')          = id

ORDER BY lockflags asc
GO

sp_configure 'allow updates',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 'district', 'pintable',true
EXEC sp_tableoption 'warehouse',          'pintable',true
EXEC sp_tableoption 'new_order',          'pintable',true
EXEC sp_tableoption 'item',          'pintable',true
GO

```

### ***idxcuscl.sql***

```

-- File:  IDXCUSCL.SQL
--      Microsoft TPC-C Benchmark Kit Ver. 4.41
--      Copyright Microsoft, 2001
-- Purpose:  Creates clustered index on customer table

use tpcc
go

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

if exists ( select name from sysindexes where name = 'customer_c1' )
drop index customer.customer_c1

create unique clustered index customer_c1 on customer(c_w_id, c_d_id, c_id)
on MSSQL_cs_fg

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. 4.41
--      Copyright Microsoft, 2001
-- Purpose:  Creates non-clustered index on customer table

use tpcc
go

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

if exists ( select name from sysindexes where name = 'customer_nc1' )
drop index customer.customer_nc1

create unique nonclustered index customer_nc1 on customer(c_w_id, c_d_id,
c_last, c_first, c_id)
on MSSQL_cs_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. 4.41
--       Copyright Microsoft, 2001
-- Purpose: Creates clustered index on district table

use tpc
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 = 'district_c1' )
    drop index district.district_c1

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

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

go
```

## **idxhiscl.sql**

```
-- File:  IDXHISCL.SQL
--       Microsoft TPC-C Benchmark Kit Ver. 4.41
--       Copyright Microsoft, 2001
-- Purpose: Creates clustered index on history table
--
-- CAUTION: *****
-- CAUTION: This index is only beneficial for systems
-- CAUTION: with 8 or more processors.
-- CAUTION: It may negatively impact performance on
-- CAUTION: on systems with less than 8 processors.
-- CAUTION: *****

use tpc
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 = 'history_c1' )
    drop index history.history_c1

create unique clustered index history_c1 on history(h_c_w_id, h_date,
h_c_d_id, h_c_id, h_amount)
on MSSQL_misc_fg

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

go
```

## **idxitmcl.sql**

```
-- File:  IDXITMCL.SQL
--       Microsoft TPC-C Benchmark Kit Ver. 4.41
--       Copyright Microsoft, 2001
-- Purpose: Creates clustered index on item table

use tpc
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 item.item_c1

create unique clustered index item_c1 on item(i_id)
on MSSQL_misc_fg

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

go
```

## **idxnodcl.sql**

```
-- File:  IDXNODCL.SQL
--       Microsoft TPC-C Benchmark Kit Ver. 4.41
--       Copyright Microsoft, 2001
-- Purpose: Creates clustered index on new_order table

use tpc
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 = 'new_order_c1' )
    drop index new_order.new_order_c1

create unique clustered index new_order_c1 on new_order(no_w_id, no_d_id,
no_o_id)
on MSSQL_misc_fg

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

go
```

## **idxodlcl.sql**

```
-- File:  IDXODLCL.SQL
--       Microsoft TPC-C Benchmark Kit Ver. 4.41
--       Copyright Microsoft, 2001
-- Purpose: Creates clustered index on order_line table
```

```

use tpcc
go

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

if exists ( select name from sysindexes where name = 'order_line_c1' )
    drop index order_line.order_line_c1

create unique clustered index order_line_c1 on order_line(ol_w_id, ol_d_id,
ol_o_id, ol_number)
    on MSSQL_misc_fg

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

go

```

### ***idxordcl.sql***

```

-- File:  IDXORDCL.SQL
--      Microsoft TPC-C Benchmark Kit Ver. 4.41
--      Copyright Microsoft, 2001
-- Purpose:  Creates clustered index on orders table

```

```

use tpcc
go

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

if exists ( select name from sysindexes where name = 'orders_c1' )
    drop index orders.orders_c1

create unique clustered index orders_c1 on orders(o_w_id, o_d_id, o_id)
    on MSSQL_misc_fg

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

Go

```

### ***idxstkcl.sql***

```

-- File:  IDXSTKCL.SQL
--      Microsoft TPC-C Benchmark Kit Ver. 4.41
--      Copyright Microsoft, 2001
-- Purpose:  Creates clustered index on stock table

```

```

use tpcc
go

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

```

```

if exists ( select name from sysindexes where name = 'stock_c1' )
    drop index stock.stock_c1

create unique clustered index stock_c1 on stock(s_i_id, s_w_id)
    on MSSQL_cs_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. 4.41
--      Copyright Microsoft, 2001
-- Purpose:  Creates clustered index on warehouse table

```

```

use tpcc
go

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

if exists ( select name from sysindexes where name = 'warehouse_c1' )
    drop index warehouse.warehouse_c1

create unique clustered index warehouse_c1 on warehouse(w_id)
    with fillfactor=100 on MSSQL_misc_fg

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

go

```

### ***sqlshutdown.sql***

```

-- File:  SQLSHUTDOWN.SQL
--      Microsoft TPC-C Benchmark Kit Ver. 4.41
--      Copyright Microsoft, 2001
-- Purpose:  Checkpoints tpcc database and issues a shutdown
--

```

```

use tpcc
go
checkpoint
go
shutdown
go

```

### ***Tables.sql***

```

-- File:  TABLES.SQL
--      Microsoft TPC-C Benchmark Kit Ver. 4.42
--      Copyright Microsoft, 2002
-- Purpose:  Creates TPC-C tables

```



```

SET ANSI_NULL_DFLT_OFF ON
go

use tpcc
go

--
-- Remove all existing TPC-C tables
--

if exists ( select name from sysobjects where name = 'warehouse' )
    drop table warehouse
go
if exists ( select name from sysobjects where name = 'district' )
    drop table district
go
if exists ( select name from sysobjects where name = 'customer' )
    drop table customer
go
if exists ( select name from sysobjects where name = 'history' )
    drop table history
go
if exists ( select name from sysobjects where name = 'new_order' )
    drop table new_order
go
if exists ( select name from sysobjects where name = 'orders' )
    drop table orders
go
if exists ( select name from sysobjects where name = 'order_line' )
    drop table order_line
go
if exists ( select name from sysobjects where name = 'item' )
    drop table item
go
if exists ( select name from sysobjects where name = 'stock' )
    drop table stock
go

--
-- Create new tables
--

create table 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 MSSQL_misc_fg
go

create table 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 MSSQL_misc_fg
go

create table 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 MSSQL_cs_fg
go

create table 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)
) on MSSQL_misc_fg
go

create table new_order
(
    no_o_id             int,
    no_d_id             tinyint,
    no_w_id             int
) on MSSQL_misc_fg
go

create table 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 MSSQL_misc_fg
go

create table 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 MSSQL_misc_fg
go

create table item
(
        i_id            int,
        i_im_id         int,
        i_name          char(24),
        i_price         numeric(5,2),
        i_data          char(50)
) on MSSQL_misc_fg
go

create table 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 MSSQL_cs_fg
Go

```

## Load Source Code

### getargs.c

```

//      File:          GETARGS.C
//
//      Microsoft TPC-C Kit Ver. 4.41
//      Copyright Microsoft, 1996, 1997,
1998, 1999, 2000, 2001
//      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("[%d]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->log_path        =
LOG_PATH;
        pargs->pack_size       = DEFALDPACKSIZE;
        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 '?': /* Fall through */
                                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':
    {
        pargs->tables_all =
FALSE;
        if (strcmp(ptr+2,"item")
== 0)
            pargs->table_item = TRUE;
        else if
            (strcmp(ptr+2,"warehouse") == 0)
            pargs->table_warehouse = TRUE;
        else if
            (strcmp(ptr+2,"customer") == 0)
            pargs->table_customer = TRUE;
        else if
            (strcmp(ptr+2,"orders") == 0)
            pargs->table_orders = TRUE;
        else
        {
            printf("\nUnrecognized
command");
            GetArgsLoaderUsage();
            exit(1);
        }
        break;
    }

case 'f':
    pargs->loader_res_file = ptr+2;
    break;

case 'L':
    pargs->log_path = ptr+2;
    break;

case 'p':
    pargs->pack_size = atol(ptr+2);
    break;

case 'i':
    pargs->build_index = atol(ptr+2);
    break;

case 'o':
    pargs->index_order = atol(ptr+2);
    break;

case 'c':
    pargs->scale_down = atol(ptr+2);
    break;

case 'd':
    pargs->index_script_path = ptr+2;
    break;

default:
    GetArgsLoaderUsage();
    exit(-1);
    break;
}

/* check for required args */
if (pargs->num_warehouses == UNDEF )
{
    printf("Number of Warehouses is required\n");
    exit(-2);
}

return;
}

//=====
//
// Function name: GetArgsLoaderUsage
//
//=====

void GetArgsLoaderUsage()
{
#ifdef DEBUG
    printf("[%ld]DBG: Entering GetArgsLoaderUsage()\n", (int)
GetCurrentThreadId());
#endif

    printf("TPCCLDR:\n\n");
    printf("Parameter                Default\n");
    printf("-----\n\n");
    printf("-W Number of Warehouses to Load      Required\n");
    printf("-S Server                               %s\n", SERVER);
    printf("-U Username                             %s\n", USER);
    printf("-P Password                             %s\n", PASSWORD);
    printf("-D Database                             %s\n", DATABASE);
    printf("-b Batch Size                           %ld\n", (long)
BATCH);
    printf("-p TDS packet size                       %ld\n", (long)
DEFLDPACKSIZE);
    printf("-f Loader Results Output Filename      %s\n",
LOADER_RES_FILE);
    printf("-s Starting Warehouse                   %ld\n", (long)
DEF_STARTING_WAREHOUSE);
    printf("-i Build Option (data = 0, data and index = 1) %ld\n",
(long) BUILD_INDEX);
    printf("-o Cluster Index Build Order (before = 1, after = 0) %ld\n",
(long) INDEX_ORDER);
    printf("-c Build Scaled Database (normal = 0, tiny = 1) %ld\n",
(long) SCALE_DOWN);
    printf("-d Index Script Path                     %s\n",
INDEX_SCRIPT_PATH);
    printf("-t Table to Load                         all tables\n");
    printf(" [item|warehouse|customer|orders]\n");
    printf(" Notes: \n");
}

```

```

printf(" - the '-t' parameter may be included multiple times to \n");
printf(" specify multiple tables to be loaded \n");
printf(" - 'item' loads ITEM table \n");
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");

printf("\nNote: Command line switches are case sensitive.\n");

exit(0);
}

```

## random.c

```

// File: RANDOM.C
// Microsoft TPC-C Kit Ver. 4.41
// Copyright Microsoft, 1996, 1997,
1998, 1999, 2000, 2001
// 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("[%d]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("[%d]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("[%d]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

//Original 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. 4.41
//      Copyright Microsoft, 1996, 1997,
1998, 1999, 2000, 2001
//      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("[%d]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("[%d]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 <%d> out of range
(0,999)\n", num);
        exit(-1);
    }

#ifdef DEBUG
    printf("[%d]DBG: LastName: num = [%d] ==> [%d][%d][%d]\n",
                (int) GetCurrentThreadId(), num, num/100,
(num/10)%10, num%10);
    printf("[%d]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("[%d]DBG: Entering MakeAlphaString()\n", (int)
GetCurrentThreadId());
#endif

    len= RandomNumber(x, y);

    for (i=0; i<len; i++)
    {
        cc = chArray[RandomNumber(0, chArrayMax)];
        str[i] = cc;
    }
    //if ( len < z )
    //    memset(str+len, ' ', z - len);
    str[len] = 0;

    return len;
}

//=====
//
// Function name: MakeOriginalAlphaString
//
//=====

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, '', ADDRESS_LEN+1);
    memset(street_2, '', ADDRESS_LEN+1);
    memset(city, '', ADDRESS_LEN+1);

    street_1[ADDRESS_LEN+1] = 0;
    street_2[ADDRESS_LEN+1] = 0;

```

```

city[ADDRESS_LEN+1] = 0;

memset(state, '', STATE_LEN+1);
state[STATE_LEN+1] = 0;

memset(zip, '', 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, '', max - len);
    name[max] = 0;

    return;
}

```

### time.c

```

// File: TIME.C
// Microsoft TPC-C Kit Ver. 4.41
// Copyright Microsoft, 1996, 1997,
1998, 1999, 2000, 2001
// 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. 4.41
// Copyright Microsoft, 1996, 1997,
1998, 1999, 2000, 2001
// Purpose: Header file for TPC-C database loader

```

```

// Build number of TPC Benchmark Kit
#define TPCKIT_VER "4.41"

```

```

// 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 DEFLDPACKSIZE 32768
#define LOADER_RES_FILE
"C:\\MSTPCC.440\\SETUP\\logs\\load.out"
#define LOG_PATH "C:\\MSTPCC.440\\SETUP\\LOGS\\";
#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          table_warehouse;
set if loading WAREHOUSE, DISTRICT, and STOCK
        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          *log_path;
        char          *synch_servername;
        long          case_sensitivity;
        long          starting_warehouse;
        long          build_index;
        long          index_order;
        long          scale_down;
        char          *index_script_path;
} TPCCLDR_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     30
#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: TPCCLDR.C
// Microsoft TPC-C Kit Ver. 4.41
// Copyright Microsoft, 1996, 1997,
// 1998, 1999, 2000, 2001
// 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 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;
    short       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;
    short       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;
    short               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];
    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;

TPCCCLR_ARGS  *aptr, args;

//=====
//=====
//

```

```

// 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*****");
    printf("\n*");
    printf("\n* Microsoft SQL Server");
    printf("\n*");
    printf("\n*");
    printf("\n* TPC-C BENCHMARK KIT: Database loader");
    printf("\n* Version %s", TPCCKIT_VER);
    printf("\n*");

    printf("\n*****\n\n");

    // process command line arguments

    apr = &args;
    GetArgsLoader(argc, argv, apr);

    // 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        bcpint[128];
    char        err_log_path[256];

    // Seed with unique number
    seed(1);

    printf("Loading item table...\n");

    // if build index before load
    if ((aptr->build_index == 1) && (aptr->index_order == 1))
        BuildIndex("idxitmcl");

    InitString(i_name, I_NAME_LEN+1);
    InitString(i_data, I_DATA_LEN+1);

    sprintf(name, "%s.%s", aptr->database, "item");

    //rc = bcp_init(i_hdbc1, name, NULL, "logs\item.err", DB_IN);
    strcpy(err_log_path, aptr->log_path);
    strcat(err_log_path, "item.err");
    rc = bcp_init(i_hdbc1, name, NULL, err_log_path, 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");
            rc = bcp_control(i_hdbc1, BCPHINTS, (void*) bcphint);
            if (rc != SUCCEEDED)
                HandleErrorDBC(i_hdbc1);
        }

        rc = bcp_bind(i_hdbc1, (BYTE *) &i_id, 0, SQL_VARLEN_DATA,
NULL, 0, SQLINT4, 1);
        if (rc != SUCCEEDED)
            HandleErrorDBC(i_hdbc1);

        rc = bcp_bind(i_hdbc1, (BYTE *) &i_im_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 2);
        if (rc != SUCCEEDED)
            HandleErrorDBC(i_hdbc1);

        rc = bcp_bind(i_hdbc1, (BYTE *) i_name, 0, I_NAME_LEN,
NULL, 0, 0, 3);
        if (rc != SUCCEEDED)
            HandleErrorDBC(i_hdbc1);

        rc = bcp_bind(i_hdbc1, (BYTE *) &i_price, 0,
SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 4);
        if (rc != SUCCEEDED)
            HandleErrorDBC(i_hdbc1);

        rc = bcp_bind(i_hdbc1, (BYTE *) i_data, 0, I_DATA_LEN, NULL,
0, 0, 5);
        if (rc != SUCCEEDED)
            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 != SUCCEEDED)
                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("idxitmcl");
}

//=====
//=====
// Function : LoadWarehouse
//
// Loads WAREHOUSE table and loads Stock and District as Warehouses are
// created
//=====
//=====

void LoadWarehouse()
{
    short   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    bcphint[128];
    char    err_log_path[256];

    // Seed with unique number
    seed(2);

    printf("Loading warehouse table...\n");

    // if build index before load...
    if ((aptr->build_index == 1) && (aptr->index_order == 1))
        BuildIndex("idxwarc1");

    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, "warehouse");

    //rc = bcp_init(w_hdbc1, name, NULL, "logs\\whouse.err", DB_IN);

    strcpy(err_log_path, aptr->log_path);
    strcat(err_log_path, "whouse.err");
    rc = bcp_init(w_hdbc1, name, NULL, err_log_path, DB_IN);
    if (rc != SUCCEEDED)
        HandleErrorDBC(w_hdbc1);

    if ((aptr->build_index == 1) && (aptr->index_order == 1))
    {
        sprintf(bcphint, "tablock, order (w_id),
ROWS_PER_BATCH = %d", aptr->num_warehouses);
        rc = bcp_control(w_hdbc1, BCPHINTS, (void*)
bcphint);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);
    }
}

```

```

        rc = bcp_bind(w_hdbc1, (BYTE *) &w_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 1);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) w_name, 0, W_NAME_LEN,
NULL, 0, 0, 2);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) w_street_1, 0, ADDRESS_LEN,
NULL, 0, 0, 3);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) w_street_2, 0, ADDRESS_LEN,
NULL, 0, 0, 4);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) w_city, 0, ADDRESS_LEN,
NULL, 0, 0, 5);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) w_state, 0, STATE_LEN,
NULL, 0, 0, 6);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) w_zip, 0, ZIP_LEN, NULL, 0, 0,
7);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) &w_tax, 0,
SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 8);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) &w_ytd, 0,
SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 9);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        time_start = (TimeNow() / MILLI);

        warehouse_rows_loaded = 0;

        for (w_id = (short)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;
    short 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];
    char err_log_path[256];

    // Seed with unique number
    seed(4);

    printf("Loading district table...\n");

    // build index before load
    if ((aptr->build_index == 1) && (aptr->index_order == 1))
        BuildIndex("idxdiscl1");

    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, "district");

    //rc = bcp_init(w_hdbc1, name, NULL, "logs\\district.err", DB_IN);
    strcpy(err_log_path, aptr->log_path);
    strcat(err_log_path, "district.err");

```

```

rc = bcp_init(w_hdbc1, name, NULL, err_log_path, DB_IN);
if (rc != SUCCEEDED)
    HandleErrorDBC(w_hdbc1);

if ((aptr->build_index == 1) && (aptr->index_order == 1))
{
    sprintf(bcp_hint, "tablock, order (d_w_id, d_id),
ROWS_PER_BATCH = %u", (aptr->num_warehouses * 10));
rc = bcp_control(w_hdbc1, BCPHINTS, (void*)
bcp_hint);
    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, SQLINT2, 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 != SUCCEEDED)
    HandleErrorDBC(w_hdbc1);

rc = bcp_bind(w_hdbc1, (BYTE *) &d_next_o_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 11);
if (rc != SUCCEEDED)
    HandleErrorDBC(w_hdbc1);

```

```

d_ytd = 30000.0;

d_next_o_id = orders_per_district+1;

time_start = (TimeNow() / MILLI);

for (w_id = aptr->starting_warehouse; w_id <=
aptr->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))/10000.00;

rc = bcp_sendrow(w_hdbc1);
if (rc != SUCCEEDED)
    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 ((aptr->build_index == 1) && (aptr->index_order == 0))
    BuildIndex("idxdiscl");

return;
}

//=====
//
// Function : Stock
//
//=====

void Stock()
{
    long   s_i_id;
    short  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];
char err_log_path[256];

// Seed with unique number
seed(3);

// if build index before load...
if ((aptr->build_index == 1) && (aptr->index_order == 1))
    BuildIndex("idxstkcl");

sprintf(name, "%s..%s", aptr->database, "stock");

//rc = bcp_init(w_hdbc1, name, NULL, "logs\\stock.err", DB_IN);
strcpy(err_log_path, aptr->log_path);
strcat(err_log_path, "stock.err");
rc = bcp_init(w_hdbc1, name, NULL, err_log_path, DB_IN);
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

if ((aptr->build_index == 1) && (aptr->index_order == 1))
{
    sprintf(bcpint, "tablock, order (s_i_id, s_w_id),
ROWS_PER_BATCH = %u", (aptr->num_warehouses * 100000));
    rc = bcp_control(w_hdbc1, BCPHINTS, (void*)
bcpint);
    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);

rc = bcp_bind(w_hdbc1, (BYTE *) &s_w_id, 0, SQL_VARLEN_DATA,
NULL, 0, SQLINT2, 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 = (short)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;
        short                  w_id;

        short                  d_id;
        DWORD
dwThreadID[MAX_CUSTOMER_THREADS];
        HANDLE
hThread[MAX_CUSTOMER_THREADS];
        char                    name[20];
        RETCODE
rc;
        DBINT
char                                rcint;
        bcphint[128];
        char                                cmd[256];
        int
num_procs;
        char
err_log_path_cust[256];
        char
err_log_path_hist[256];
        // SQLRETURN                                rc_1;
        // SQLSMALLINT                               recnum,
        MsgLen;
        // SQLCHAR
        SqlState[6], Msg[SQL_MAX_MESSAGE_LENGTH];
        // SQLINTEGER                                NativeError;

        // Seed with unique number
        seed(5);

        printf("Loading customer and history tables...\n");

        // if build index before load...
        if ((aptr->build_index == 1) && (aptr->index_order == 1))
        {
                BuildIndex("idxcuscl");
                // check the number of processors on this system
                // if 8 or more processors, then build index on History.
                // if less than 8 processors, do not build the index
                num_procs = atoi(getenv(
"NUMBER_OF_PROCESSORS" ));
                if ( num_procs >= 8 )
                        BuildIndex("idxhiscl");
        }

        // Initialize bulk copy
        sprintf(name, "%s.%s", aptr->database, "customer");

        //rc = bcp_init(c_hdbc1, name, NULL, "logs\customer.err",
DB_IN);
        strcpy(err_log_path_cust,aptr->log_path);
        strcat(err_log_path_cust,"customer.err");
        rc = bcp_init(c_hdbc1, name, NULL, err_log_path_cust, DB_IN);
        if (rc != SUCCEED)
                HandleErrorDBC(c_hdbc1);

        if ((aptr->build_index == 1) && (aptr->index_order == 1))
        {
                sprintf(bcphint, "tablock, order (c_w_id, c_d_id, c_id),
ROWS_PER_BATCH = %u", (aptr->num_warehouses * 30000));
                rc = bcp_control(c_hdbc1, BCPHINTS, (void*) bcphint);

                if (rc != SUCCEED)
                        HandleErrorDBC(c_hdbc1);
        }
}

```

```

sprintf(name, "%s.%s", apr->database, "history");

rc = bcp_init(c_hdbc2, name, NULL, "logs\\history.err", DB_IN);
strcpy(err_log_path_hist, apr->log_path);
strcat(err_log_path_hist, "history.err");
rc = bcp_init(c_hdbc2, name, NULL, err_log_path_hist, DB_IN);
if (rc != SUCCEED)
    HandleErrorDBC(c_hdbc2);

sprintf(bcphint, "tablock");
rc = bcp_control(c_hdbc2, BCPHINTS, (void*) bcphint);
if (rc != SUCCEED)
    HandleErrorDBC(c_hdbc2);

customer_rows_loaded = 0;
history_rows_loaded = 0;

CustomerBufInit();

customer_time_start.time_start = (TimeNow() / MILLI);
history_time_start.time_start = (TimeNow() / MILLI);

for (w_id = (short)apr->starting_warehouse; w_id <=
apr->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 ((apr->build_index == 1) && (apr->index_order == 0))
{
    BuildIndex("idxcuscl");
    // check the number of processors on this system
    // if 8 or more processors, then build index on History.
    // if less than 8 processors, do not build the index
    num_procs = atoi(getenv(
"NUMBER_OF_PROCESSORS" ));
    if (num_procs >= 8)
        BuildIndex("idxhisc1");
}

// build non-clustered index
if (apr->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, "osql -S%s -U%s -P%s -d%s -e -Q"update customer
set c_first = 'C_LOAD = %d' where c_id = 1 and c_w_id = 1 and c_d_id = 1" >
logs\\nurand_load.log",
sprintf(cmd, "osql -S%s -U%s -P%s -d%s -e -Q"update customer set
c_first = 'C_LOAD = %d' where c_id = 1 and c_w_id = 1 and c_d_id = 1" >
%snurand_load.log",

aptr->server,
aptr->user,
aptr->password,
aptr->database,
LOADER_NURAND_C,
aptr->log_path);

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
        // 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;

```

problem.

```

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;
    short  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, SQLINT2, 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;
    short 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, SQLINT2, 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, SQLINT2, 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;
    short w_id;

    short d_id;
    DWORD
dwThreadID[MAX_ORDER_THREADS];
    HANDLE
hThread[MAX_ORDER_THREADS];
    char name[20];
    RETCODE

rc;
    char
bcphint[128];
    char
err_log_path_ord[256];
    char
err_log_path_nord[256];

```

```

    char
err_log_path_ordl[256];

    // seed with unique number
    seed(6);

    printf("Loading orders...\n");

    // if build index before load...
    if ((aptr->build_index == 1) && (aptr->index_order == 1))
    {
        BuildIndex("idxordcl");
        BuildIndex("idxnodcl");
        BuildIndex("idxodlcl");
    }

    // initialize bulk copy
    sprintf(name, "%s..%s", aptr->database, "orders");

    rc = bcp_init(o_hdbc1, name, NULL, "logs\\orders.err", DB_IN);
    strcpy(err_log_path_ord, aptr->log_path);
    strcat(err_log_path_ord, "orders.err");
    rc = bcp_init(o_hdbc1, name, NULL, err_log_path_ord, DB_IN);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc1);

    if ((aptr->build_index == 1) && (aptr->index_order == 1))
    {
        sprintf(bcphint, "tablock, order (o_w_id, o_d_id, o_id),
ROWS_PER_BATCH = %u", (aptr->num_warehouses * 30000));
        rc = bcp_control(o_hdbc1, BCPHINTS, (void*) bcphint);

        if (rc != SUCCEEDED)
            HandleErrorDBC(o_hdbc1);
    }

    sprintf(name, "%s..%s", aptr->database, "new_order");

    rc = bcp_init(o_hdbc2, name, NULL, "logs\\neword.err", DB_IN);
    strcpy(err_log_path_nord, aptr->log_path);
    strcat(err_log_path_nord, "neword.err");
    rc = bcp_init(o_hdbc2, name, NULL, err_log_path_nord, DB_IN);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc2);

    if ((aptr->build_index == 1) && (aptr->index_order == 1))
    {
        sprintf(bcphint, "tablock, order (no_w_id, no_d_id,
no_o_id), ROWS_PER_BATCH = %u", (aptr->num_warehouses * 9000));
        rc = bcp_control(o_hdbc2, BCPHINTS, (void*) bcphint);

        if (rc != SUCCEEDED)
            HandleErrorDBC(o_hdbc2);
    }

    sprintf(name, "%s..%s", aptr->database, "order_line");

    rc = bcp_init(o_hdbc3, name, NULL, "logs\\ordline.err", DB_IN);
    strcpy(err_log_path_ordl, aptr->log_path);
    strcat(err_log_path_ordl, "ordline.err");
    rc = bcp_init(o_hdbc3, name, NULL, err_log_path_ordl, DB_IN);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc3);

    if ((aptr->build_index == 1) && (aptr->index_order == 1))
    {

```

```

        sprintf(bcphint, "tablock, order (ol_w_id, ol_d_id,
ol_o_id, ol_number), ROWS_PER_BATCH = %u", (aptr->num_warehouses *
300000));
        rc = bcp_control(o_hdbc3, BCPHINTS, (void*) bcphint);
        if (rc != SUCCEEDED)
            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 = (short)aptr->starting_warehouse; w_id <=
aptr->num_warehouses; w_id++)
    {
        for (d_id = 1; d_id <= DISTRICT_PER_WAREHOUSE;
d_id++)
        {
            OrdersBufLoad(d_id, w_id);
            // start parallel loading threads here...

            // start Orders table thread
            printf("...Loading Order Table for: d_id = %d,
w_id = %d\n", d_id, w_id);

            hThread[0] = CreateThread(NULL,
0,
(LPTHREAD_START_ROUTINE) LoadOrdersTable,
&orders_time_start,
0,
&dwThreadID[0]);

            if (hThread[0] == NULL)
            {
                printf("Error, failed in creating
creating thread = 0.\n");
                exit(-1);
            }
            // start NewOrder table thread
            printf("...Loading New-Order Table for: d_id
= %d, w_id = %d\n", d_id, w_id);

            hThread[1] = CreateThread(NULL,
0,
(LPTHREAD_START_ROUTINE) LoadNewOrderTable,
&new_order_time_start,
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;
    short 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 != SUCCEEDED)
        HandleErrorDBC(o_hdbc1);

    rc = bcp_bind(o_hdbc1, (BYTE *) &o_d_id, 0, SQL_VARLEN_DATA,
    NULL, 0, SQLINT2, 2);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc1);

    rc = bcp_bind(o_hdbc1, (BYTE *) &o_w_id, 0, SQL_VARLEN_DATA,
    NULL, 0, SQLINT2, 3);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc1);

    rc = bcp_bind(o_hdbc1, (BYTE *) &o_c_id, 0, SQL_VARLEN_DATA,
    NULL, 0, SQLINT4, 4);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc1);

    rc = bcp_bind(o_hdbc1, (BYTE *) &o_entry_d, 0,
    O_ENTRY_D_LEN, NULL, 0, SQLCHARACTER, 5);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc1);

    rc = bcp_bind(o_hdbc1, (BYTE *) &o_carrier_id, 0,
    SQL_VARLEN_DATA, NULL, 0, SQLINT2, 6);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc1);

    rc = bcp_bind(o_hdbc1, (BYTE *) &o_ol_cnt, 0, SQL_VARLEN_DATA,
    NULL, 0, SQLINT2, 7);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc1);

    rc = bcp_bind(o_hdbc1, (BYTE *) &o_all_local, 0, SQL_VARLEN_DATA,
    NULL, 0, SQLINT2, 8);
    if (rc != SUCCEEDED)
        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 != SUCCEEDED)
    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) && (apr->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;
    short 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 != SUCCEEDED)
        HandleErrorDBC(o_hdbc2);

    rc = bcp_bind(o_hdbc2, (BYTE *) &o_d_id, 0, SQL_VARLEN_DATA,
    NULL, 0, SQLINT2, 2);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc2);
}

```

```

rc = bcp_bind(o_hdbc2, (BYTE *) &o_w_id, 0, SQL_VARLEN_DATA,
NULL, 0, SQLINT2, 3);
if (rc != SUCCEEDED)
    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 != SUCCEEDED)
        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 == aptr->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 ((aptr->build_index == 1) && (aptr->index_order ==
0))
        BuildIndex("idxnodcl");
}
}

//=====
//
// Function : LoadOrderLineTable
//
//=====

void LoadOrderLineTable(LOADER_TIME_STRUCT *order_line_time_start)
{
    int    ij;
    long   o_id;
    short  o_d_id;
    short  o_w_id;
    long   ol;
    long   ol_i_id;
    short  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 != SUCCEEDED)
    HandleErrorDBC(o_hdbc3);

rc = bcp_bind(o_hdbc3, (BYTE *) &o_d_id, 0, SQL_VARLEN_DATA,
NULL, 0, SQLINT2, 2);
if (rc != SUCCEEDED)
    HandleErrorDBC(o_hdbc3);

rc = bcp_bind(o_hdbc3, (BYTE *) &o_w_id, 0, SQL_VARLEN_DATA,
NULL, 0, SQLINT2, 3);
if (rc != SUCCEEDED)
    HandleErrorDBC(o_hdbc3);

rc = bcp_bind(o_hdbc3, (BYTE *) &ol, 0, SQL_VARLEN_DATA, NULL,
0, SQLINT4, 4);
if (rc != SUCCEEDED)
    HandleErrorDBC(o_hdbc3);

rc = bcp_bind(o_hdbc3, (BYTE *) &ol_i_id, 0, SQL_VARLEN_DATA,
NULL, 0, SQLINT4, 5);
if (rc != SUCCEEDED)
    HandleErrorDBC(o_hdbc3);

rc = bcp_bind(o_hdbc3, (BYTE *) &ol_supply_w_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 6);
if (rc != SUCCEEDED)
    HandleErrorDBC(o_hdbc3);

rc = bcp_bind(o_hdbc3, (BYTE *) &ol_delivery_d, 0,
OL_DELIVERY_D_LEN, NULL, 0, SQLCHARACTER, 7);
if (rc != SUCCEEDED)
    HandleErrorDBC(o_hdbc3);

rc = bcp_bind(o_hdbc3, (BYTE *) &ol_quantity, 0, SQL_VARLEN_DATA,
NULL, 0, SQLINT2, 8);
if (rc != SUCCEEDED)
    HandleErrorDBC(o_hdbc3);

rc = bcp_bind(o_hdbc3, (BYTE *) &ol_amount, 0, SQL_VARLEN_DATA,
NULL, 0, SQLFLT8, 9);
if (rc != SUCCEEDED)
    HandleErrorDBC(o_hdbc3);

rc = bcp_bind(o_hdbc3, (BYTE *) ol_dist_info, 0, DIST_INFO_LEN,
NULL, 0, 0, 10);
if (rc != SUCCEEDED)
    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 == aptr->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 ((aptr->build_index == 1) && (aptr->index_order ==
0))
        BuildIndex("idxodlcl");

}

}

//=====
//
// 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,
                    int 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 );

```

<pre> SQLAllocHandle(SQL_HANDLE_DBC, henv, &amp;i_hdbc1); SQLAllocHandle(SQL_HANDLE_DBC, henv, &amp;w_hdbc1); SQLAllocHandle(SQL_HANDLE_DBC, henv, &amp;c_hdbc1); SQLAllocHandle(SQL_HANDLE_DBC, henv, &amp;c_hdbc2); SQLAllocHandle(SQL_HANDLE_DBC, henv, &amp;o_hdbc1); SQLAllocHandle(SQL_HANDLE_DBC, henv, &amp;o_hdbc2); SQLAllocHandle(SQL_HANDLE_DBC, henv, &amp;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-&gt;server, aptr-&gt;password, aptr-&gt;database );  rc = SQLSetConnectOption (i_hdbc1, SQL_PACKET_SIZE, aptr-&gt;pack_size); if (rc != SUCCEED)     HandleErrorDBC(i_hdbc1);  rc = SQLDriverConnect ( i_hdbc1, NULL, (SQLCHAR*)&amp;szDriverString[0] , (SQLCHAR*)&amp;szDriverStringOut[0], sizeof(szDriverStringOut), &amp;cbDriverStringOut, SQL_DRIVER_NOPROMPT ); if (rc != SUCCEED)     HandleErrorDBC(i_hdbc1);  // Connection 2  sprintf( szDriverString, "DRIVER={SQL Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s", aptr-&gt;server, aptr-&gt;password, </pre>	<pre> aptr-&gt;database );  rc = SQLSetConnectOption (w_hdbc1, SQL_PACKET_SIZE, aptr-&gt;pack_size); if (rc != SUCCEED)     HandleErrorDBC(w_hdbc1);  rc = SQLDriverConnect ( w_hdbc1, NULL, (SQLCHAR*)&amp;szDriverString[0] , SQL_NTS, (SQLCHAR*)&amp;szDriverStringOut[0], sizeof(szDriverStringOut), &amp;cbDriverStringOut, SQL_DRIVER_NOPROMPT ); if (rc != SUCCEED)     HandleErrorDBC(w_hdbc1);  // Connection 3  sprintf( szDriverString, "DRIVER={SQL Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s", aptr-&gt;server, aptr-&gt;password, aptr-&gt;database );  rc = SQLSetConnectOption (c_hdbc1, SQL_PACKET_SIZE, aptr-&gt;pack_size); if (rc != SUCCEED)     HandleErrorDBC(c_hdbc1);  rc = SQLDriverConnect ( c_hdbc1, NULL, (SQLCHAR*)&amp;szDriverString[0] , SQL_NTS, (SQLCHAR*)&amp;szDriverStringOut[0], sizeof(szDriverStringOut), &amp;cbDriverStringOut, SQL_DRIVER_NOPROMPT ); if (rc != SUCCEED)     HandleErrorDBC(c_hdbc1);  // Connection 4  sprintf( szDriverString, "DRIVER={SQL Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s", aptr-&gt;server, aptr-&gt;password, aptr-&gt;database ); </pre>
--	---

```

aptr->password,
aptr->database );

    rc = SQLSetConnectOption ( c_hdbc2, SQL_PACKET_SIZE,
aptr->pack_size);
    if (rc != SUCCEED)
        HandleErrorDBC(c_hdbc2);

    rc = SQLDriverConnect ( c_hdbc2,
NULL,
(SQLCHAR*)&szDriverString[0] ,
SQL_NTS,
(SQLCHAR*)&szDriverStringOut[0],
sizeof(szDriverStringOut),
&cbDriverStringOut,
SQL_DRIVER_NOPROMPT );
    if (rc != SUCCEED)
        HandleErrorDBC(c_hdbc2);

    // Connection 5

    sprintf( szDriverString , "DRIVER={SQL
Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s" ,
aptr->server,
aptr->password,
aptr->database );

    rc = SQLSetConnectOption ( o_hdbc1, SQL_PACKET_SIZE,
aptr->pack_size);
    if (rc != SUCCEED)
        HandleErrorDBC(o_hdbc1);

    rc = SQLDriverConnect ( o_hdbc1,
NULL,
(SQLCHAR*)&szDriverString[0] ,
SQL_NTS,
(SQLCHAR*)&szDriverStringOut[0],
sizeof(szDriverStringOut),
&cbDriverStringOut,
SQL_DRIVER_NOPROMPT );
    if (rc != SUCCEED)
        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 != SUCCEED)
        HandleErrorDBC(o_hdbc2);

    rc = SQLDriverConnect ( o_hdbc2,
NULL,
(SQLCHAR*)&szDriverString[0] ,
SQL_NTS,
(SQLCHAR*)&szDriverStringOut[0],
sizeof(szDriverStringOut),
&cbDriverStringOut,
SQL_DRIVER_NOPROMPT );
    if (rc != SUCCEED)
        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 != SUCCEED)
        HandleErrorDBC(o_hdbc3);

    rc = SQLDriverConnect ( o_hdbc3,
NULL,
(SQLCHAR*)&szDriverString[0] ,
SQL_NTS,
(SQLCHAR*)&szDriverStringOut[0],
sizeof(szDriverStringOut),
&cbDriverStringOut,
SQL_DRIVER_NOPROMPT );
    if (rc != SUCCEED)
        HandleErrorDBC(o_hdbc3);
}

```

```

//=====
//
// Function name: BuildIndex
//
//=====
void BuildIndex(char *index_script)
{
    char    cmd[256];

    printf("Starting index creation: %s\n",index_script);

    sprintf(cmd, "osql -S%s -U%s -P%s -e -i%s\\%s.sql > %s%s.log",
            apr->server,
            apr->user,
            apr->password,
            apr->index_script_path,
            index_script,
            apr->log_path,
            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];
    char    err_log_path[256];
    FILE    *fp1;

    i = 1;
    while (( rc2 = SQLGetDiagRec(SQL_HANDLE_DBC , hdbc1 , i,
    SqlState , &NativeError,
                                Msg, sizeof(Msg) , &MsgLen ))
    != SQL_NO_DATA )
    {
        printf( szLastError , "%s" , Msg );

        _strtime(timebuf);
        _strdate(datebuf);

        printf( "[%s : %s] %s\n" , datebuf, timebuf, szLastError);

        strcpy(err_log_path,apr->log_path);
        strcat(err_log_path,"tpccldr.err");
        fp1 = fopen(err_log_path,"w");
        //fp1 = fopen("logs\\tpccldr.err","w");
        if (fp1 == NULL)
            printf("ERROR: Unable to open errorlog
file.\n");
        else
        {
            fprintf(fp1, "[%s : %s] %s\n" , datebuf,
timebuf, szLastError);
            fclose(fp1);
        }
        i++;
    }
}

void FormatDate ( char* szTimeCOutput )
{
    struct tm when;
    time_t now;

    time( &now );
    when = *localtime( &now );

    mktime( &when );

    // odbc datetime format
    strftime( szTimeCOutput , 30 , "%Y-%m-%d %H:%M:%S.000",
    &when );
}

```

```

        return;
    }
}
//=====
//
// Function : 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 LOGS\ directory for database creation
errors.\n");

        // cleanup database connections and handles
        SQLFreeHandle(SQL_HANDLE_STMT, v_hstmt);
        SQLDisconnect(v_hdbc);
        SQLFreeHandle(SQL_HANDLE_DBC, v_hdbc);

        // since there is not a database, exit back to SETUP.CMD
        exit(1);
    }

    if ( SQLAllocHandle(SQL_HANDLE_STMT, v_hdbc , &v_hstmt)
!= SQL_SUCCESS )
        HandleErrorDBC(v_hdbc);

    if ( SQLBindCol(v_hstmt, 1, SQL_C_ULONG, &TabCount, 0,
&TabCountInd) != SQL_SUCCESS )
        HandleErrorSTMT(v_hstmt);

    // count the number of user tables from sysobjects
    rc = SQLExecDirect(v_hstmt, "select count(*) from sysobjects
where xtype = '\U'", SQL_NTS);
    if ((rc != SQL_SUCCESS) && (rc !=
SQL_SUCCESS_WITH_INFO))
        HandleErrorSTMT(v_hstmt);

    if ( SQLFetch(v_hstmt) != SQL_SUCCESS )
        HandleErrorSTMT(v_hstmt);

    // if the number of tables is less than 9, select all the user tables in
TPCC
    if (TabCount != 9)
    {
        SQLFreeHandle(SQL_HANDLE_STMT, v_hstmt);

        SQLAllocHandle(SQL_HANDLE_STMT, v_hdbc ,
&v_hstmt);

        if ( SQLBindCol(v_hstmt, 1, SQL_C_CHAR,
&TabName, sizeof(TabName), &TabNameInd) != SQL_SUCCESS )
            HandleErrorSTMT(v_hstmt);

        // select the list of user tables into a result set
        rc = SQLExecDirect(v_hstmt, "select * from sysobjects
where xtype = '\U'", SQL_NTS);
        if ((rc != SQL_SUCCESS) && (rc !=
SQL_SUCCESS_WITH_INFO))
            HandleErrorSTMT(v_hstmt);

        // go through the result set and set the bitmap for each
found table
        // set the bitmap to '1' if the table name is found

        while ((rc = SQLFetch(v_hstmt)) != SQL_NO_DATA)
        {

```

<pre> 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 &lt;= 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') </pre>	<pre> {     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 LOGS\ directory for database\n");     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); } } </pre>
---	--



```

// cleanup database connections and handles
SQLFreeHandle(SQL_HANDLE_STMT, v_hstmt);
SQLDisconnect(v_hdbc);
SQLFreeHandle(SQL_HANDLE_DBC, v_hdbc);

return;
}

```

## tpcldr.mak

```

# Microsoft Developer Studio Generated NMAKE File, Format Version 4.10
# ** DO NOT EDIT **

```

```

# TARGETTYPE "Win32 (x86) Console Application" 0x0103

```

```

!IF "$(CFG)" == ""
CFG=tpcldr - Win32 Debug
!MESSAGE No configuration specified. Defaulting to tpcldr - Win32 Debug.
!ENDIF

```

```

!IF "$(CFG)" != "tpcldr - Win32 Release" && "$(CFG)" !=
"tpcldr - 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 "tpcldr.mak" CFG="tpcldr - Win32 Debug"
!MESSAGE
!MESSAGE Possible choices for configuration are:
!MESSAGE
!MESSAGE "tpcldr - Win32 Release" (based on "Win32 (x86) Console
Application")
!MESSAGE "tpcldr - 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 "tpcldr - Win32 Debug"
RSC=rc.exe
CPP=cl.exe

```

```

!IF "$(CFG)" == "tpcldr - 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)\tpcldr.exe"

```

```

CLEAN :
-@erase "$(INTDIR)\getargs.obj"
-@erase "$(INTDIR)\random.obj"
-@erase "$(INTDIR)\strings.obj"
-@erase "$(INTDIR)\time.obj"
-@erase "$(INTDIR)\tpcldr.obj"
-@erase "$(OUTDIR)\tpcldr.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\ddlib\include" /D
"NDEBUG" /D "WIN32" /D "_CONSOLE" /D "DBNTWIN32" /c
# SUBTRACT CPP /YX
CPP_PROJ=/nologo /MT /W3 /GX /O2 /I "c:\mssql\ddlib\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)\tpcldr.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 odbcc32.lib
odbc32.lib /nologo /subsystem:console /machine:I386
# ADD LINK32 c:\mssql\ddlib\lib\ntwdblib.lib kernel32.lib user32.lib gdi32.lib
winspool.lib comdlg32.lib advapi32.lib shell32.lib ole32.lib oleaut32.lib
uuid.lib odbcc32.lib odbc32.lib /nologo /subsystem:console /pdb:none
/machine:I386
LINK32_FLAGS=c:\mssql\ddlib\lib\ntwdblib.lib kernel32.lib user32.lib
gdi32.lib\
winspool.lib comdlg32.lib advapi32.lib shell32.lib ole32.lib oleaut32.lib\
uuid.lib odbcc32.lib odbccp32.lib /nologo /subsystem:console /pdb:none\
/machine:I386 /out:"$(OUTDIR)\tpcldr.exe"
LINK32_OBJS= \

```

```

"$$(INTDIR)\getargs.obj" \
"$$(INTDIR)\random.obj" \
"$$(INTDIR)\strings.obj" \
"$$(INTDIR)\time.obj" \
"$$(INTDIR)\tpcldr.obj"

```

```

"$$(OUTDIR)\tpcldr.exe" : "$$(OUTDIR)" $(DEF_FILE) $(LINK32_OBJS)
$(LINK32) @<<
$(LINK32_FLAGS) $(LINK32_OBJS)
<<

```

```

!ELSEIF "$(CFG)" == "tpcldr - 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 "$(INTDIR)\tpccldr.obj"
    -@erase "$(INTDIR)\vc40.idb"
    -@erase "$(INTDIR)\vc40.pdb"
    -@erase "$(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 odbccp32.lib
odbc32.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 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 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_TPCCCL=\
    ".\src\tpcc.h"
    "\mssql\dblib\include\sqldb.h"
    "\mssql\dblib\include\sqlfront.h"

"${INTDIR}\tpccldr.obj" : $(SOURCE) $(DEP_CPP_TPCCCL) "${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
#####
#####

```

## VerifyTpccLoad.sql

```

-- File:  VERIFYTPCCLOAD.SQL
--       Microsoft TPC-C Benchmark Kit Ver. 4.41
--       Copyright Microsoft, 2001
-- Purpose: Performs series of TPCC database checks to verify
--          that database load completed correctly

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

use tpcc
go

-- *****

```

```

--
-- Check rows per table from SYSINDEXES
--
-- *****

print  'WAREHOUSE TABLE'

select  rows
from    sysindexes
where   id      = object_id("warehouse")
go

print  'DISTRICT TABLE = (10 * No of warehouses)'

select  rows
from    sysindexes
where   id      =object_id("district")
go

print  'ITEM TABLE = 100,000'

select  rows
from    sysindexes
where   id      =object_id("item")
go

print  'CUSTOMER TABLE = (30,000 * No of warehouses)'

select  rows
from    sysindexes
where   id      =object_id("customer")
go

print 'ORDERS TABLE = (30,000 * No of warehouses)'

select  rows
from    sysindexes
where   id      =object_id("orders")
go

print  'HISTORY TABLE = (30,000 * No of warehouses)'

select  rows
from    sysindexes
where   id      =object_id("history")
go

print  'STOCK TABLE = (100,000 * No of warehouses)'

select  rows
from    sysindexes
where   id      =object_id("stock")
go

print  'ORDER_LINE TABLE = (300,000 * No of warehouses + some
change)'

select  count_big(*)
from    order_line
go

print  'NEW_ORDER TABLE = (9000 * No of warehouses)'

select  rows
from    sysindexes
where   id      =object_id("new_order")
go

```

```

--      *****
--
--      Check indices
--
--      *****

print '*****Index Check*****'

use tpcc
go

sp_helpindex      customer
go

sp_helpindex      stock
go

sp_helpindex      district
go

sp_helpindex      item
go

sp_helpindex      new_order
go

sp_helpindex      orders
go

sp_helpindex      order_line
go

sp_helpindex      warehouse
go

```

### ***version.sql***

```

-- File:  VERSION.SQL
--      Microsoft TPC-C Benchmark Kit Ver. 4.41
--      Copyright Microsoft, 2001
-- Purpose: Extracts current version of SQL Server

use master
go

SELECT  CONVERT(char(20), SERVERPROPERTY('ProductVersion'))
go

SELECT  CONVERT(char(20), SERVERPROPERTY('ProductLevel'))
go

SELECT  CONVERT(char(30), getdate(),9)
go

```

# Appendix C: Tunable Parameters

## Microsoft SQL Server 2000 Configuration Parameters

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

```
-- File:  VERSION.SQL
--      Microsoft TPC-C Benchmark Kit Ver. 4.22
--      Copyright Microsoft, 2001
-- Purpose: Returns SQL Server version string
```

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

-----  
Jul 15 2003 5:47:23:880PM

(1 row affected)

```
1> 2> 3>
select @@version
```

```
-----
-----
-----
Microsoft SQL Server 2000 - 8.00.782 (Intel X86)
Jun 16 2003 16:49:53
Cop
yright (c) 1988-2003 Microsoft Corporation
Enterprise Edition on Windo
ws NT 5.2 (Build 3790: )
```

(1 row affected)

```
1> 2>
1> 2> 3> 4> 5> 6> 7> 8> 9> 10>
-- File:  CONFIG.SQL
--      Microsoft TPC-C Benchmark Kit Ver. 4.22
--      Copyright Microsoft, 2001
-- Purpose: Collects SQL Server configuration parameters
```

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

-----  
Jul 15 2003 5:47:24:783PM

(1 row affected)

```
1> 2> 3> DBCC execution completed. If DBCC printed error messages, contact
your system administrator.
Configuration option 'show advanced options' changed from 1 to 1. Run the
RECONFIGURE statement to install.
```

```
sp_configure "show advanced",1
```

1> 2> reconfigure with override

1> 2> sp\_configure

name	minimum	maximum
config_value run_value		
-----		
affinity mask	-2147483648	2147483647
-2 -2		
allow updates	0	1
0		
awe enabled	0	1
1		
c2 audit mode	0	1
0		
cost threshold for parallelism	0	32767
5 5		
Cross DB Ownership Chaining	0	1
0 0		
cursor threshold	-1	2147483647
-1 -1		
default full-text language	0	2147483647
1033 1033		
default language	0	9999
0		
fill factor (%)	0	100
0		
index create memory (KB)	704	2147483647
0 0		
lightweight pooling	0	1
1		
locks	5000	2147483647
0		
max degree of parallelism	0	32
1		
max server memory (MB)	4	2147483647
62600 62600		
max text repl size (B)	0	2147483647
65536 65536		
max worker threads	32	32767
700 700		
media retention	0	365
0		
min memory per query (KB)	512	2147483647
1024 1024		
min server memory (MB)	0	2147483647
0 0		
nested triggers	0	1
1		
network packet size (B)	512	65536
4096 4096		
open objects	0	2147483647
0		
priority boost	0	1
1		
query governor cost limit	0	2147483647
0 0		
query wait (s)	-1	2147483647
-1		
recovery interval (min)	0	32767
112 112		
remote access	0	1
1		
remote login timeout (s)	0	2147483647
0 0		
remote proc trans	0	1
0		
remote query timeout (s)	0	2147483647
0 0		

```

scan for startup procs          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>

```

## Microsoft Windows Server 2003 Datacenter Edition

### Changes to the SUT

Changes made to the default installation of .NET DataCenter Edition on the SUT  
Add Windows Server 2003 Hotfix Package KB817812

All services were left in their default setup.

Control Panel - System - Advanced - Performance Options - Visual Effects - Adjust for best performance

Control Panel - System - Advanced - Performance Options - Advanced - Memory usage - Adjust for best performance of: Programs

c:\boot.ini added /PAE /MAXMEMORY=65535 /hal=halx440.dll

gpedit.msc - Computer Configuration - Windows Settings - Security Settings - Local Policies - User Rights Assignments - policy 'Lock pages in memory' added group 'Administrators'

Enabled VIA protocol for SQL Server

Microsoft SQL Server Startup Parameters

C:\Program Files\Microsoft SQL Server\MSSQL\Binn\sqlservr -c -x -t3502

where:

- c Start SQL Server independent of the Service Control Manager
- x Disable the keeping of CPU time and cache hit ratio statistic
- t3502 writes a message to the SQL Server Errorlog showing the beginning and ending time of each checkpoint
- T3428 flag allows use of half the memory available on the system during the recovery process to keep a hash table that enables recovery to proceed smoothly.

In a command prompt run the following prior to starting SQL Server:  
set logaffinity=0x80000000  
This ties SQL log affinity to the first processor in the system.

regedit  
Added DWORD value to  
HKLM\SYSTEM\CurrentControlSet\Services\tcpip\Parameters  
"MaxUserPort" 0x9c40

```

Key Name:
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Session
Manager\Memory Management
Class Name: <NO CLASS>
Last Write Time: 1/23/2003 - 2:44 PM
Value 0
Name: ClearPageFileAtShutdown
Type: REG_DWORD
Data: 0
Value 1
Name: DisablePagingExecutive
Type: REG_DWORD
Data: 0
Value 2
Name: LargeSystemCache
Type: REG_DWORD
Data: 0
Value 3
Name: NonPagedPoolQuota
Type: REG_DWORD
Data: 0
Value 4
Name: NonPagedPoolSize
Type: REG_DWORD
Data: 0
Value 5
Name: PagedPoolQuota
Type: REG_DWORD
Data: 0
Value 6
Name: PagedPoolSize
Type: REG_DWORD
Data: 0
Value 7
Name: SecondLevelDataCache
Type: REG_DWORD
Data: 0
Value 8
Name: SystemPages
Type: REG_DWORD
Data: 0x33000
Value 9
Name: PagingFiles
Type: REG_MULTI_SZ
Data: c:\pagefile.sys 2046 4092
Value 10
Name: PhysicalAddressExtension
Type: REG_DWORD
Data: 0x1
Value 11
Name: WriteWatch
Type: REG_DWORD
Data: 0x1
Value 12
Name: SessionViewSize
Type: REG_DWORD

```

Data: 0x30  
Value 13  
Name: SessionPoolSize  
Type: REG\_DWORD  
Data: 0x4  
Value 14  
Name: DontVerifyRandomDrivers  
Type: REG\_DWORD  
Data: 0x1

Key Name:  
HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Control\Session  
Manager\I/O System  
Class Name: <NO CLASS>  
Last Write Time: 1/23/2003 - 1:41 PM  
Value 0  
Name: CountOperations  
Type: REG\_DWORD  
Data: 0

Key Name:  
HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\qlvika  
Class Name: <NO CLASS>  
Last Write Time: 7/15/2003 - 4:12 PM  
Value 0  
Name: ErrorControl  
Type: REG\_DWORD  
Data: 0x1

Value 1  
Name: start  
Type: REG\_DWORD  
Data: 0x2

Value 2  
Name: type  
Type: REG\_DWORD  
Data: 0x1

Value 3  
Name: Tag  
Type: REG\_DWORD  
Data: 0x1

Value 4  
Name: group  
Type: REG\_SZ  
Data: MVIA

Value 5  
Name: ImagePath  
Type: REG\_EXPAND\_SZ  
Data: system32\DRIVERS\qlvika.sys

Key Name:  
HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\qlvika\Ada  
pters  
Class Name: <NO CLASS>  
Last Write Time: 5/19/2003 - 5:13 PM

Key Name:  
HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\qlvika\Ada  
pters\210000E08B0723B0  
Class Name: <NO CLASS>  
Last Write Time: 4/16/2003 - 2:13 PM

Value 0  
Name: IPAddress  
Type: REG\_MULTI\_SZ  
Data: 192.168.191.1

Value 1  
Name: VIDeviceNumber  
Type: REG\_DWORD  
Data: 0

Key Name:  
HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\qlvika\Ada  
pters\210000E08B07E5AF  
Class Name: <NO CLASS>  
Last Write Time: 4/16/2003 - 2:13 PM

Value 0  
Name: IPAddress  
Type: REG\_MULTI\_SZ  
Data: 192.168.195.1

Value 1  
Name: VIDeviceNumber  
Type: REG\_DWORD  
Data: 0x1

Key Name:  
HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\qlvika\Ada  
pters\210000E08b09315a  
Class Name: <NO CLASS>  
Last Write Time: 4/16/2003 - 2:13 PM

Value 0  
Name: IPAddress  
Type: REG\_MULTI\_SZ  
Data: 192.168.193.1

Value 1  
Name: VIDeviceNumber  
Type: REG\_DWORD  
Data: 0x2

Key Name:  
HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\qlvika\Ada  
pters\210000e08b09335a  
Class Name: <NO CLASS>  
Last Write Time: 4/16/2003 - 2:13 PM

Value 0  
Name: IPAddress  
Type: REG\_MULTI\_SZ  
Data: 192.168.197.1

Value 1  
Name: VIDeviceNumber  
Type: REG\_DWORD  
Data: 0x3

Key Name:  
HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\qlvika\Ada  
pters\210000E08B09A05A  
Class Name: Key  
Last Write Time: 5/19/2003 - 5:17 PM

Value 0  
Name: IPAddress  
Type: REG\_MULTI\_SZ  
Data: 192.168.199.1

Value 1  
Name: VIDeviceNumber  
Type: REG\_DWORD  
Data: 0x6

Key Name:  
HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\qlvika\Adapters\210000E08B0BCEFF0  
Class Name: Key  
Last Write Time: 5/19/2003 - 3:03 PM

Value 0  
Name: IPAddress  
Type: REG\_MULTI\_SZ  
Data: 192.168.192.1

Value 1  
Name: VIDeviceNumber  
Type: REG\_DWORD  
Data: 0x4

Key Name:  
HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\qlvika\Adapters\210000E08B0BCFF0  
Class Name: Key  
Last Write Time: 5/19/2003 - 3:02 PM

Value 0  
Name: IPAddress  
Type: REG\_MULTI\_SZ  
Data: 192.168.194.1

Value 1  
Name: VIDeviceNumber  
Type: REG\_DWORD  
Data: 0x5

Key Name:  
HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\qlvika\Adapters\210000E08B0BD1F0  
Class Name: Key  
Last Write Time: 5/19/2003 - 3:02 PM

Value 0  
Name: IPAddress  
Type: REG\_MULTI\_SZ  
Data: 192.168.198.1

Value 1  
Name: VIDeviceNumber  
Type: REG\_DWORD  
Data: 0x7

Key Name:  
HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\qlvika\Parameters  
Class Name: <NO CLASS>  
Last Write Time: 4/16/2003 - 3:19 PM

Value 0  
Name: MaxRegisterMBytes  
Type: REG\_DWORD  
Data: 0x200

Value 1  
Name: MaxRegisterRdmaMBytes  
Type: REG\_DWORD

Data: 0x200

Value 2  
Name: MaxCQEntries  
Type: REG\_DWORD  
Data: 0x2000

Value 3  
Name: MaxRegisterRegions  
Type: REG\_DWORD  
Data: 0x1000

Value 4  
Name: MaxVIs  
Type: REG\_DWORD  
Data: 0x400

Value 5  
Name: MaxCQs  
Type: REG\_DWORD  
Data: 0x400

Value 6  
Name: MaxTransferSize  
Type: REG\_DWORD  
Data: 0x10000

Value 7  
Name: MaxPTags  
Type: REG\_DWORD  
Data: 0x800

Value 8  
Name: IuBuffers  
Type: REG\_DWORD  
Data: 0x100

Value 9  
Name: SendDescQuota  
Type: REG\_DWORD  
Data: 0x8

Value 10  
Name: RecvDescQuota  
Type: REG\_DWORD  
Data: 0x8

Value 11  
Name: SupportPrototypeCards  
Type: REG\_DWORD  
Data: 0x1

Key Name:  
HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\qlvika\Security  
Class Name: <NO CLASS>  
Last Write Time: 4/16/2003 - 2:09 PM

Value 0  
Name: Security  
Type: REG\_BINARY  
Data:  
00000000 01 00 14 80 90 00 00 00 - 9c 00 00 00 14 00 00 00 .....  
00000010 30 00 00 00 02 00 1c 00 - 01 00 00 00 02 80 14 00 0.....  
00000020 ff 01 0f 00 01 01 00 00 - 00 00 00 01 00 00 00 00 ý.....  
00000030 02 00 60 00 04 00 00 00 - 00 00 14 00 fd 01 02 00 ..`.....ý...  
00000040 01 01 00 00 00 00 00 05 - 12 00 00 00 00 00 18 00 .....  
00000050 ff 01 0f 00 01 02 00 00 - 00 00 00 05 20 00 00 00 ý.....





Memory Address 0xE9000000-0xE90FFFFF	LSI Logic PCI-X Ultra320 SCSI Host Adapter	0x00000700-0x0000070F	VIA Bus Master IDE Controller	OK
Memory Address 0xA0000-0xA7FFF	PCI bus	0x000001F0-0x000001F7	Primary IDE Channel	OK
Memory Address 0xA0000-0xA7FFF	RAGE XL PCI	0x000003F6-0x000003F6	Primary IDE Channel	OK
Memory Address 0xF7600000-0xF76FFFFF	PCI bus	0x00000170-0x00000177	Secondary IDE Channel	OK
Memory Address 0xF7600000-0xF76FFFFF	LSI Logic PCI-X Ultra320 SCSI Host Adapter	0x00000376-0x00000376	Secondary IDE Channel	OK
		0x00002000-0x000027FF	PCI bus	OK
		0x00002000-0x000027FF	LSI Logic PCI-X Ultra320 SCSI Host	
		0x00002100-0x000021FF	LSI Logic PCI-X Ultra320 SCSI Host	
I/O Port 0x0000C600-0x0000C7FF	PCI bus	Adapter OK		
I/O Port 0x0000C600-0x0000C7FF	QLogic QLA23xx PCI Fibre Channel Adapter	0x00004000-0x00005FFF	PCI bus	OK
		0x00005000-0x000050FF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
Memory Address 0xE8C00000-0xE8DFFFFF	PCI bus	0x00003000-0x00003FFF	PCI bus	OK
Memory Address 0xE8C00000-0xE8DFFFFF	Other PCI Bridge Device	0x00002800-0x000029FF	PCI bus	OK
		0x00002800-0x000029FF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
[DMA]		0x00006000-0x00007FFF	PCI bus	OK
Resource Device Status		0x00007000-0x000070FF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
Channel 2 Standard floppy disk controller	OK	0x0000C000-0x0000C1FF	PCI bus	OK
Channel 4 Direct memory access controller	OK	0x0000C100-0x0000C17F	Other PCI Bridge Device	OK
		0x0000C200-0x0000C3FF	PCI bus	OK
[Forced Hardware]		0x0000C200-0x0000C3FF	LSI Logic PCI-X Ultra320 SCSI Host	
Device PNP Device ID		Adapter OK		
		0x0000C300-0x0000C3FF	LSI Logic PCI-X Ultra320 SCSI Host	
		Adapter OK		
		0x0000C400-0x0000C5FF	PCI bus	OK
		0x0000C400-0x0000C5FF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
		0x0000C500-0x0000C5FF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
		0x0000C600-0x0000C7FF	PCI bus	OK
		0x0000C600-0x0000C7FF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
		0x0000C800-0x0000C9FF	PCI bus	OK
		0x0000CC00-0x0000CFFF	PCI bus	OK
		0x0000CE00-0x0000CEFF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
		0x00008000-0x000083FF	PCI bus	OK
		0x00008400-0x000087FF	PCI bus	OK
		0x00008600-0x000086FF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
		0x0000A000-0x0000BFFF	PCI bus	OK
		0x0000B000-0x0000B0FF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
		0x0000E000-0x0000FFFF	PCI bus	OK
		0x0000E000-0x0000FFFF	Intel 21154 PCI to PCI bridge	OK
		0x0000E000-0x0000FFFF	Intel(R) PRO/100 S Dual Port Server Adapter	#3 OK
		0x0000E040-0x0000E07F	Intel(R) PRO/100 S Dual Port Server Adapter	#4 OK
		0x0000D000-0x0000D3FF	PCI bus	OK
		0x0000D400-0x0000D5FF	PCI bus	OK
		0x0000D400-0x0000D5FF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
		0x0000D500-0x0000D5FF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
		[IRQs]		
		Resource Device Status		
		IRQ 9 Microsoft ACPI-Compliant System		OK
		IRQ 39 RAGE XL PCI		OK
		IRQ 5 Other PCI Bridge Device		OK
		IRQ 1 Standard 101/102-Key or Microsoft Natural PS/2 Keyboard		OK
		IRQ 12 PS/2 Compatible Mouse		OK

IRQ 6	Standard floppy disk controller	OK		0xE9010000-0xE901FFFF	LSI Logic PCI-X Ultra320 SCSI Host Adapter	OK
IRQ 3	Communications Port (COM1)	OK		0xE9020000-0xE902FFFF	LSI Logic PCI-X Ultra320 SCSI Host Adapter	OK
IRQ 0	System timer	OK		0xE9030000-0xE903FFFF	LSI Logic PCI-X Ultra320 SCSI Host Adapter	OK
IRQ 8	System CMOS/real time clock	OK		0xE9100000-0xE910FFFF	PCI bus	OK
IRQ 13	Numeric data processor	OK		0xE9120000-0xE9120FFF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
IRQ 14	Primary IDE Channel	OK		0xE9121000-0xE9121FFF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
IRQ 15	Secondary IDE Channel	OK		0xE9300000-0xE930FFFF	PCI bus	OK
IRQ 40	LSI Logic PCI-X Ultra320 SCSI Host Adapter	OK		0xE9320000-0xE9320FFF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
IRQ 41	LSI Logic PCI-X Ultra320 SCSI Host Adapter	OK		0xE8800000-0xE880FFFF	PCI bus	OK
IRQ 51	QLogic QLA23xx PCI Fibre Channel Adapter	OK		0xEA000000-0xEA00FFFF	PCI bus	OK
IRQ 67	QLogic QLA23xx PCI Fibre Channel Adapter	OK		0xE8000000-0xE800FFFF	PCI bus	OK
IRQ 59	QLogic QLA23xx PCI Fibre Channel Adapter	OK		0xEA400000-0xEA40FFFF	PCI bus	OK
IRQ 10	Other PCI Bridge Device	OK		0xE8420000-0xE8420FFF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
IRQ 193	LSI Logic PCI-X Ultra320 SCSI Host Adapter	OK		0xF0000000-0xF000FFFF	PCI bus	OK
IRQ 194	LSI Logic PCI-X Ultra320 SCSI Host Adapter	OK		0xF2000000-0xF200FFFF	PCI bus	OK
IRQ 173	QLogic QLA23xx PCI Fibre Channel Adapter	OK		0xF0800000-0xF080FFFF	PCI bus	OK
IRQ 174	QLogic QLA23xx PCI Fibre Channel Adapter	OK		0xF2800000-0xF280FFFF	PCI bus	OK
IRQ 175	QLogic QLA23xx PCI Fibre Channel Adapter	OK		0xF0C20000-0xF0C20FFF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
IRQ 177	QLogic QLA23xx PCI Fibre Channel Adapter	OK		0xF1000000-0xF100FFFF	PCI bus	OK
IRQ 109	QLogic QLA23xx PCI Fibre Channel Adapter	OK		0xF2C00000-0xF2C0FFFF	PCI bus	OK
IRQ 102	QLogic QLA23xx PCI Fibre Channel Adapter	OK		0xF1420000-0xF1420FFF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
IRQ 141	Intel(R) PRO/100 S Dual Port Server Adapter #3	OK		0xEC000000-0xEC00FFFF	PCI bus	OK
IRQ 142	Intel(R) PRO/100 S Dual Port Server Adapter #4	OK		0xEE800000-0xEE80FFFF	PCI bus	OK
IRQ 123	QLogic QLA23xx PCI Fibre Channel Adapter	OK		0xEC400000-0xEC40FFFF	Intel 21154 PCI to PCI bridge	OK
IRQ 127	QLogic QLA23xx PCI Fibre Channel Adapter	OK		0xEC400000-0xEC40FFFF	Intel(R) PRO/100 S Dual Port Server Adapter #3	OK

[Memory]

Resource	Device	Status	
0xA0000-0xA7FFF	PCI bus	OK	
0xA0000-0xA7FFF	RAGE XL PCI	OK	
0xA8000-0xAFFFF	PCI bus	OK	
0xB0000-0xB7FFF	PCI bus	OK	
0xB8000-0xBFFFF	PCI bus	OK	
0xF4000000-0xF5FFFFFFF	PCI bus	OK	
0xF4000000-0xF5FFFFFFF	RAGE XL PCI	OK	
0xF5200000-0xF5200FFF	RAGE XL PCI	OK	
0xF5000000-0xF51FFFFFFF	Other PCI Bridge Device	OK	
0x0400-0x04FF	System board	OK	
0x100000-0xBFFAC33F	Memory Module	OK	
0xF7600000-0xF76FFFFFFF	PCI bus	OK	
0xF7600000-0xF76FFFFFFF	LSI Logic PCI-X Ultra320 SCSI Host Adapter	OK	
0xF7610000-0xF7610FFF	LSI Logic PCI-X Ultra320 SCSI Host Adapter	OK	
0xF7620000-0xF7620FFF	LSI Logic PCI-X Ultra320 SCSI Host Adapter	OK	
0xF7630000-0xF7630FFF	LSI Logic PCI-X Ultra320 SCSI Host Adapter	OK	
0xF6000000-0xF67FFFFFFF	PCI bus	OK	
0xF8000000-0xF83FFFFFFF	PCI bus	OK	
0xF6420000-0xF6420FFF	QLogic QLA23xx PCI Fibre Channel Adapter	OK	
0xF7000000-0xF73FFFFFFF	PCI bus	OK	
0xF8400000-0xF87FFFFFFF	PCI bus	OK	
0xF7800000-0xF78FFFFFFF	PCI bus	OK	
0xF7820000-0xF7820FFF	QLogic QLA23xx PCI Fibre Channel Adapter	OK	
0xF6800000-0xF6FFFFFFF	PCI bus	OK	
0xF8800000-0xF8BFFFFFFF	PCI bus	OK	
0xF6C20000-0xF6C20FFF	QLogic QLA23xx PCI Fibre Channel Adapter	OK	
0xE8C00000-0xE8DFFFFFFF	PCI bus	OK	
0xE8C00000-0xE8DFFFFFFF	Other PCI Bridge Device	OK	
0xE9000000-0xE90FFFFFFF	PCI bus	OK	
0xE9000000-0xE90FFFFFFF	LSI Logic PCI-X Ultra320 SCSI Host Adapter	OK	

0xE9010000-0xE901FFFF	LSI Logic PCI-X Ultra320 SCSI Host Adapter	OK
0xE9020000-0xE902FFFF	LSI Logic PCI-X Ultra320 SCSI Host Adapter	OK
0xE9030000-0xE903FFFF	LSI Logic PCI-X Ultra320 SCSI Host Adapter	OK
0xE9100000-0xE910FFFF	PCI bus	OK
0xE9120000-0xE9120FFF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
0xE9121000-0xE9121FFF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
0xE9300000-0xE930FFFF	PCI bus	OK
0xE9320000-0xE9320FFF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
0xE8800000-0xE880FFFF	PCI bus	OK
0xEA000000-0xEA00FFFF	PCI bus	OK
0xE8000000-0xE800FFFF	PCI bus	OK
0xEA400000-0xEA40FFFF	PCI bus	OK
0xE8420000-0xE8420FFF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
0xF0000000-0xF000FFFF	PCI bus	OK
0xF2000000-0xF200FFFF	PCI bus	OK
0xF0800000-0xF080FFFF	PCI bus	OK
0xF2800000-0xF280FFFF	PCI bus	OK
0xF0C20000-0xF0C20FFF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
0xF1000000-0xF100FFFF	PCI bus	OK
0xF2C00000-0xF2C0FFFF	PCI bus	OK
0xF1420000-0xF1420FFF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
0xEC000000-0xEC00FFFF	PCI bus	OK
0xEE800000-0xEE80FFFF	PCI bus	OK
0xEC400000-0xEC40FFFF	Intel 21154 PCI to PCI bridge	OK
0xEC400000-0xEC40FFFF	Intel(R) PRO/100 S Dual Port Server Adapter #3	OK
0xEC440000-0xEC440FFF	Intel(R) PRO/100 S Dual Port Server Adapter #3	OK
0xEC441000-0xEC441FFF	Intel(R) PRO/100 S Dual Port Server Adapter #4	OK
0xEC420000-0xEC420FFF	Intel(R) PRO/100 S Dual Port Server Adapter #4	OK
0xEC800000-0xEC80FFFF	PCI bus	OK
0xEE000000-0xEE00FFFF	PCI bus	OK
0xED200000-0xED20FFFF	PCI bus	OK
0xED220000-0xED220FFF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
0xED221000-0xED221FFF	QLogic QLA23xx PCI Fibre Channel Adapter	OK

[Components]

[Multimedia]

[Audio Codecs]

CODEC	Manufacturer	Description	Status	File
Version	Size	Creation Date		
c:\windows\system32\sl_anet.acm			Sipro Lab Telecom Inc.	
Sipro Lab Telecom Audio Codec			OK	
C:\WINDOWS\system32\SL_ANET.ACM	3.02	84.00 KB (86,016 bytes)		3/25/2003 7:00 AM
c:\windows\system32\msaud32.acm			Microsoft Corporation	
Windows Media Audio Codec			OK	

C:\WINDOWS\system32\MSAUD32.ACM 8.00.00.4487 288.00 KB  
 (294,912 bytes) 3/25/2003 7:00 AM  
 c:\windows\system32\imaadp32.acm Microsoft Corporation  
 OK C:\WINDOWS\system32\IMAADP32.ACM 5.2.3790.0  
 (srv03\_rtm.030324-2048) 15.50 KB (15,872 bytes) 3/25/2003  
 7:00 AM  
 c:\windows\system32\msg711.acm Microsoft Corporation  
 OK C:\WINDOWS\system32\MSG711.ACM 5.2.3790.0  
 (srv03\_rtm.030324-2048) 10.00 KB (10,240 bytes) 3/25/2003  
 7:00 AM  
 c:\windows\system32\tssoft32.acm DSP GROUP, INC.  
 OK C:\WINDOWS\system32\TSSOFT32.ACM 1.01 9.50 KB  
 (9,728 bytes) 3/25/2003 7:00 AM  
 c:\windows\system32\msg723.acm Microsoft Corporation  
 OK C:\WINDOWS\system32\MSG723.ACM 4.4.4000 116.00 KB  
 (118,784 bytes) 4/10/2003 10:26 AM  
 c:\windows\system32\msadp32.acm Microsoft Corporation  
 OK C:\WINDOWS\system32\MSADP32.ACM 5.2.3790.0  
 (srv03\_rtm.030324-2048) 14.50 KB (14,848 bytes) 3/25/2003  
 7:00 AM  
 c:\windows\system32\msgsm32.acm Microsoft Corporation  
 OK C:\WINDOWS\system32\MSGSM32.ACM 5.2.3790.0  
 (srv03\_rtm.030324-2048) 20.50 KB (20,992 bytes) 3/25/2003  
 7:00 AM

[Video Codecs]

CODEC	Manufacturer	Description	Status	File
Version	Size	Creation Date		
c:\windows\system32\msh261.drv	Microsoft Corporation			
OK	C:\WINDOWS\system32\MSH261.DRV	4.4.4000	180.00 KB	(184,320 bytes) 4/10/2003 10:26 AM
c:\windows\system32\msrle32.dll	Microsoft Corporation			
OK	C:\WINDOWS\system32\MSRLE32.DLL	5.2.3790.0		(srv03_rtm.030324-2048) 10.50 KB (10,752 bytes) 3/25/2003 7:00 AM
c:\windows\system32\msvidc32.dll	Microsoft Corporation			
OK	C:\WINDOWS\system32\MSVIDC32.DLL	5.2.3790.0		(srv03_rtm.030324-2048) 26.50 KB (27,136 bytes) 3/25/2003 7:00 AM
c:\windows\system32\msh263.drv	Microsoft Corporation			
OK	C:\WINDOWS\system32\MSH263.DRV	4.4.4000	284.00 KB	(290,816 bytes) 3/24/2003 8:46 PM
c:\windows\system32\tsbyuv.dll	Microsoft Corporation			
OK	C:\WINDOWS\system32\TSBYUV.DLL	5.2.3790.0		(srv03_rtm.030324-2048) 8.00 KB (8,192 bytes) 3/24/2003 8:50 PM
c:\windows\system32\iyuv_32.dll	Microsoft Corporation			
OK	C:\WINDOWS\system32\IYUV_32.DLL	5.2.3790.0		(srv03_rtm.030324-2048) 45.00 KB (46,080 bytes) 3/24/2003 8:49 PM
c:\windows\system32\msyuv.dll	Microsoft Corporation			
OK	C:\WINDOWS\system32\MSYUV.DLL	5.2.3790.0		(srv03_rtm.030324-2048) 16.50 KB (16,896 bytes) 3/24/2003 8:49 PM

[CD-ROM]

Item	Value
Drive	D:
Description	CD-ROM Drive
Media Loaded	Yes
Media Type	CD-ROM
Name	HL-DT-ST DVD-ROM GDR8081N
Manufacturer	(Standard CD-ROM drives)
Status	OK
Transfer Rate	2706.77 kbytes/sec
SCSI Target ID	0

PNP Device ID  
 IDE\CDROMHL-DT-ST\_DVD-ROM\_GDR8081N\_\_\_\_\_0012\_\_\_\_\_  
 \_5&CBC355F&0&0.0.0  
 Driver c:\windows\system32\drivers\cdrom.sys (5.2.3790.0  
 (srv03\_rtm.030324-2048), 49.50 KB (50,688 bytes), 3/25/2003 7:00 AM)

[Sound Device]

Item	Value
------	-------

[Display]

Item	Value
Name	RAGE XL PCI
PNP Device ID	PCI\VEN_1002&DEV_4752&SUBSYS_02A61014&REV_27\3&267A616A&0&18
Adapter Type	ATI RAGE XL PCI (B41), ATI Technologies Inc. compatible
Adapter Description	RAGE XL PCI
Adapter RAM	8.00 MB (8,388,608 bytes)
Installed Drivers	ati2drad.dll
Driver Version	5.00.2195.5012
INF File	oem10.inf (ati2mpad section)
Color Planes	1
Color Table Entries	65536
Resolution	1024 x 768 x 70 hertz
Bits/Pixel	16
Memory Address	0xF4000000-0xF5FFFFFF
I/O Port	0x00001800-0x000018FF
Memory Address	0xF5200000-0xF5200FFF
IRQ Channel	IRQ 39
I/O Port	0x000003B0-0x000003BB
I/O Port	0x000003C0-0x000003DF
Memory Address	0xA0000-0xA7FFF
Driver	c:\windows\system32\drivers\ati2mpad.sys (5.00.2195.5012, 316.20 KB (323,793 bytes), 12/21/2001 10:09 AM)

[Infrared]

Item	Value
------	-------

[Input]

[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&7FD7688&0
Number of Function Keys	12
IRQ Channel	IRQ 1
I/O Port	0x00000064-0x00000064
I/O Port	0x00000060-0x00000060
Driver	c:\windows\system32\drivers\i8042prt.sys (5.2.3790.0 (srv03_rtm.030324-2048), 68.50 KB (70,144 bytes), 3/25/2003 7:00 AM)

[Pointing Device]

Item	Value
Hardware Type	PS/2 Compatible Mouse
Number of Buttons	3
Status	OK
PNP Device ID	ACPI\PNP0F13\4&7FD7688&0

Power Management Supported No  
Double Click Threshold 6  
Handedness Right Handed Operation  
IRQ Channel IRQ 12  
Driver c:\windows\system32\drivers\i8042prt.sys (5.2.3790.0  
(srv03\_rtm.030324-2048), 68.50 KB (70,144 bytes), 3/25/2003 7:00 AM)

[Modem]

Item Value

[Network]

[Adapter]

Item Value  
Name [00000001] RAS Async Adapter  
Adapter Type Not Available  
Product Type RAS Async Adapter  
Installed Yes  
PNP Device ID Not Available  
Last Reset 7/15/2003 4:12 PM  
Index 1  
Service Name AsyncMac  
IP Address Not Available  
IP Subnet Not Available  
Default IP Gateway Not Available  
DHCP Enabled No  
DHCP Server Not Available  
DHCP Lease Expires Not Available  
DHCP Lease Obtained Not Available  
MAC Address Not Available

Name [00000002] WAN Miniport (L2TP)  
Adapter Type Not Available  
Product Type WAN Miniport (L2TP)  
Installed Yes  
PNP Device ID ROOT\MS\_L2TPMINIPOINT\0000  
Last Reset 7/15/2003 4:12 PM  
Index 2  
Service Name Rasl2tp  
IP Address Not Available  
IP Subnet Not Available  
Default IP Gateway Not Available  
DHCP Enabled No  
DHCP Server Not Available  
DHCP Lease Expires Not Available  
DHCP Lease Obtained Not Available  
MAC Address Not Available  
Driver c:\windows\system32\drivers\rasl2tp.sys (5.2.3790.0  
(srv03\_rtm.030324-2048), 77.00 KB (78,848 bytes), 3/25/2003 7:00 AM)

Name [00000003] WAN Miniport (PPTP)  
Adapter Type Wide Area Network (WAN)  
Product Type WAN Miniport (PPTP)  
Installed Yes  
PNP Device ID ROOT\MS\_PPTPMINIPOINT\0000  
Last Reset 7/15/2003 4:12 PM  
Index 3  
Service Name PptpMiniport  
IP Address Not Available  
IP Subnet Not Available  
Default IP Gateway Not Available  
DHCP Enabled No  
DHCP Server Not Available  
DHCP Lease Expires Not Available

DHCP Lease Obtained Not Available  
MAC Address 50:50:54:50:30:30  
Driver c:\windows\system32\drivers\rasppoe.sys (5.2.3790.0  
(srv03\_rtm.030324-2048), 70.50 KB (72,192 bytes), 3/25/2003 7:00 AM)

Name [00000004] WAN Miniport (PPPOE)  
Adapter Type Wide Area Network (WAN)  
Product Type WAN Miniport (PPPOE)  
Installed Yes  
PNP Device ID ROOT\MS\_PPPOEMINIPOINT\0000  
Last Reset 7/15/2003 4:12 PM

Index 4  
Service Name RasPppoe  
IP Address Not Available  
IP Subnet Not Available  
Default IP Gateway Not Available  
DHCP Enabled No  
DHCP Server Not Available  
DHCP Lease Expires Not Available  
DHCP Lease Obtained Not Available  
MAC Address 33:50:6F:45:30:30  
Driver c:\windows\system32\drivers\rasppoe.sys (5.2.3790.0  
(srv03\_rtm.030324-2048), 38.00 KB (38,912 bytes), 3/25/2003 7:00 AM)

Name [00000005] Direct Parallel  
Adapter Type Not Available  
Product Type Direct Parallel  
Installed Yes  
PNP Device ID ROOT\MS\_PTMINIPOINT\0000  
Last Reset 7/15/2003 4:12 PM

Index 5  
Service Name Raspti  
IP Address Not Available  
IP Subnet Not Available  
Default IP Gateway Not Available  
DHCP Enabled No  
DHCP Server Not Available  
DHCP Lease Expires Not Available  
DHCP Lease Obtained Not Available  
MAC Address Not Available  
Driver c:\windows\system32\drivers\raspti.sys (5.2.3790.0  
(srv03\_rtm.030324-2048), 18.50 KB (18,944 bytes), 3/25/2003 7:00 AM)

Name [00000006] WAN Miniport (IP)  
Adapter Type Not Available  
Product Type WAN Miniport (IP)  
Installed Yes  
PNP Device ID ROOT\MS\_NDISWANIP\0000  
Last Reset 7/15/2003 4:12 PM

Index 6  
Service Name NdisWan  
IP Address Not Available  
IP Subnet Not Available  
Default IP Gateway Not Available  
DHCP Enabled No  
DHCP Server Not Available  
DHCP Lease Expires Not Available  
DHCP Lease Obtained Not Available  
MAC Address Not Available  
Driver c:\windows\system32\drivers\ndiswan.sys (5.2.3790.0  
(srv03\_rtm.030324-2048), 96.50 KB (98,816 bytes), 3/25/2003 7:00 AM)

Name [00000007] Broadcom NetXtreme Gigabit Ethernet  
Adapter Type Not Available  
Product Type Broadcom NetXtreme Gigabit Ethernet  
Installed Yes

PNP Device ID  
 PCI\VEN\_14E4&DEV\_1648&SUBSYS\_02A61014&REV\_02\3&13C0B0C5  
 &0&20  
 Last Reset 7/15/2003 4:12 PM  
 Index 7  
 Service Name b57w2k  
 IP Address Not Available  
 IP Subnet Not Available  
 Default IP Gateway Not Available  
 DHCP Enabled Yes  
 DHCP Server Not Available  
 DHCP Lease Expires Not Available  
 DHCP Lease Obtained Not Available  
 MAC Address Not Available  
 Driver c:\windows\system32\drivers\b57xp32.sys (6.34.0.0 built by:  
 WinDDK, 166.88 KB (170,880 bytes), 4/16/2003 9:56 AM)

Name [00000008] Broadcom NetXtreme Gigabit Ethernet  
 Adapter Type Not Available  
 Product Type Broadcom NetXtreme Gigabit Ethernet  
 Installed Yes  
 PNP Device ID  
 PCI\VEN\_14E4&DEV\_1648&SUBSYS\_02A61014&REV\_02\3&13C0B0C5  
 &0&21  
 Last Reset 7/15/2003 4:12 PM  
 Index 8  
 Service Name b57w2k  
 IP Address Not Available  
 IP Subnet Not Available  
 Default IP Gateway Not Available  
 DHCP Enabled Yes  
 DHCP Server Not Available  
 DHCP Lease Expires Not Available  
 DHCP Lease Obtained Not Available  
 MAC Address Not Available  
 Driver c:\windows\system32\drivers\b57xp32.sys (6.34.0.0 built by:  
 WinDDK, 166.88 KB (170,880 bytes), 4/16/2003 9:56 AM)

Name [00000009] Broadcom NetXtreme Gigabit Ethernet  
 Adapter Type Not Available  
 Product Type Broadcom NetXtreme Gigabit Ethernet  
 Installed Yes  
 PNP Device ID  
 PCI\VEN\_14E4&DEV\_1648&SUBSYS\_02A61014&REV\_02\3&1D521019&  
 0&20  
 Last Reset 7/15/2003 4:12 PM  
 Index 9  
 Service Name b57w2k  
 IP Address Not Available  
 IP Subnet Not Available  
 Default IP Gateway Not Available  
 DHCP Enabled Yes  
 DHCP Server Not Available  
 DHCP Lease Expires Not Available  
 DHCP Lease Obtained Not Available  
 MAC Address Not Available  
 Driver c:\windows\system32\drivers\b57xp32.sys (6.34.0.0 built by:  
 WinDDK, 166.88 KB (170,880 bytes), 4/16/2003 9:56 AM)

Name [00000010] Broadcom NetXtreme Gigabit Ethernet  
 Adapter Type Not Available  
 Product Type Broadcom NetXtreme Gigabit Ethernet  
 Installed Yes  
 PNP Device ID  
 PCI\VEN\_14E4&DEV\_1648&SUBSYS\_02A61014&REV\_02\3&1D521019&  
 0&21  
 Last Reset 7/15/2003 4:12 PM  
 Index 10

Service Name b57w2k  
 IP Address Not Available  
 IP Subnet Not Available  
 Default IP Gateway Not Available  
 DHCP Enabled Yes  
 DHCP Server Not Available  
 DHCP Lease Expires Not Available  
 DHCP Lease Obtained Not Available  
 MAC Address Not Available  
 Driver c:\windows\system32\drivers\b57xp32.sys (6.34.0.0 built by:  
 WinDDK, 166.88 KB (170,880 bytes), 4/16/2003 9:56 AM)

Name [00000011] Intel(R) PRO/100 S Dual Port Server Adapter  
 Adapter Type Not Available  
 Product Type Intel(R) PRO/100 S Dual Port Server Adapter  
 Installed Yes  
 PNP Device ID Not Available  
 Last Reset 7/15/2003 4:12 PM  
 Index 11  
 Service Name E100B  
 IP Address Not Available  
 IP Subnet Not Available  
 Default IP Gateway Not Available  
 DHCP Enabled No  
 DHCP Server Not Available  
 DHCP Lease Expires Not Available  
 DHCP Lease Obtained Not Available  
 MAC Address Not Available

Name [00000012] Intel(R) PRO/100 S Dual Port Server Adapter  
 Adapter Type Not Available  
 Product Type Intel(R) PRO/100 S Dual Port Server Adapter  
 Installed Yes  
 PNP Device ID Not Available  
 Last Reset 7/15/2003 4:12 PM  
 Index 12  
 Service Name E100B  
 IP Address Not Available  
 IP Subnet Not Available  
 Default IP Gateway Not Available  
 DHCP Enabled No  
 DHCP Server Not Available  
 DHCP Lease Expires Not Available  
 DHCP Lease Obtained Not Available  
 MAC Address Not Available

Name [00000013] Intel(R) PRO/100 S Dual Port Server Adapter  
 Adapter Type Ethernet 802.3  
 Product Type Intel(R) PRO/100 S Dual Port Server Adapter  
 Installed Yes  
 PNP Device ID  
 PCI\VEN\_8086&DEV\_1229&SUBSYS\_10158086&REV\_0D\4&18FB0D9C  
 &0&2010  
 Last Reset 7/15/2003 4:12 PM  
 Index 13  
 Service Name E100B  
 IP Address 192.168.125.232  
 IP Subnet 255.255.255.0  
 Default IP Gateway Not Available  
 DHCP Enabled No  
 DHCP Server Not Available  
 DHCP Lease Expires Not Available  
 DHCP Lease Obtained Not Available  
 MAC Address 00:02:B3:A7:30:63  
 Memory Address 0xEC440000-0xEC440FFF  
 I/O Port 0x0000E000-0x0000FFFF  
 Memory Address 0xEC400000-0xEC4FFFFF  
 IRQ Channel IRQ 141

Driver c:\windows\system32\drivers\e100b325.sys (7.0.26.0 built by: WinDDK, 142.00 KB (145,408 bytes), 3/4/2003 10:56 AM)

Name [00000014] Intel(R) PRO/100 S Dual Port Server Adapter  
Adapter Type Ethernet 802.3  
Product Type Intel(R) PRO/100 S Dual Port Server Adapter  
Installed Yes  
PNP Device ID  
PCI\VEN\_8086&DEV\_1229&SUBSYS\_10158086&REV\_0D\4&18FB0D9C&0&2810  
Last Reset 7/15/2003 4:12 PM  
Index 14  
Service Name E100B  
IP Address 192.168.128.10  
IP Subnet 255.255.255.0  
Default IP Gateway Not Available  
DHCP Enabled No  
DHCP Server Not Available  
DHCP Lease Expires Not Available  
DHCP Lease Obtained Not Available  
MAC Address 00:02:B3:A7:30:64  
Memory Address 0xEC441000-0xEC441FFF  
I/O Port 0x0000E040-0x0000E07F  
Memory Address 0xEC420000-0xEC43FFFF  
IRQ Channel IRQ 142  
Driver c:\windows\system32\drivers\e100b325.sys (7.0.26.0 built by: WinDDK, 142.00 KB (145,408 bytes), 3/4/2003 10:56 AM)

[Protocol]

Item	Value
Name	MSAFD Tcpip [TCP/IP]
Connectionless Service	No
Guarantees Delivery	Yes
Guarantees Sequencing	Yes
Maximum Address Size	16 bytes
Maximum Message Size	0 bytes
Message Oriented	No
Minimum Address Size	16 bytes
Pseudo Stream Oriented	No
Supports Broadcasting	No
Supports Connect Data	No
Supports Disconnect Data	No
Supports Encryption	No
Supports Expedited Data	Yes
Supports Graceful Closing	Yes
Supports Guaranteed Bandwidth	No
Supports Multicasting	No

Name	MSAFD Tcpip [UDP/IP]
Connectionless Service	Yes
Guarantees Delivery	No
Guarantees Sequencing	No
Maximum Address Size	16 bytes
Maximum Message Size	63.93 KB (65,467 bytes)
Message Oriented	Yes
Minimum Address Size	16 bytes
Pseudo Stream Oriented	No
Supports Broadcasting	Yes
Supports Connect Data	No
Supports Disconnect Data	No
Supports Encryption	No
Supports Expedited Data	No
Supports Graceful Closing	No
Supports Guaranteed Bandwidth	No
Supports Multicasting	Yes

Name RSVP UDP Service Provider

Connectionless Service	Yes
Guarantees Delivery	No
Guarantees Sequencing	No
Maximum Address Size	16 bytes
Maximum Message Size	63.93 KB (65,467 bytes)
Message Oriented	Yes
Minimum Address Size	16 bytes
Pseudo Stream Oriented	No
Supports Broadcasting	Yes
Supports Connect Data	No
Supports Disconnect Data	No
Supports Encryption	Yes
Supports Expedited Data	No
Supports Graceful Closing	No
Supports Guaranteed Bandwidth	No
Supports Multicasting	Yes

Name	RSVP TCP Service Provider
Connectionless Service	No
Guarantees Delivery	Yes
Guarantees Sequencing	Yes
Maximum Address Size	16 bytes
Maximum Message Size	0 bytes
Message Oriented	No
Minimum Address Size	16 bytes
Pseudo Stream Oriented	No
Supports Broadcasting	No
Supports Connect Data	No
Supports Disconnect Data	No
Supports Encryption	Yes
Supports Expedited Data	Yes
Supports Graceful Closing	Yes
Supports Guaranteed Bandwidth	No
Supports Multicasting	No

Name	MSAFD NetBIOS [\Device\NetBT_Tcpip_{17A02C9E-6E11-4588-835C-DCA46ADAB834}]
SEQPACKET	9
Connectionless Service	No
Guarantees Delivery	Yes
Guarantees Sequencing	Yes
Maximum Address Size	20 bytes
Maximum Message Size	62.50 KB (64,000 bytes)
Message Oriented	Yes
Minimum Address Size	20 bytes
Pseudo Stream Oriented	No
Supports Broadcasting	No
Supports Connect Data	No
Supports Disconnect Data	No
Supports Encryption	No
Supports Expedited Data	No
Supports Graceful Closing	No
Supports Guaranteed Bandwidth	No
Supports Multicasting	No

Name	MSAFD NetBIOS [\Device\NetBT_Tcpip_{17A02C9E-6E11-4588-835C-DCA46ADAB834}]
DATAGRAM	9
Connectionless Service	Yes
Guarantees Delivery	No
Guarantees Sequencing	No
Maximum Address Size	20 bytes
Maximum Message Size	62.50 KB (64,000 bytes)
Message Oriented	Yes
Minimum Address Size	20 bytes
Pseudo Stream Oriented	No
Supports Broadcasting	Yes
Supports Connect Data	No

Supports Disconnect Data No  
 Supports Encryption No  
 Supports Expedited Data No  
 Supports Graceful Closing No  
 Supports Guaranteed Bandwidth No  
 Supports Multicasting No

Name MSAFD NetBIOS  
 [\Device\NetBT\_Tcpip\_{73066ADB-3CD3-4429-B6D7-F12D6A3D9FA7}]  
 SEQPACKET 8  
 Connectionless Service No  
 Guarantees Delivery Yes  
 Guarantees Sequencing Yes  
 Maximum Address Size 20 bytes  
 Maximum Message Size 62.50 KB (64,000 bytes)  
 Message Oriented Yes  
 Minimum Address Size 20 bytes  
 Pseudo Stream Oriented No  
 Supports Broadcasting No  
 Supports Connect Data No  
 Supports Disconnect Data No  
 Supports Encryption No  
 Supports Expedited Data No  
 Supports Graceful Closing No  
 Supports Guaranteed Bandwidth No  
 Supports Multicasting No

Name MSAFD NetBIOS  
 [\Device\NetBT\_Tcpip\_{73066ADB-3CD3-4429-B6D7-F12D6A3D9FA7}]  
 DATAGRAM 8  
 Connectionless Service Yes  
 Guarantees Delivery No  
 Guarantees Sequencing No  
 Maximum Address Size 20 bytes  
 Maximum Message Size 62.50 KB (64,000 bytes)  
 Message Oriented Yes  
 Minimum Address Size 20 bytes  
 Pseudo Stream Oriented No  
 Supports Broadcasting Yes  
 Supports Connect Data No  
 Supports Disconnect Data No  
 Supports Encryption No  
 Supports Expedited Data No  
 Supports Graceful Closing No  
 Supports Guaranteed Bandwidth No  
 Supports Multicasting No

Name MSAFD NetBIOS  
 [\Device\NetBT\_Tcpip\_{5B5E9384-911D-4840-9799-608987251691}]  
 SEQPACKET 7  
 Connectionless Service No  
 Guarantees Delivery Yes  
 Guarantees Sequencing Yes  
 Maximum Address Size 20 bytes  
 Maximum Message Size 62.50 KB (64,000 bytes)  
 Message Oriented Yes  
 Minimum Address Size 20 bytes  
 Pseudo Stream Oriented No  
 Supports Broadcasting No  
 Supports Connect Data No  
 Supports Disconnect Data No  
 Supports Encryption No  
 Supports Expedited Data No  
 Supports Graceful Closing No  
 Supports Guaranteed Bandwidth No  
 Supports Multicasting No

Name MSAFD NetBIOS  
 [\Device\NetBT\_Tcpip\_{5B5E9384-911D-4840-9799-608987251691}]  
 DATAGRAM 7  
 Connectionless Service Yes  
 Guarantees Delivery No  
 Guarantees Sequencing No  
 Maximum Address Size 20 bytes  
 Maximum Message Size 62.50 KB (64,000 bytes)  
 Message Oriented Yes  
 Minimum Address Size 20 bytes  
 Pseudo Stream Oriented No  
 Supports Broadcasting Yes  
 Supports Connect Data No  
 Supports Disconnect Data No  
 Supports Encryption No  
 Supports Expedited Data No  
 Supports Graceful Closing No  
 Supports Guaranteed Bandwidth No  
 Supports Multicasting No

Name MSAFD NetBIOS  
 [\Device\NetBT\_Tcpip\_{0F0205EF-A50D-4D31-996A-D4B587F902F0}]  
 SEQPACKET 6  
 Connectionless Service No  
 Guarantees Delivery Yes  
 Guarantees Sequencing Yes  
 Maximum Address Size 20 bytes  
 Maximum Message Size 62.50 KB (64,000 bytes)  
 Message Oriented Yes  
 Minimum Address Size 20 bytes  
 Pseudo Stream Oriented No  
 Supports Broadcasting No  
 Supports Connect Data No  
 Supports Disconnect Data No  
 Supports Encryption No  
 Supports Expedited Data No  
 Supports Graceful Closing No  
 Supports Guaranteed Bandwidth No  
 Supports Multicasting No

Name MSAFD NetBIOS  
 [\Device\NetBT\_Tcpip\_{0F0205EF-A50D-4D31-996A-D4B587F902F0}]  
 DATAGRAM 6  
 Connectionless Service Yes  
 Guarantees Delivery No  
 Guarantees Sequencing No  
 Maximum Address Size 20 bytes  
 Maximum Message Size 62.50 KB (64,000 bytes)  
 Message Oriented Yes  
 Minimum Address Size 20 bytes  
 Pseudo Stream Oriented No  
 Supports Broadcasting Yes  
 Supports Connect Data No  
 Supports Disconnect Data No  
 Supports Encryption No  
 Supports Expedited Data No  
 Supports Graceful Closing No  
 Supports Guaranteed Bandwidth No  
 Supports Multicasting No

Name MSAFD NetBIOS  
 [\Device\NetBT\_Tcpip\_{AF999C9B-2C8C-4C05-B890-0FA343F778B6}]  
 SEQPACKET 5  
 Connectionless Service No  
 Guarantees Delivery Yes  
 Guarantees Sequencing Yes  
 Maximum Address Size 20 bytes  
 Maximum Message Size 62.50 KB (64,000 bytes)



Message Oriented Yes  
 Minimum Address Size 20 bytes  
 Pseudo Stream Oriented No  
 Supports Broadcasting No  
 Supports Connect Data No  
 Supports Disconnect Data No  
 Supports Encryption No  
 Supports Expedited Data No  
 Supports Graceful Closing No  
 Supports Guaranteed Bandwidth No  
 Supports Multicasting No

Name MSAFD NetBIOS  
 [\Device\NetBT\_Tcpip\_{AF999C9B-2C8C-4C05-B890-0FA343F778B6}]  
 DATAGRAM 5  
 Connectionless Service Yes  
 Guarantees Delivery No  
 Guarantees Sequencing No  
 Maximum Address Size 20 bytes  
 Maximum Message Size 62.50 KB (64,000 bytes)  
 Message Oriented Yes  
 Minimum Address Size 20 bytes  
 Pseudo Stream Oriented No  
 Supports Broadcasting Yes  
 Supports Connect Data No  
 Supports Disconnect Data No  
 Supports Encryption No  
 Supports Expedited Data No  
 Supports Graceful Closing No  
 Supports Guaranteed Bandwidth No  
 Supports Multicasting No

Name MSAFD NetBIOS  
 [\Device\NetBT\_Tcpip\_{811C0403-9D5B-43AA-AFE0-F13D1D459C5F}]  
 SEQPACKET 4  
 Connectionless Service No  
 Guarantees Delivery Yes  
 Guarantees Sequencing Yes  
 Maximum Address Size 20 bytes  
 Maximum Message Size 62.50 KB (64,000 bytes)  
 Message Oriented Yes  
 Minimum Address Size 20 bytes  
 Pseudo Stream Oriented No  
 Supports Broadcasting No  
 Supports Connect Data No  
 Supports Disconnect Data No  
 Supports Encryption No  
 Supports Expedited Data No  
 Supports Graceful Closing No  
 Supports Guaranteed Bandwidth No  
 Supports Multicasting No

Name MSAFD NetBIOS  
 [\Device\NetBT\_Tcpip\_{811C0403-9D5B-43AA-AFE0-F13D1D459C5F}]  
 DATAGRAM 4  
 Connectionless Service Yes  
 Guarantees Delivery No  
 Guarantees Sequencing No  
 Maximum Address Size 20 bytes  
 Maximum Message Size 62.50 KB (64,000 bytes)  
 Message Oriented Yes  
 Minimum Address Size 20 bytes  
 Pseudo Stream Oriented No  
 Supports Broadcasting Yes  
 Supports Connect Data No  
 Supports Disconnect Data No  
 Supports Encryption No  
 Supports Expedited Data No

Supports Graceful Closing No  
 Supports Guaranteed Bandwidth No  
 Supports Multicasting No

Name MSAFD NetBIOS  
 [\Device\NetBT\_Tcpip\_{747202E3-E6E5-47BF-9288-D55EB9D7EF32}]  
 SEQPACKET 3  
 Connectionless Service No  
 Guarantees Delivery Yes  
 Guarantees Sequencing Yes  
 Maximum Address Size 20 bytes  
 Maximum Message Size 62.50 KB (64,000 bytes)  
 Message Oriented Yes  
 Minimum Address Size 20 bytes  
 Pseudo Stream Oriented No  
 Supports Broadcasting No  
 Supports Connect Data No  
 Supports Disconnect Data No  
 Supports Encryption No  
 Supports Expedited Data No  
 Supports Graceful Closing No  
 Supports Guaranteed Bandwidth No  
 Supports Multicasting No

Name MSAFD NetBIOS  
 [\Device\NetBT\_Tcpip\_{747202E3-E6E5-47BF-9288-D55EB9D7EF32}]  
 DATAGRAM 3  
 Connectionless Service Yes  
 Guarantees Delivery No  
 Guarantees Sequencing No  
 Maximum Address Size 20 bytes  
 Maximum Message Size 62.50 KB (64,000 bytes)  
 Message Oriented Yes  
 Minimum Address Size 20 bytes  
 Pseudo Stream Oriented No  
 Supports Broadcasting Yes  
 Supports Connect Data No  
 Supports Disconnect Data No  
 Supports Encryption No  
 Supports Expedited Data No  
 Supports Graceful Closing No  
 Supports Guaranteed Bandwidth No  
 Supports Multicasting No

Name MSAFD NetBIOS  
 [\Device\NetBT\_Tcpip\_{378CE042-9BCA-4EC4-A2FE-CC343D7A841E}]  
 SEQPACKET 0  
 Connectionless Service No  
 Guarantees Delivery Yes  
 Guarantees Sequencing Yes  
 Maximum Address Size 20 bytes  
 Maximum Message Size 62.50 KB (64,000 bytes)  
 Message Oriented Yes  
 Minimum Address Size 20 bytes  
 Pseudo Stream Oriented No  
 Supports Broadcasting No  
 Supports Connect Data No  
 Supports Disconnect Data No  
 Supports Encryption No  
 Supports Expedited Data No  
 Supports Graceful Closing No  
 Supports Guaranteed Bandwidth No  
 Supports Multicasting No

Name MSAFD NetBIOS  
 [\Device\NetBT\_Tcpip\_{378CE042-9BCA-4EC4-A2FE-CC343D7A841E}]  
 DATAGRAM 0  
 Connectionless Service Yes

Guarantees Delivery No  
 Guarantees Sequencing No  
 Maximum Address Size 20 bytes  
 Maximum Message Size 62.50 KB (64,000 bytes)  
 Message Oriented Yes  
 Minimum Address Size 20 bytes  
 Pseudo Stream Oriented No  
 Supports Broadcasting Yes  
 Supports Connect Data No  
 Supports Disconnect Data No  
 Supports Encryption No  
 Supports Expedited Data No  
 Supports Graceful Closing No  
 Supports Guaranteed Bandwidth No  
 Supports Multicasting No

Name MSAFD NetBIOS  
 [\Device\NetBT\_Tcpip\_{45083998-E918-40EA-AFD3-1D5597364964}]  
 SEQPACKET 1  
 Connectionless Service No  
 Guarantees Delivery Yes  
 Guarantees Sequencing Yes  
 Maximum Address Size 20 bytes  
 Maximum Message Size 62.50 KB (64,000 bytes)  
 Message Oriented Yes  
 Minimum Address Size 20 bytes  
 Pseudo Stream Oriented No  
 Supports Broadcasting No  
 Supports Connect Data No  
 Supports Disconnect Data No  
 Supports Encryption No  
 Supports Expedited Data No  
 Supports Graceful Closing No  
 Supports Guaranteed Bandwidth No  
 Supports Multicasting No

Name MSAFD NetBIOS  
 [\Device\NetBT\_Tcpip\_{45083998-E918-40EA-AFD3-1D5597364964}]  
 DATAGRAM 1  
 Connectionless Service Yes  
 Guarantees Delivery No  
 Guarantees Sequencing No  
 Maximum Address Size 20 bytes  
 Maximum Message Size 62.50 KB (64,000 bytes)  
 Message Oriented Yes  
 Minimum Address Size 20 bytes  
 Pseudo Stream Oriented No  
 Supports Broadcasting Yes  
 Supports Connect Data No  
 Supports Disconnect Data No  
 Supports Encryption No  
 Supports Expedited Data No  
 Supports Graceful Closing No  
 Supports Guaranteed Bandwidth No  
 Supports Multicasting No

Name MSAFD NetBIOS  
 [\Device\NetBT\_Tcpip\_{21365DBD-7EE7-4802-ACC1-D5C0E5FC2966}]  
 SEQPACKET 2  
 Connectionless Service No  
 Guarantees Delivery Yes  
 Guarantees Sequencing Yes  
 Maximum Address Size 20 bytes  
 Maximum Message Size 62.50 KB (64,000 bytes)  
 Message Oriented Yes  
 Minimum Address Size 20 bytes  
 Pseudo Stream Oriented No  
 Supports Broadcasting No

Supports Connect Data No  
 Supports Disconnect Data No  
 Supports Encryption No  
 Supports Expedited Data No  
 Supports Graceful Closing No  
 Supports Guaranteed Bandwidth No  
 Supports Multicasting No

Name MSAFD NetBIOS  
 [\Device\NetBT\_Tcpip\_{21365DBD-7EE7-4802-ACC1-D5C0E5FC2966}]  
 DATAGRAM 2  
 Connectionless Service Yes  
 Guarantees Delivery No  
 Guarantees Sequencing No  
 Maximum Address Size 20 bytes  
 Maximum Message Size 62.50 KB (64,000 bytes)  
 Message Oriented Yes  
 Minimum Address Size 20 bytes  
 Pseudo Stream Oriented No  
 Supports Broadcasting Yes  
 Supports Connect Data No  
 Supports Disconnect Data No  
 Supports Encryption No  
 Supports Expedited Data No  
 Supports Graceful Closing No  
 Supports Guaranteed Bandwidth No  
 Supports Multicasting No

[WinSock]

Item	Value
File	c:\windows\system32\winsock.dll
Size	2.80 KB (2,864 bytes)
Version	3.10
File	c:\windows\system32\wsock32.dll
Size	22.00 KB (22,528 bytes)
Version	5.2.3790.0 (srv03_rtm.030324-2048)

[Ports]

[Serial]

Item	Value
Name	Communications Port (COM1)
Status	OK
PNP Device ID	ACPI\PNP0501\1
Maximum Input Buffer Size	0
Maximum Output Buffer Size	No
Settable Baud Rate	Yes
Settable Data Bits	Yes
Settable Flow Control	Yes
Settable Parity	Yes
Settable Parity Check	Yes
Settable Stop Bits	Yes
Settable RLSD	Yes
Supports RLSD	Yes
Supports 16 Bit Mode	No
Supports Special Characters	No
Baud Rate	9600
Bits/Byte	8
Stop Bits	1
Parity	None
Busy	No
Abort Read/Write on Error	No
Binary Mode Enabled	Yes

Continue XMit on XOff No  
CTS Outflow Control No  
Discard NULL Bytes No  
DSR Outflow Control 0  
DSR Sensitivity 0  
DTR Flow Control Type Enable  
EOF Character 0  
Error Replace Character 0  
Error Replacement Enabled No  
Event Character 0  
Parity Check Enabled No  
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 Channel IRQ 3  
I/O Port 0x000002F8-0x000002FF  
Driver c:\windows\system32\drivers\serial.sys (5.2.3790.0  
(srv03\_rtm.030324-2048), 76.00 KB (77,824 bytes), 3/25/2003 7:00 AM)

[Parallel]

Item Value

[Storage]

[Drives]

Item Value

Drive A:  
Description 3 1/2 Inch Floppy Drive

Drive C:  
Description Local Fixed Disk  
Compressed No  
File System NTFS  
Size 33.90 GB (36,396,830,720 bytes)  
Free Space 26.95 GB (28,941,176,832 bytes)  
Volume Name  
Volume Serial Number 980D5B00

Drive D:  
Description CD-ROM Disc

Drive E:  
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

Drive F:  
Description Local Fixed Disk  
Compressed No  
File System NTFS  
Size 41.50 GB (44,564,533,248 bytes)  
Free Space 41.42 GB (44,470,390,784 bytes)  
Volume Name F\_drive  
Volume Serial Number 8071243A

Drive G:

Description Local Fixed Disk  
Compressed No  
File System NTFS  
Size 438.48 GB (470,814,994,432 bytes)  
Free Space 54.24 GB (58,238,713,856 bytes)  
Volume Name G\_Backup1  
Volume Serial Number 84DE7204

Drive H:  
Description Local Fixed Disk  
Compressed No  
File System NTFS  
Size 438.48 GB (470,814,994,432 bytes)  
Free Space 54.24 GB (58,238,713,856 bytes)  
Volume Name H\_backup2  
Volume Serial Number 08EBCCF1

Drive I:  
Description Local Fixed Disk  
Compressed No  
File System NTFS  
Size 438.48 GB (470,814,994,432 bytes)  
Free Space 54.24 GB (58,238,713,856 bytes)  
Volume Name I\_backup3  
Volume Serial Number 60F85D49

Drive J:  
Description Local Fixed Disk  
Compressed No  
File System NTFS  
Size 438.48 GB (470,814,994,432 bytes)  
Free Space 54.24 GB (58,238,844,928 bytes)  
Volume Name J\_backup4  
Volume Serial Number C0062820

Drive K:  
Description Local Fixed Disk  
Compressed No  
File System NTFS  
Size 438.48 GB (470,814,994,432 bytes)  
Free Space 54.24 GB (58,238,844,928 bytes)  
Volume Name K\_backup5  
Volume Serial Number ECD8CD54

Drive L:  
Description Local Fixed Disk  
Compressed No  
File System NTFS  
Size 438.48 GB (470,814,994,432 bytes)  
Free Space 54.24 GB (58,238,844,928 bytes)  
Volume Name L\_backup6  
Volume Serial Number F4E67E1B

Drive M:  
Description Local Fixed Disk  
Compressed No  
File System NTFS  
Size 401.37 GB (430,971,736,064 bytes)  
Free Space 210.50 GB (226,024,730,624 bytes)  
Volume Name M\_drive  
Volume Serial Number A4D73110

Drive N:  
Description Local Fixed Disk  
Compressed No  
File System NTFS  
Size 401.37 GB (430,971,736,064 bytes)  
Free Space 210.50 GB (226,024,730,624 bytes)

Volume Name N\_drive  
Volume Serial Number 3CE80562

Drive O:  
Description Local Fixed Disk  
Compressed No  
File System NTFS  
Size 401.37 GB (430,971,736,064 bytes)  
Free Space 210.50 GB (226,024,730,624 bytes)  
Volume Name O\_drive  
Volume Serial Number D8F68A2F

Drive P:  
Description Local Fixed Disk  
Compressed No  
File System NTFS  
Size 401.37 GB (430,971,736,064 bytes)  
Free Space 210.50 GB (226,024,796,160 bytes)  
Volume Name P\_drive  
Volume Serial Number B8FEE9B9

Drive Q:  
Description Local Fixed Disk  
Compressed No  
File System NTFS  
Size 401.37 GB (430,971,736,064 bytes)  
Free Space 210.50 GB (226,024,796,160 bytes)  
Volume Name Q\_drive  
Volume Serial Number 1010F14C

Drive R:  
Description Local Fixed Disk  
Compressed No  
File System NTFS  
Size 401.37 GB (430,971,736,064 bytes)  
Free Space 210.50 GB (226,024,796,160 bytes)  
Volume Name R\_drive  
Volume Serial Number D81F92F1

Drive S:  
Description Local Fixed Disk  
Compressed No  
File System NTFS  
Size 401.37 GB (430,971,736,064 bytes)  
Free Space 207.93 GB (223,262,892,032 bytes)  
Volume Name S\_drive  
Volume Serial Number 0C2E2624

Drive T:  
Description Local Fixed Disk  
Compressed No  
File System NTFS  
Size 400.88 GB (430,445,318,144 bytes)  
Free Space 400.81 GB (430,364,581,888 bytes)  
Volume Name T\_drive  
Volume Serial Number 4877D5B8

Drive U:  
Description Local Fixed Disk  
Compressed No  
File System NTFS  
Size 401.37 GB (430,971,736,064 bytes)  
Free Space 207.93 GB (223,262,912,512 bytes)  
Volume Name U\_drive  
Volume Serial Number 4043BC97

Drive V:  
Description Local Fixed Disk

Compressed No  
File System NTFS  
Size 401.37 GB (430,971,736,064 bytes)  
Free Space 207.93 GB (223,262,978,048 bytes)  
Volume Name V\_drive  
Volume Serial Number 4C4B72C8

Drive W:  
Description Local Fixed Disk  
Compressed No  
File System NTFS  
Size 401.37 GB (430,971,736,064 bytes)  
Free Space 207.93 GB (223,262,978,048 bytes)  
Volume Name W\_drive  
Volume Serial Number 905FE8DD

Drive X:  
Description Local Fixed Disk  
Compressed No  
File System NTFS  
Size 401.37 GB (430,971,736,064 bytes)  
Free Space 207.93 GB (223,262,978,048 bytes)  
Volume Name X\_drive  
Volume Serial Number 40682B8E

Drive Z:  
Description Network Connection  
Provider Name \\fsserv\hdrive

[Disks]

Item	Value
Description	Disk drive
Manufacturer	(Standard disk drives)
Model	IBM 1742 SCSI Disk Device
Bytes/Sector	512
Media Loaded	Yes
Media Type	Fixed hard disk
Partitions	3
SCSI Bus	0
SCSI Logical Unit	0
SCSI Port	9
SCSI Target ID	0
Sectors/Track	63
Size	473.52 GB (508,437,457,920 bytes)
Total Cylinders	61,814
Total Sectors	993,041,910
Total Tracks	15,762,570
Tracks/Cylinder	255
Partition Disk #13, Partition #0	
Partition Size	45.90 GB (49,285,845,504 bytes)
Partition Starting Offset	8,257,536 bytes
Partition Disk #13, Partition #1	
Partition Size	25.39 GB (27,266,770,944 bytes)
Partition Starting Offset	49,294,135,296 bytes
Partition Disk #13, Partition #2	
Partition Size	401.37 GB (430,971,738,624 bytes)
Partition Starting Offset	76,560,938,496 bytes

Description	Disk drive
Manufacturer	(Standard disk drives)
Model	IBM 1742 SCSI Disk Device
Bytes/Sector	512
Media Loaded	Yes
Media Type	Fixed hard disk
Partitions	3
SCSI Bus	0
SCSI Logical Unit	1

SCSI Port 9  
 SCSI Target ID 0  
 Sectors/Track 63  
 Size 473.52 GB (508,437,457,920 bytes)  
 Total Cylinders 61,814  
 Total Sectors 993,041,910  
 Total Tracks 15,762,570  
 Tracks/Cylinder 255  
 Partition Disk #14, Partition #0  
 Partition Size 45.90 GB (49,285,845,504 bytes)  
 Partition Starting Offset 8,257,536 bytes  
 Partition Disk #14, Partition #1  
 Partition Size 25.39 GB (27,266,770,944 bytes)  
 Partition Starting Offset 49,294,135,296 bytes  
 Partition Disk #14, Partition #2  
 Partition Size 401.37 GB (430,971,738,624 bytes)  
 Partition Starting Offset 76,560,938,496 bytes

Description Disk drive  
 Manufacturer (Standard disk drives)  
 Model IBM 1742 SCSI Disk Device  
 Bytes/Sector 512  
 Media Loaded Yes  
 Media Type Fixed hard disk  
 Partitions 3  
 SCSI Bus 0  
 SCSI Logical Unit 2  
 SCSI Port 9  
 SCSI Target ID 1  
 Sectors/Track 63  
 Size 473.52 GB (508,437,457,920 bytes)  
 Total Cylinders 61,814  
 Total Sectors 993,041,910  
 Total Tracks 15,762,570  
 Tracks/Cylinder 255  
 Partition Disk #15, Partition #0  
 Partition Size 45.90 GB (49,285,845,504 bytes)  
 Partition Starting Offset 8,257,536 bytes  
 Partition Disk #15, Partition #1  
 Partition Size 25.39 GB (27,266,770,944 bytes)  
 Partition Starting Offset 49,294,135,296 bytes  
 Partition Disk #15, Partition #2  
 Partition Size 401.37 GB (430,971,738,624 bytes)  
 Partition Starting Offset 76,560,938,496 bytes

Description Disk drive  
 Manufacturer (Standard disk drives)  
 Model IBM 1742 SCSI Disk Device  
 Bytes/Sector 512  
 Media Loaded Yes  
 Media Type Fixed hard disk  
 Partitions 3  
 SCSI Bus 0  
 SCSI Logical Unit 3  
 SCSI Port 9  
 SCSI Target ID 1  
 Sectors/Track 63  
 Size 473.52 GB (508,437,457,920 bytes)  
 Total Cylinders 61,814  
 Total Sectors 993,041,910  
 Total Tracks 15,762,570  
 Tracks/Cylinder 255  
 Partition Disk #16, Partition #0  
 Partition Size 45.90 GB (49,285,845,504 bytes)  
 Partition Starting Offset 8,257,536 bytes  
 Partition Disk #16, Partition #1  
 Partition Size 25.39 GB (27,266,770,944 bytes)  
 Partition Starting Offset 49,294,135,296 bytes

Partition Disk #16, Partition #2  
 Partition Size 401.37 GB (430,971,738,624 bytes)  
 Partition Starting Offset 76,560,938,496 bytes

Description Disk drive  
 Manufacturer (Standard disk drives)  
 Model IBM 1742 SCSI Disk Device  
 Bytes/Sector 512  
 Media Loaded Yes  
 Media Type Fixed hard disk  
 Partitions 3  
 SCSI Bus 0  
 SCSI Logical Unit 0  
 SCSI Port 8  
 SCSI Target ID 0  
 Sectors/Track 63  
 Size 473.52 GB (508,437,457,920 bytes)  
 Total Cylinders 61,814  
 Total Sectors 993,041,910  
 Total Tracks 15,762,570  
 Tracks/Cylinder 255  
 Partition Disk #9, Partition #0  
 Partition Size 45.90 GB (49,285,845,504 bytes)  
 Partition Starting Offset 8,257,536 bytes  
 Partition Disk #9, Partition #1  
 Partition Size 25.39 GB (27,266,770,944 bytes)  
 Partition Starting Offset 49,294,135,296 bytes  
 Partition Disk #9, Partition #2  
 Partition Size 401.37 GB (430,971,738,624 bytes)  
 Partition Starting Offset 76,560,938,496 bytes

Description Disk drive  
 Manufacturer (Standard disk drives)  
 Model IBM 1742 SCSI Disk Device  
 Bytes/Sector 512  
 Media Loaded Yes  
 Media Type Fixed hard disk  
 Partitions 3  
 SCSI Bus 0  
 SCSI Logical Unit 1  
 SCSI Port 8  
 SCSI Target ID 0  
 Sectors/Track 63  
 Size 473.52 GB (508,437,457,920 bytes)  
 Total Cylinders 61,814  
 Total Sectors 993,041,910  
 Total Tracks 15,762,570  
 Tracks/Cylinder 255  
 Partition Disk #10, Partition #0  
 Partition Size 45.90 GB (49,285,845,504 bytes)  
 Partition Starting Offset 8,257,536 bytes  
 Partition Disk #10, Partition #1  
 Partition Size 25.39 GB (27,266,770,944 bytes)  
 Partition Starting Offset 49,294,135,296 bytes  
 Partition Disk #10, Partition #2  
 Partition Size 401.37 GB (430,971,738,624 bytes)  
 Partition Starting Offset 76,560,938,496 bytes

Description Disk drive  
 Manufacturer (Standard disk drives)  
 Model IBM 1742 SCSI Disk Device  
 Bytes/Sector 512  
 Media Loaded Yes  
 Media Type Fixed hard disk  
 Partitions 3  
 SCSI Bus 0  
 SCSI Logical Unit 2  
 SCSI Port 8

SCSI Target ID 1  
 Sectors/Track 63  
 Size 473.52 GB (508,437,457,920 bytes)  
 Total Cylinders 61,814  
 Total Sectors 993,041,910  
 Total Tracks 15,762,570  
 Tracks/Cylinder 255  
 Partition Disk #11, Partition #0  
 Partition Size 45.90 GB (49,285,845,504 bytes)  
 Partition Starting Offset 8,257,536 bytes  
 Partition Disk #11, Partition #1  
 Partition Size 25.39 GB (27,266,770,944 bytes)  
 Partition Starting Offset 49,294,135,296 bytes  
 Partition Disk #11, Partition #2  
 Partition Size 401.37 GB (430,971,738,624 bytes)  
 Partition Starting Offset 76,560,938,496 bytes

Description Disk drive  
 Manufacturer (Standard disk drives)  
 Model IBM 1742 SCSI Disk Device  
 Bytes/Sector 512  
 Media Loaded Yes  
 Media Type Fixed hard disk  
 Partitions 3  
 SCSI Bus 0  
 SCSI Logical Unit 3  
 SCSI Port 8

SCSI Target ID 1  
 Sectors/Track 63  
 Size 473.52 GB (508,437,457,920 bytes)  
 Total Cylinders 61,814  
 Total Sectors 993,041,910  
 Total Tracks 15,762,570  
 Tracks/Cylinder 255  
 Partition Disk #12, Partition #0  
 Partition Size 45.90 GB (49,285,845,504 bytes)  
 Partition Starting Offset 8,257,536 bytes  
 Partition Disk #12, Partition #1  
 Partition Size 25.39 GB (27,266,770,944 bytes)  
 Partition Starting Offset 49,294,135,296 bytes  
 Partition Disk #12, Partition #2  
 Partition Size 401.37 GB (430,971,738,624 bytes)  
 Partition Starting Offset 76,560,938,496 bytes

Description Disk drive  
 Manufacturer (Standard disk drives)  
 Model IBM 1742 SCSI Disk Device  
 Bytes/Sector 512  
 Media Loaded Yes  
 Media Type Fixed hard disk  
 Partitions 3  
 SCSI Bus 0  
 SCSI Logical Unit 0  
 SCSI Port 4

SCSI Target ID 0  
 Sectors/Track 63  
 Size 473.52 GB (508,437,457,920 bytes)  
 Total Cylinders 61,814  
 Total Sectors 993,041,910  
 Total Tracks 15,762,570  
 Tracks/Cylinder 255  
 Partition Disk #5, Partition #0  
 Partition Size 45.90 GB (49,285,845,504 bytes)  
 Partition Starting Offset 8,257,536 bytes  
 Partition Disk #5, Partition #1  
 Partition Size 25.39 GB (27,266,770,944 bytes)  
 Partition Starting Offset 49,294,135,296 bytes  
 Partition Disk #5, Partition #2

Partition Size 401.37 GB (430,971,738,624 bytes)  
 Partition Starting Offset 76,560,938,496 bytes

Description Disk drive  
 Manufacturer (Standard disk drives)  
 Model IBM 1742 SCSI Disk Device  
 Bytes/Sector 512  
 Media Loaded Yes  
 Media Type Fixed hard disk  
 Partitions 3  
 SCSI Bus 0

SCSI Logical Unit 1  
 SCSI Port 4  
 SCSI Target ID 0  
 Sectors/Track 63  
 Size 473.52 GB (508,437,457,920 bytes)  
 Total Cylinders 61,814  
 Total Sectors 993,041,910  
 Total Tracks 15,762,570  
 Tracks/Cylinder 255  
 Partition Disk #6, Partition #0  
 Partition Size 45.90 GB (49,285,845,504 bytes)  
 Partition Starting Offset 8,257,536 bytes  
 Partition Disk #6, Partition #1  
 Partition Size 25.39 GB (27,266,770,944 bytes)  
 Partition Starting Offset 49,294,135,296 bytes  
 Partition Disk #6, Partition #2  
 Partition Size 401.37 GB (430,971,738,624 bytes)  
 Partition Starting Offset 76,560,938,496 bytes

Description Disk drive  
 Manufacturer (Standard disk drives)  
 Model IBM 1742 SCSI Disk Device  
 Bytes/Sector 512  
 Media Loaded Yes  
 Media Type Fixed hard disk  
 Partitions 3  
 SCSI Bus 0

SCSI Logical Unit 2  
 SCSI Port 4  
 SCSI Target ID 1  
 Sectors/Track 63  
 Size 473.52 GB (508,437,457,920 bytes)  
 Total Cylinders 61,814  
 Total Sectors 993,041,910  
 Total Tracks 15,762,570  
 Tracks/Cylinder 255  
 Partition Disk #7, Partition #0  
 Partition Size 45.90 GB (49,285,845,504 bytes)  
 Partition Starting Offset 8,257,536 bytes  
 Partition Disk #7, Partition #1  
 Partition Size 25.39 GB (27,266,770,944 bytes)  
 Partition Starting Offset 49,294,135,296 bytes  
 Partition Disk #7, Partition #2  
 Partition Size 401.37 GB (430,971,738,624 bytes)  
 Partition Starting Offset 76,560,938,496 bytes

Description Disk drive  
 Manufacturer (Standard disk drives)  
 Model IBM 1742 SCSI Disk Device  
 Bytes/Sector 512  
 Media Loaded Yes  
 Media Type Fixed hard disk  
 Partitions 3  
 SCSI Bus 0

SCSI Logical Unit 3  
 SCSI Port 4  
 SCSI Target ID 1

Sectors/Track 63  
 Size 473.52 GB (508,437,457,920 bytes)  
 Total Cylinders 61,814  
 Total Sectors 993,041,910  
 Total Tracks 15,762,570  
 Tracks/Cylinder 255  
 Partition Disk #8, Partition #0  
 Partition Size 45.90 GB (49,285,845,504 bytes)  
 Partition Starting Offset 8,257,536 bytes  
 Partition Disk #8, Partition #1  
 Partition Size 25.39 GB (27,266,770,944 bytes)  
 Partition Starting Offset 49,294,135,296 bytes  
 Partition Disk #8, Partition #2  
 Partition Size 401.37 GB (430,971,738,624 bytes)  
 Partition Starting Offset 76,560,938,496 bytes

Description Disk drive  
 Manufacturer (Standard disk drives)  
 Model IBM 1742 SCSI Disk Device  
 Bytes/Sector 512  
 Media Loaded Yes  
 Media Type Fixed hard disk  
 Partitions 3  
 SCSI Bus 0  
 SCSI Logical Unit 0  
 SCSI Port 3  
 SCSI Target ID 0  
 Sectors/Track 63  
 Size 473.52 GB (508,437,457,920 bytes)

Total Cylinders 61,814  
 Total Sectors 993,041,910  
 Total Tracks 15,762,570  
 Tracks/Cylinder 255  
 Partition Disk #1, Partition #0  
 Partition Size 45.90 GB (49,285,845,504 bytes)  
 Partition Starting Offset 8,257,536 bytes  
 Partition Disk #1, Partition #1  
 Partition Size 25.39 GB (27,266,770,944 bytes)  
 Partition Starting Offset 49,294,135,296 bytes  
 Partition Disk #1, Partition #2  
 Partition Size 401.37 GB (430,971,738,624 bytes)  
 Partition Starting Offset 76,560,938,496 bytes

Description Disk drive  
 Manufacturer (Standard disk drives)  
 Model IBM 1742 SCSI Disk Device  
 Bytes/Sector 512  
 Media Loaded Yes  
 Media Type Fixed hard disk  
 Partitions 3  
 SCSI Bus 0  
 SCSI Logical Unit 1  
 SCSI Port 3  
 SCSI Target ID 0  
 Sectors/Track 63  
 Size 473.52 GB (508,437,457,920 bytes)

Total Cylinders 61,814  
 Total Sectors 993,041,910  
 Total Tracks 15,762,570  
 Tracks/Cylinder 255  
 Partition Disk #2, Partition #0  
 Partition Size 45.90 GB (49,285,845,504 bytes)  
 Partition Starting Offset 8,257,536 bytes  
 Partition Disk #2, Partition #1  
 Partition Size 25.39 GB (27,266,770,944 bytes)  
 Partition Starting Offset 49,294,135,296 bytes  
 Partition Disk #2, Partition #2  
 Partition Size 401.37 GB (430,971,738,624 bytes)

Partition Starting Offset 76,560,938,496 bytes

Description Disk drive  
 Manufacturer (Standard disk drives)  
 Model IBM 1742 SCSI Disk Device  
 Bytes/Sector 512  
 Media Loaded Yes  
 Media Type Fixed hard disk  
 Partitions 3  
 SCSI Bus 0  
 SCSI Logical Unit 2  
 SCSI Port 3  
 SCSI Target ID 1  
 Sectors/Track 63  
 Size 473.52 GB (508,437,457,920 bytes)  
 Total Cylinders 61,814  
 Total Sectors 993,041,910  
 Total Tracks 15,762,570  
 Tracks/Cylinder 255  
 Partition Disk #3, Partition #0  
 Partition Size 45.90 GB (49,285,845,504 bytes)  
 Partition Starting Offset 8,257,536 bytes  
 Partition Disk #3, Partition #1  
 Partition Size 25.39 GB (27,266,770,944 bytes)  
 Partition Starting Offset 49,294,135,296 bytes  
 Partition Disk #3, Partition #2  
 Partition Size 401.37 GB (430,971,738,624 bytes)  
 Partition Starting Offset 76,560,938,496 bytes

Description Disk drive  
 Manufacturer (Standard disk drives)  
 Model IBM 1742 SCSI Disk Device  
 Bytes/Sector 512  
 Media Loaded Yes  
 Media Type Fixed hard disk  
 Partitions 3  
 SCSI Bus 0  
 SCSI Logical Unit 3  
 SCSI Port 3  
 SCSI Target ID 1  
 Sectors/Track 63  
 Size 473.52 GB (508,437,457,920 bytes)

Total Cylinders 61,814  
 Total Sectors 993,041,910  
 Total Tracks 15,762,570  
 Tracks/Cylinder 255  
 Partition Disk #4, Partition #0  
 Partition Size 45.90 GB (49,285,845,504 bytes)  
 Partition Starting Offset 8,257,536 bytes  
 Partition Disk #4, Partition #1  
 Partition Size 25.39 GB (27,266,770,944 bytes)  
 Partition Starting Offset 49,294,135,296 bytes  
 Partition Disk #4, Partition #2  
 Partition Size 400.88 GB (430,445,320,704 bytes)  
 Partition Starting Offset 76,560,938,496 bytes

Description Disk drive  
 Manufacturer (Standard disk drives)  
 Model IBM 1742 SCSI Disk Device  
 Bytes/Sector 512  
 Media Loaded Yes  
 Media Type Fixed hard disk  
 Partitions 3  
 SCSI Bus 0  
 SCSI Logical Unit 0  
 SCSI Port 11  
 SCSI Target ID 0  
 Sectors/Track 63

Size 473.52 GB (508,437,457,920 bytes)  
 Total Cylinders 61,814  
 Total Sectors 993,041,910  
 Total Tracks 15,762,570  
 Tracks/Cylinder 255  
 Partition Disk #17, Partition #0  
 Partition Size 45.90 GB (49,285,845,504 bytes)  
 Partition Starting Offset 8,257,536 bytes  
 Partition Disk #17, Partition #1  
 Partition Size 25.39 GB (27,266,770,944 bytes)  
 Partition Starting Offset 49,294,135,296 bytes  
 Partition Disk #17, Partition #2  
 Partition Size 401.37 GB (430,971,738,624 bytes)  
 Partition Starting Offset 76,560,938,496 bytes

Description Disk drive  
 Manufacturer (Standard disk drives)  
 Model IBM 1742 SCSI Disk Device  
 Bytes/Sector 512  
 Media Loaded Yes  
 Media Type Fixed hard disk  
 Partitions 3  
 SCSI Bus 0  
 SCSI Logical Unit 1  
 SCSI Port 11  
 SCSI Target ID 0  
 Sectors/Track 63

Size 473.52 GB (508,437,457,920 bytes)  
 Total Cylinders 61,814  
 Total Sectors 993,041,910  
 Total Tracks 15,762,570  
 Tracks/Cylinder 255  
 Partition Disk #18, Partition #0  
 Partition Size 45.90 GB (49,285,845,504 bytes)  
 Partition Starting Offset 8,257,536 bytes  
 Partition Disk #18, Partition #1  
 Partition Size 25.39 GB (27,266,770,944 bytes)  
 Partition Starting Offset 49,294,135,296 bytes  
 Partition Disk #18, Partition #2  
 Partition Size 401.37 GB (430,971,738,624 bytes)  
 Partition Starting Offset 76,560,938,496 bytes

Description Disk drive  
 Manufacturer (Standard disk drives)  
 Model IBM 1742 SCSI Disk Device  
 Bytes/Sector 512  
 Media Loaded Yes  
 Media Type Fixed hard disk  
 Partitions 3  
 SCSI Bus 0  
 SCSI Logical Unit 2  
 SCSI Port 11  
 SCSI Target ID 1  
 Sectors/Track 63

Size 473.52 GB (508,437,457,920 bytes)  
 Total Cylinders 61,814  
 Total Sectors 993,041,910  
 Total Tracks 15,762,570  
 Tracks/Cylinder 255  
 Partition Disk #19, Partition #0  
 Partition Size 45.90 GB (49,285,845,504 bytes)  
 Partition Starting Offset 8,257,536 bytes  
 Partition Disk #19, Partition #1  
 Partition Size 25.39 GB (27,266,770,944 bytes)  
 Partition Starting Offset 49,294,135,296 bytes  
 Partition Disk #19, Partition #2  
 Partition Size 401.37 GB (430,971,738,624 bytes)  
 Partition Starting Offset 76,560,938,496 bytes

Description Disk drive  
 Manufacturer (Standard disk drives)  
 Model IBM 1742 SCSI Disk Device  
 Bytes/Sector 512  
 Media Loaded Yes  
 Media Type Fixed hard disk  
 Partitions 3  
 SCSI Bus 0  
 SCSI Logical Unit 3  
 SCSI Port 11  
 SCSI Target ID 1  
 Sectors/Track 63

Size 948.14 GB (1,018,059,356,160 bytes)  
 Total Cylinders 123,772  
 Total Sectors 1,988,397,180  
 Total Tracks 31,561,860  
 Tracks/Cylinder 255  
 Partition Disk #20, Partition #0  
 Partition Size 45.90 GB (49,285,845,504 bytes)  
 Partition Starting Offset 8,257,536 bytes  
 Partition Disk #20, Partition #1  
 Partition Size 25.39 GB (27,266,770,944 bytes)  
 Partition Starting Offset 49,294,135,296 bytes  
 Partition Disk #20, Partition #2  
 Partition Size 875.98 GB (940,577,186,304 bytes)  
 Partition Starting Offset 76,560,938,496 bytes

Description Disk drive  
 Manufacturer (Standard disk drives)  
 Model IBM 1742 SCSI Disk Device  
 Bytes/Sector 512  
 Media Loaded Yes  
 Media Type Fixed hard disk  
 Partitions 1  
 SCSI Bus 0  
 SCSI Logical Unit 1  
 SCSI Port 12  
 SCSI Target ID 1  
 Sectors/Track 63

Size 440.20 GB (472,665,715,200 bytes)  
 Total Cylinders 57,465  
 Total Sectors 923,175,225  
 Total Tracks 14,653,575  
 Tracks/Cylinder 255  
 Partition Disk #21, Partition #0  
 Partition Size 438.48 GB (470,814,994,944 bytes)  
 Partition Starting Offset 8,257,536 bytes

Description Disk drive  
 Manufacturer (Standard disk drives)  
 Model IBM 1742 SCSI Disk Device  
 Bytes/Sector 512  
 Media Loaded Yes  
 Media Type Fixed hard disk  
 Partitions 1  
 SCSI Bus 0  
 SCSI Logical Unit 2  
 SCSI Port 12  
 SCSI Target ID 1  
 Sectors/Track 63

Size 440.20 GB (472,665,715,200 bytes)  
 Total Cylinders 57,465  
 Total Sectors 923,175,225  
 Total Tracks 14,653,575  
 Tracks/Cylinder 255  
 Partition Disk #22, Partition #0  
 Partition Size 438.48 GB (470,814,994,944 bytes)



Partition Starting Offset 8,257,536 bytes

Description Disk drive  
 Manufacturer (Standard disk drives)  
 Model IBM 1742 SCSI Disk Device  
 Bytes/Sector 512  
 Media Loaded Yes  
 Media Type Fixed hard disk  
 Partitions 1  
 SCSI Bus 0  
 SCSI Logical Unit 3  
 SCSI Port 12  
 SCSI Target ID 1  
 Sectors/Track 63  
 Size 440.20 GB (472,665,715,200 bytes)  
 Total Cylinders 57,465  
 Total Sectors 923,175,225  
 Total Tracks 14,653,575  
 Tracks/Cylinder 255  
 Partition Disk #23, Partition #0  
 Partition Size 438.48 GB (470,814,994,944 bytes)  
 Partition Starting Offset 8,257,536 bytes

Description Disk drive  
 Manufacturer (Standard disk drives)  
 Model IBM 1742 SCSI Disk Device  
 Bytes/Sector 512  
 Media Loaded Yes  
 Media Type Fixed hard disk  
 Partitions 1  
 SCSI Bus 0  
 SCSI Logical Unit 4  
 SCSI Port 12  
 SCSI Target ID 1  
 Sectors/Track 63  
 Size 440.20 GB (472,665,715,200 bytes)  
 Total Cylinders 57,465  
 Total Sectors 923,175,225  
 Total Tracks 14,653,575  
 Tracks/Cylinder 255  
 Partition Disk #24, Partition #0  
 Partition Size 438.48 GB (470,814,994,944 bytes)  
 Partition Starting Offset 8,257,536 bytes

Description Disk drive  
 Manufacturer (Standard disk drives)  
 Model IBM 1742 SCSI Disk Device  
 Bytes/Sector 512  
 Media Loaded Yes  
 Media Type Fixed hard disk  
 Partitions 1  
 SCSI Bus 0  
 SCSI Logical Unit 5  
 SCSI Port 12  
 SCSI Target ID 1  
 Sectors/Track 63  
 Size 440.20 GB (472,665,715,200 bytes)  
 Total Cylinders 57,465  
 Total Sectors 923,175,225  
 Total Tracks 14,653,575  
 Tracks/Cylinder 255  
 Partition Disk #25, Partition #0  
 Partition Size 438.48 GB (470,814,994,944 bytes)  
 Partition Starting Offset 8,257,536 bytes

Description Disk drive  
 Manufacturer (Standard disk drives)  
 Model IBM 1742 SCSI Disk Device

Bytes/Sector 512  
 Media Loaded Yes  
 Media Type Fixed hard disk  
 Partitions 1  
 SCSI Bus 0  
 SCSI Logical Unit 6  
 SCSI Port 12  
 SCSI Target ID 1  
 Sectors/Track 63  
 Size 440.20 GB (472,665,715,200 bytes)  
 Total Cylinders 57,465  
 Total Sectors 923,175,225  
 Total Tracks 14,653,575  
 Tracks/Cylinder 255  
 Partition Disk #26, Partition #0  
 Partition Size 438.48 GB (470,814,994,944 bytes)  
 Partition Starting Offset 8,257,536 bytes

Description Disk drive  
 Manufacturer (Standard disk drives)  
 Model IBM 1742 SCSI Disk Device  
 Bytes/Sector 512  
 Media Loaded Yes  
 Media Type Fixed hard disk  
 Partitions 2  
 SCSI Bus 0  
 SCSI Logical Unit 0  
 SCSI Port 2  
 SCSI Target ID 0  
 Sectors/Track 63  
 Size 474.07 GB (509,029,678,080 bytes)  
 Total Cylinders 61,886  
 Total Sectors 994,198,590  
 Total Tracks 15,780,930  
 Tracks/Cylinder 255  
 Partition Disk #0, Partition #0  
 Partition Size 431.65 GB (463,478,045,184 bytes)  
 Partition Starting Offset 8,257,536 bytes  
 Partition Disk #0, Partition #1  
 Partition Size 41.50 GB (44,564,534,784 bytes)  
 Partition Starting Offset 463,486,334,976 bytes

Description Disk drive  
 Manufacturer (Standard disk drives)  
 Model IBM-ESXS DTN036C3UCDY10FN SCSI Disk Device  
 Bytes/Sector 512  
 Media Loaded Yes  
 Media Type Fixed hard disk  
 Partitions 1  
 SCSI Bus 0  
 SCSI Logical Unit 0  
 SCSI Port 13  
 SCSI Target ID 0  
 Sectors/Track 63  
 Size 33.90 GB (36,396,864,000 bytes)  
 Total Cylinders 4,425  
 Total Sectors 71,087,625  
 Total Tracks 1,128,375  
 Tracks/Cylinder 255  
 Partition Disk #27, Partition #0  
 Partition Size 33.90 GB (36,396,831,744 bytes)  
 Partition Starting Offset 32,256 bytes

[SCSI]

Item	Value
Name	LSI Logic PCI-X Ultra320 SCSI Host Adapter
Manufacturer	LSI Logic Inc.

Status OK  
PNP Device ID  
PCI\VEN\_1000&DEV\_0030&SUBSYS\_02921014&REV\_07\3&13C0B0C5&0&18  
I/O Port 0x00002000-0x000027FF  
Memory Address 0xF7600000-0xF76FFFFF  
Memory Address 0xF7610000-0xF761FFFF  
IRQ Channel IRQ 40  
Driver c:\windows\system32\drivers\symmpi.sys (1.08.18.00 (NT.021001-2000), 25.88 KB (26,496 bytes), 3/25/2003 7:00 AM)

Name LSI Logic PCI-X Ultra320 SCSI Host Adapter  
Manufacturer LSI Logic Inc.  
Status OK  
PNP Device ID  
PCI\VEN\_1000&DEV\_0030&SUBSYS\_02921014&REV\_07\3&13C0B0C5&0&19  
I/O Port 0x00002100-0x000021FF  
Memory Address 0xF7620000-0xF762FFFF  
Memory Address 0xF7630000-0xF763FFFF  
IRQ Channel IRQ 41  
Driver c:\windows\system32\drivers\symmpi.sys (1.08.18.00 (NT.021001-2000), 25.88 KB (26,496 bytes), 3/25/2003 7:00 AM)

Name QLogic QLA23xx PCI Fibre Channel Adapter  
Manufacturer QLogic  
Status OK  
PNP Device ID  
PCI\VEN\_1077&DEV\_2312&SUBSYS\_01001077&REV\_02\3&1070020&0&08  
I/O Port 0x00005000-0x000050FF  
Memory Address 0xF6420000-0xF6420FFF  
IRQ Channel IRQ 51  
Driver c:\windows\system32\drivers\ql2300.sys (8.2.0 Beta 10 (W2K VI), 431.38 KB (441,733 bytes), 6/4/2003 5:44 PM)

Name QLogic QLA23xx PCI Fibre Channel Adapter  
Manufacturer QLogic  
Status OK  
PNP Device ID  
PCI\VEN\_1077&DEV\_2312&SUBSYS\_01001077&REV\_02\3&172E68DD&0&18  
I/O Port 0x00002800-0x000029FF  
Memory Address 0xF7820000-0xF7820FFF  
IRQ Channel IRQ 67  
Driver c:\windows\system32\drivers\ql2300.sys (8.2.0 Beta 10 (W2K VI), 431.38 KB (441,733 bytes), 6/4/2003 5:44 PM)

Name QLogic QLA23xx PCI Fibre Channel Adapter  
Manufacturer QLogic  
Status OK  
PNP Device ID  
PCI\VEN\_1077&DEV\_2312&SUBSYS\_010C1077&REV\_02\3&474B838&0&08  
I/O Port 0x00007000-0x000070FF  
Memory Address 0xF6C20000-0xF6C20FFF  
IRQ Channel IRQ 59  
Driver c:\windows\system32\drivers\ql2300.sys (8.2.0 Beta 10 (W2K VI), 431.38 KB (441,733 bytes), 6/4/2003 5:44 PM)

Name LSI Logic PCI-X Ultra320 SCSI Host Adapter  
Manufacturer LSI Logic Inc.  
Status OK  
PNP Device ID  
PCI\VEN\_1000&DEV\_0030&SUBSYS\_02921014&REV\_07\3&1D521019&0&18  
I/O Port 0x0000C200-0x0000C3FF  
Memory Address 0xE9000000-0xE900FFFF

Memory Address 0xE9010000-0xE901FFFF  
IRQ Channel IRQ 193  
Driver c:\windows\system32\drivers\symmpi.sys (1.08.18.00 (NT.021001-2000), 25.88 KB (26,496 bytes), 3/25/2003 7:00 AM)

Name LSI Logic PCI-X Ultra320 SCSI Host Adapter  
Manufacturer LSI Logic Inc.  
Status OK  
PNP Device ID  
PCI\VEN\_1000&DEV\_0030&SUBSYS\_02921014&REV\_07\3&1D521019&0&19  
I/O Port 0x0000C300-0x0000C3FF  
Memory Address 0xE9020000-0xE902FFFF  
Memory Address 0xE9030000-0xE903FFFF  
IRQ Channel IRQ 194  
Driver c:\windows\system32\drivers\symmpi.sys (1.08.18.00 (NT.021001-2000), 25.88 KB (26,496 bytes), 3/25/2003 7:00 AM)

Name QLogic QLA23xx PCI Fibre Channel Adapter  
Manufacturer QLogic  
Status OK  
PNP Device ID  
PCI\VEN\_1077&DEV\_2312&SUBSYS\_01001077&REV\_02\3&300BC0BE&0&08  
I/O Port 0x0000C400-0x0000C5FF  
Memory Address 0xE9120000-0xE9120FFF  
IRQ Channel IRQ 173  
Driver c:\windows\system32\drivers\ql2300.sys (8.2.0 Beta 10 (W2K VI), 431.38 KB (441,733 bytes), 6/4/2003 5:44 PM)

Name QLogic QLA23xx PCI Fibre Channel Adapter  
Manufacturer QLogic  
Status OK  
PNP Device ID  
PCI\VEN\_1077&DEV\_2312&SUBSYS\_010C1077&REV\_02\3&300BC0BE&0&10  
I/O Port 0x0000C500-0x0000C5FF  
Memory Address 0xE9121000-0xE9121FFF  
IRQ Channel IRQ 174  
Driver c:\windows\system32\drivers\ql2300.sys (8.2.0 Beta 10 (W2K VI), 431.38 KB (441,733 bytes), 6/4/2003 5:44 PM)

Name QLogic QLA23xx PCI Fibre Channel Adapter  
Manufacturer QLogic  
Status OK  
PNP Device ID  
PCI\VEN\_1077&DEV\_2312&SUBSYS\_01001077&REV\_02\3&72AA75C&0&20  
I/O Port 0x0000C600-0x0000C7FF  
Memory Address 0xE9320000-0xE9320FFF  
IRQ Channel IRQ 175  
Driver c:\windows\system32\drivers\ql2300.sys (8.2.0 Beta 10 (W2K VI), 431.38 KB (441,733 bytes), 6/4/2003 5:44 PM)

Name QLogic QLA23xx PCI Fibre Channel Adapter  
Manufacturer QLogic  
Status OK  
PNP Device ID  
PCI\VEN\_1077&DEV\_2312&SUBSYS\_01001077&REV\_02\3&2C9E08A6&0&08  
I/O Port 0x0000CE00-0x0000CEFF  
Memory Address 0xE8420000-0xE8420FFF  
IRQ Channel IRQ 177  
Driver c:\windows\system32\drivers\ql2300.sys (8.2.0 Beta 10 (W2K VI), 431.38 KB (441,733 bytes), 6/4/2003 5:44 PM)

Name QLogic QLA23xx PCI Fibre Channel Adapter  
Manufacturer QLogic

Status OK  
 PNP Device ID  
 PCI\VEN\_1077&DEV\_2312&SUBSYS\_01001077&REV\_02\3&F2FC708&&08  
 I/O Port 0x00008600-0x000086FF  
 Memory Address 0xF0C20000-0xF0C20FFF  
 IRQ Channel IRQ 109  
 Driver c:\windows\system32\drivers\ql2300.sys (8.2.0 Beta 10 (W2K VI), 431.38 KB (441,733 bytes), 6/4/2003 5:44 PM)

Name QLogic QLA23xx PCI Fibre Channel Adapter  
 Manufacturer QLogic  
 Status OK  
 PNP Device ID  
 PCI\VEN\_1077&DEV\_2312&SUBSYS\_010C1077&REV\_02\3&21E977AD&&0&08  
 I/O Port 0x0000B000-0x0000B0FF  
 Memory Address 0xF1420000-0xF1420FFF  
 IRQ Channel IRQ 102  
 Driver c:\windows\system32\drivers\ql2300.sys (8.2.0 Beta 10 (W2K VI), 431.38 KB (441,733 bytes), 6/4/2003 5:44 PM)

Name QLogic QLA23xx PCI Fibre Channel Adapter  
 Manufacturer QLogic  
 Status OK  
 PNP Device ID  
 PCI\VEN\_1077&DEV\_2312&SUBSYS\_010C1077&REV\_02\3&3ACAD158&&0&08  
 I/O Port 0x0000D400-0x0000D5FF  
 Memory Address 0xED220000-0xED220FFF  
 IRQ Channel IRQ 123  
 Driver c:\windows\system32\drivers\ql2300.sys (8.2.0 Beta 10 (W2K VI), 431.38 KB (441,733 bytes), 6/4/2003 5:44 PM)

Name QLogic QLA23xx PCI Fibre Channel Adapter  
 Manufacturer QLogic  
 Status OK  
 PNP Device ID  
 PCI\VEN\_1077&DEV\_2312&SUBSYS\_01001077&REV\_02\3&3ACAD158&&0&10  
 I/O Port 0x0000D500-0x0000D5FF  
 Memory Address 0xED221000-0xED221FFF  
 IRQ Channel IRQ 127  
 Driver c:\windows\system32\drivers\ql2300.sys (8.2.0 Beta 10 (W2K VI), 431.38 KB (441,733 bytes), 6/4/2003 5:44 PM)

[IDE]

Item Value  
 Name VIA Bus Master IDE Controller  
 Manufacturer VIA Technologies, Inc.  
 Status OK  
 PNP Device ID  
 PCI\VEN\_1106&DEV\_0571&SUBSYS\_02A61014&REV\_06\3&267A616A&&0&29  
 I/O Port 0x00000700-0x0000070F  
 Driver c:\windows\system32\drivers\vialide.sys (1.00.01.00, 7.00 KB (7,168 bytes), 3/25/2003 7:00 AM)

Name Primary IDE Channel  
 Manufacturer (Standard IDE ATA/ATAPI controllers)  
 Status OK  
 PNP Device ID PCI\IDE\IDECHANNEL\4&29582549&0&0  
 I/O Port 0x000001F0-0x000001F7  
 I/O Port 0x000003F6-0x000003F6  
 IRQ Channel IRQ 14  
 Driver c:\windows\system32\drivers\atapi.sys (5.2.3790.0 (srv03\_rtm.030324-2048), 89.00 KB (91,136 bytes), 3/25/2003 7:00 AM)

Name Secondary IDE Channel  
 Manufacturer (Standard IDE ATA/ATAPI controllers)  
 Status OK  
 PNP Device ID PCI\IDE\IDECHANNEL\4&29582549&0&1  
 I/O Port 0x00000170-0x00000177  
 I/O Port 0x00000376-0x00000376  
 IRQ Channel IRQ 15  
 Driver c:\windows\system32\drivers\atapi.sys (5.2.3790.0 (srv03\_rtm.030324-2048), 89.00 KB (91,136 bytes), 3/25/2003 7:00 AM)

[Printing]

Name	Driver	Port Name	Server Name
------	--------	-----------	-------------

[Problem Devices]

Device	PNP Device ID	Error Code
Not Available	ACPI\IBM37D4\2&DABA3FF&0	The drivers for this device are not installed.
Other PCI Bridge Device		
PCI\VEN_1014&DEV_010F&SUBSYS_01131014&REV_00\3&267A616A&&0&20		The drivers for this device are not installed.
VIA Rev 5 or later USB Universal Host Controller		
PCI\VEN_1106&DEV_3038&SUBSYS_02A61014&REV_16\3&267A616A&&0&2A		This device is disabled.
VIA Rev 5 or later USB Universal Host Controller		
PCI\VEN_1106&DEV_3038&SUBSYS_02A61014&REV_16\3&267A616A&&0&2B		This device is disabled.
Broadcom NetXtreme Gigabit Ethernet		
PCI\VEN_14E4&DEV_1648&SUBSYS_02A61014&REV_02\3&13C0B0C5&&0&20		This device is disabled.
Broadcom NetXtreme Gigabit Ethernet #2		
PCI\VEN_14E4&DEV_1648&SUBSYS_02A61014&REV_02\3&13C0B0C5&&0&21		This device is disabled.
Other PCI Bridge Device		
PCI\VEN_1014&DEV_010F&SUBSYS_01131014&REV_00\3&A985F74&&0&20		The drivers for this device are not installed.
Broadcom NetXtreme Gigabit Ethernet #3		
PCI\VEN_14E4&DEV_1648&SUBSYS_02A61014&REV_02\3&1D521019&&0&20		This device is disabled.
Broadcom NetXtreme Gigabit Ethernet #4		
PCI\VEN_14E4&DEV_1648&SUBSYS_02A61014&REV_02\3&1D521019&&0&21		This device is disabled.

[USB]

Device	PNP Device ID
VIA Rev 5 or later USB Universal Host Controller	PCI\VEN_1106&DEV_3038&SUBSYS_02A61014&REV_16\3&267A616A&&0&2A
VIA Rev 5 or later USB Universal Host Controller	PCI\VEN_1106&DEV_3038&SUBSYS_02A61014&REV_16\3&267A616A&&0&2B

[Software Environment]

[System Drivers]

Name	Description	File	Type	Started	Start Mode
State	Status	Error Control	Accept Pause		Accept Stop
abiosdsk	Abiosdsk	Not Available	Kernel Driver		No
Disabled	Stopped	OK	Ignore	No	No
acpi	Microsoft ACPI Driver	c:\windows\system32\drivers\acpi.sys	Kernel Driver		Yes
Boot	Running	OK	Normal	No	Yes

acpiec Driver No	ACPIEC No	c:\windows\system32\drivers\acpiec.sys Disabled Stopped	OK Normal	Kernel Driver No	No	
adpu160m Disabled	adpu160m Stopped	Not Available OK	Normal No	Kernel Driver No	No	
adpu320 Disabled	adpu320 Stopped	Not Available OK	Normal No	Kernel Driver No	No	
afcnt Disabled	afcnt Stopped	Not Available OK	Normal No	Kernel Driver No	No	
afd	AFD Networking Support Environment					
c:\windows\system32\drivers\afd.sys Auto	Running OK	Normal No	Yes	Kernel Driver Yes		
aha154x Disabled	Aha154x Stopped	Not Available OK	Normal No	Kernel Driver No	No	
aic78u2 Disabled	aic78u2 Stopped	Not Available OK	Normal No	Kernel Driver No	No	
aic78xx Disabled	aic78xx Stopped	Not Available OK	Normal No	Kernel Driver No	No	
aliide Disabled	AliIde Stopped	Not Available OK	Normal No	Kernel Driver No	No	
asynmac	RAS Asynchronous Media Driver					
c:\windows\system32\drivers\asynmac.sys Manual	Stopped OK	Normal No	No	Kernel Driver No		
atapi	Standard IDE/ESDI Hard Disk Controller					
c:\windows\system32\drivers\atapi.sys Boot	Running OK	Normal No	Yes	Kernel Driver Yes		
atdisk Disabled	Atdisk Stopped	Not Available OK	Ignore No	Kernel Driver No	No	
ati2mpad Driver	ati2mpad Yes	c:\windows\system32\drivers\ati2mpad.sys Manual	Running OK	Ignore No	Kernel Driver No	
atmarpc	ATM ARP Client Protocol					
c:\windows\system32\drivers\atmarpc.sys Manual	Stopped OK	Normal No	No	Kernel Driver No		
audstub Kernel Driver No	Audio Stub Driver Yes	c:\windows\system32\drivers\audstub.sys Manual	Running OK	Normal No		
b57w2k	Broadcom NetXtreme Gigabit Ethernet					
c:\windows\system32\drivers\b57xp32.sys Manual	Stopped OK	Normal No	No	Kernel Driver No		
beep Driver	Beep Yes	c:\windows\system32\drivers\beep.sys System	Running OK	Normal No	Kernel Driver No	
cbidf2k Driver	cbidf2k No	c:\windows\system32\drivers\cbidf2k.sys Disabled	Stopped OK	Normal No	Kernel Driver No	
cdfs Driver	Cdfs Yes	c:\windows\system32\drivers\cdfs.sys Disabled	Running OK	Normal No	File System No	
cdrom Kernel Driver No	CD-ROM Driver Yes	c:\windows\system32\drivers\cdrom.sys System	Running OK	Normal No		
changer System	Changer Stopped	Not Available OK	Ignore No	Kernel Driver No	No	
clusdisk Kernel Driver No	Cluster Disk Driver No	c:\windows\system32\drivers\clusdisk.sys Disabled	Stopped OK	Normal No		
cmdide Disabled	CmdIde Stopped	Not Available OK	Normal No	Kernel Driver No	No	
cpqarray Disabled	Cpqarray Stopped	Not Available OK	Normal No	Kernel Driver No	No	
cpqarray2 Disabled	cpqarray2 Stopped	Not Available OK	Normal No	Kernel Driver No	No	
cpqcissm Disabled	cpqcissm Stopped	Not Available OK	Normal No	Kernel Driver No	No	
cpqfcalm Disabled	cpqfcalm Stopped	Not Available OK	Normal No	Kernel Driver No	No	
credisk	CRC Disk Filter Driver					
c:\windows\system32\drivers\credisk.sys Boot	Running OK	Normal No	Yes	Kernel Driver Yes		
dac960nt Disabled	dac960nt Stopped	Not Available OK	Normal No	Kernel Driver No	No	
dellcerc Disabled	dellcerc Stopped	Not Available OK	Normal No	Kernel Driver No	No	
dfsdriver Driver	DfsDriver Yes	c:\windows\system32\drivers\dfs.sys Boot	Running OK	Normal No	File System No	
disk	Disk Driver					
c:\windows\system32\drivers\disk.sys Kernel Driver	Yes	Boot	Running	OK	Normal	
dmboot Driver	dmboot No	c:\windows\system32\drivers\dmboot.sys Disabled	Stopped OK	Normal No	Kernel Driver No	
dmio	Logical Disk Manager Driver					
c:\windows\system32\drivers\dmio.sys Boot	Running OK	Normal No	Yes	Kernel Driver Yes		
dmload Driver	dmload Yes	c:\windows\system32\drivers\dmload.sys Boot	Running OK	Normal No	Kernel Driver No	
dpti2o Disabled	dpti2o Stopped	Not Available OK	Normal No	Kernel Driver No	No	
e100b	Intel(R) PRO Adapter Driver					
c:\windows\system32\drivers\e100b325.sys Manual	Running OK	Normal No	Yes	Kernel Driver Yes		
em Driver	em No	c:\windows\system32\drivers\em.sys Manual	Stopped OK	Normal No	Kernel Driver No	
fastfat Driver	Fastfat No	c:\windows\system32\drivers\fastfat.sys Disabled	Stopped OK	Normal No	File System No	
fdc	Floppy Disk Controller Driver					
c:\windows\system32\drivers\fdc.sys Manual	Running OK	Normal No	Yes	Kernel Driver Yes		
fips Driver	Fips Yes	c:\windows\system32\drivers\fips.sys System	Running OK	Normal No	Kernel Driver No	
flpydisk Kernel Driver	Floppy Disk Driver Yes	c:\windows\system32\drivers\flpydisk.sys Manual	Running OK	Normal No	Kernel Driver No	
ftdisk	Volume Manager Driver					
c:\windows\system32\drivers\ftdisk.sys Boot	Running OK	Normal No	Yes	Kernel Driver Yes		
gpc	Generic Packet Classifier					
c:\windows\system32\drivers\msgpc.sys Manual	Running OK	Normal No	Yes	Kernel Driver Yes		
hpn Disabled	hpn Stopped	Not Available OK	Normal No	Kernel Driver No	No	
hpt3xx Disabled	hpt3xx Stopped	Not Available OK	Normal No	Kernel Driver No	No	
http Driver	HTTP No	c:\windows\system32\drivers\http.sys Manual	Stopped OK	Normal No	Kernel Driver No	
i2omgmt System	i2omgmt Stopped	Not Available OK	Normal No	Kernel Driver No	No	
i2omp Disabled	i2omp Stopped	Not Available OK	Normal No	Kernel Driver No	No	
i8042prt c:\windows\system32\drivers\i8042prt.sys	i8042 Keyboard and PS/2 Mouse Port Driver Running	OK	Normal	No	Yes	Kernel Driver Yes
iirsp Disabled	iirsp Stopped	Not Available OK	Normal No	Kernel Driver No	No	
imapi c:\windows\system32\drivers\imapi.sys	CD-Burning Filter Driver Stopped	OK	Normal	No	No	Kernel Driver No

intelide	IntelIde	Not Available	Kernel Driver	No	No					netbios	NetBIOS Interface	c:\windows\system32\drivers\netbios.sys									
Disabled	Stopped	OK	Normal	No	No					File System Driver	Yes	System	Running	OK	Normal						
ipfilterdriver	IP Traffic Filter Driver									No	Yes										
c:\windows\system32\drivers\ipfltdrv.sys	Kernel Driver	No	No							netbt	NetBios over Tcpip	c:\windows\system32\drivers\netbt.sys									
Manual	Stopped	OK	Normal	No	No					Kernel Driver	Yes	System	Running	OK	Normal						
ipinip	IP in IP Tunnel Driver	c:\windows\system32\drivers\ipinip.sys	Kernel Driver	No	Manual	Stopped	OK	Normal		No	Yes										
Kernel Driver	No	Manual	Stopped	OK	Normal					nfrd960	nfrd960	Not Available	Kernel Driver	No	No						
No	Yes									Disabled	Stopped	OK	Normal	No	No						
ipnat	IP Network Address Translator	c:\windows\system32\drivers\ipnat.sys	Kernel Driver	No						npfs	Npfs	c:\windows\system32\drivers\npfs.sys	File System								
Manual	Stopped	OK	Normal	No	No					Driver	Yes	System	Running	OK	Normal	No					
ipsec	IPSEC driver	c:\windows\system32\drivers\ipsec.sys	Kernel Driver	No	Yes	System	Running	OK	Normal	Yes											
Kernel Driver	Yes	System	Running	OK	Normal					ntfs	Ntfs	c:\windows\system32\drivers\ntfs.sys	File System								
No	Yes									Driver	Yes	Disabled	Running	OK	Normal	No					
ipsraidn	ipsraidn	Not Available	Kernel Driver	No						Yes											
Disabled	Stopped	OK	Normal	No	No					null	Null	c:\windows\system32\drivers\null.sys	Kernel								
isapnp	PnP ISA/EISA Bus Driver	c:\windows\system32\drivers\isapnp.sys	Kernel Driver	Yes						Driver	Yes	System	Running	OK	Normal	No					
Boot	Running	OK	Critical	No	Yes					Yes											
kbdclass	Keyboard Class Driver	c:\windows\system32\drivers\kbdclass.sys	Kernel Driver	Yes						parport	Parport	c:\windows\system32\drivers\parport.sys	Kernel								
System	Running	OK	Normal	No	Yes					Driver	No	Manual	Stopped	OK	Ignore	No					
ksecdd	KSecDD	c:\windows\system32\drivers\ksecdd.sys	Kernel	No						No											
Driver	Yes	Boot	Running	OK	Normal	No				partmgr	Partition Manager	c:\windows\system32\drivers\partmgr.sys	Kernel								
Yes										Kernel Driver	Yes	Boot	Running	OK	Normal						
lp6nds35	lp6nds35	Not Available	Kernel Driver	No						No	Yes										
Disabled	Stopped	OK	Normal	No	No					pci	PCI Bus Driver	c:\windows\system32\drivers\pci.sys	Kernel								
mmmd	mmmd	c:\windows\system32\drivers\mmmd.sys	Kernel	No						Kernel Driver	Yes	Boot	Running	OK	Critical						
Driver	Yes	System	Running	OK	Ignore	No				No	Yes										
Yes										pciide	PCIIde	Not Available	Kernel Driver	No	No						
modem	Modem	c:\windows\system32\drivers\modem.sys	Kernel	No						Disabled	Stopped	OK	Normal	No	No						
Driver	No	Manual	Stopped	OK	Ignore	No				pciview	PCIVIEW	??c:\pciview\pciview.sys	Kernel Driver	No	No						
No										No	Manual	Stopped	OK	Normal	No	No					
mouclass	Mouse Class Driver	c:\windows\system32\drivers\mouclass.sys	Kernel Driver	No	Yes	System	Running	OK	Normal	pcmcia	Pcmcia	c:\windows\system32\drivers\pcmcia.sys	Kernel								
Kernel Driver	Yes	System	Running	OK	Normal					Driver	No	Disabled	Stopped	OK	Normal	No					
No	Yes									No											
mountmgr	Mount Point Manager	c:\windows\system32\drivers\mountmgr.sys	Kernel Driver	No	Yes	Boot	Running	OK	Normal	pdcomp	PDCOMP	Not Available	Kernel Driver	No	No						
Kernel Driver	Yes	Boot	Running	OK	Normal					Manual	Stopped	OK	Ignore	No	No						
No	Yes									No	PDFRAME	Not Available	Kernel Driver	No	No						
mraid35x	mraid35x	Not Available	Kernel Driver	No						pdframe	PDFRAME	Not Available	Kernel Driver	No	No						
Disabled	Stopped	OK	Normal	No	No					No	Manual	Stopped	OK	Ignore	No	No					
mrxdav	WebDav Client Redirector	c:\windows\system32\drivers\mrxdav.sys	File System Driver	No						pdreli	PDRELI	Not Available	Kernel Driver	No	No						
Manual	Stopped	OK	Normal	No	No					Manual	Stopped	OK	Ignore	No	No						
mrxsmb	MRXSMB	c:\windows\system32\drivers\mrxsmb.sys	File System	No						pdrframe	PDRFRAME	Not Available	Kernel Driver	No	No						
Driver	Yes	System	Running	OK	Normal	No				No	Manual	Stopped	OK	Ignore	No	No					
Yes										perc2	perc2	Not Available	Kernel Driver	No	No						
msfs	Msfs	c:\windows\system32\drivers\msfs.sys	File System	No						Disabled	Stopped	OK	Normal	No	No						
Driver	Yes	System	Running	OK	Normal	No				perc2hib	perc2hib	Not Available	Kernel Driver	No	No						
Yes										Disabled	Stopped	OK	Normal	No	No						
mup	Mup	c:\windows\system32\drivers\mup.sys	File System	No						pnpmem	Microsoft Memory Module Driver	c:\windows\system32\drivers\pnpmem.sys	Kernel Driver	Yes							
Driver	Yes	Boot	Running	OK	Normal	No				Manual	Running	OK	Normal	No	Yes						
Yes										pptpminiport	WAN Miniport (PPTP)	c:\windows\system32\drivers\rasppptp.sys	Kernel Driver	Yes							
ndis	NDIS System Driver	c:\windows\system32\drivers\ndis.sys	Kernel Driver	No	Yes	Boot	Running	OK	Normal	Manual	Running	OK	Normal	No	Yes						
Kernel Driver	Yes	Boot	Running	OK	Normal					processor	Processor Driver	c:\windows\system32\drivers\processr.sys	Kernel Driver	Yes	Manual	Running	OK	Normal			
No	Yes									No	Yes										
ndistapi	Remote Access NDIS TAPI Driver	c:\windows\system32\drivers\ndistapi.sys	Kernel Driver	Yes						ptilink	Direct Parallel Link Driver	c:\windows\system32\drivers\ptilink.sys	Kernel Driver	Yes							
Manual	Running	OK	Normal	No	Yes					Manual	Running	OK	Normal	No	Yes						
ndisuio	NDIS Usermode I/O Protocol	c:\windows\system32\drivers\ndisuio.sys	Kernel Driver	No						q11080	q11080	Not Available	Kernel Driver	No	No						
Manual	Stopped	OK	Normal	No	No					Disabled	Stopped	OK	Normal	No	No						
ndiswan	Remote Access NDIS WAN Driver	c:\windows\system32\drivers\ndiswan.sys	Kernel Driver	Yes						q110wnt	Q110wnt	Not Available	Kernel Driver	No	No						
Manual	Running	OK	Normal	No	Yes					Disabled	Stopped	OK	Normal	No	No						
ndproxy	NDIS Proxy	c:\windows\system32\drivers\ndproxy.sys	Kernel Driver	No						q112160	q112160	Not Available	Kernel Driver	No	No						
Kernel Driver	Yes	Manual	Running	OK	Normal	No				Disabled	Stopped	OK	Normal	No	No						
No	Yes									q11240	q11240	Not Available	Kernel Driver	No	No						
										Disabled	Stopped	OK	Normal	No	No						
										q11280	q11280	Not Available	Kernel Driver	No	No						
										Disabled	Stopped	OK	Normal	No	No						



ACPI Multiprocessor Node	No	COMPUTER	1.20.0.0	Numeric data processor	Yes	SYSTEM	5.2.3790.0	10/1/2002
2/11/2002 (Node-based Computers)		oem0.inf	Not Available	(Standard system devices)	machine.inf			Not Available
ROOT\ACPI_HAL\0000				ACPI\PNP0C04\4&7FD7688&0				
Microsoft ACPI-Compliant System	Yes	SYSTEM	5.2.3790.0	Motherboard resources	Yes	SYSTEM	5.2.3790.0	10/1/2002
10/1/2002 Microsoft acpi.inf	Not Available			(Standard system devices)	machine.inf			Not Available
ACPI_HAL\PNP0C08\0				ACPI\PNP0C02\3				
Not Available	Not Available	Not Available	Not Available	VIA Bus Master IDE Controller	Yes	HDC	5.2.3790.0	10/1/2002
Available	Not Available	Not Available	Not Available	VIA Technologies, Inc.	mshdc.inf			Not Available
Not Available	ACPI\IBM37D4\2&DABA3FF&0			PCI\VEN_1106&DEV_0571&SUBSYS_02A61014&REV_06\3&267A616A&0&29				
PCI bus	Yes	SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)	machine.inf			Not Available
ACPI\PNP0A03\0				Primary IDE Channel	Yes	HDC	5.2.3790.0	10/1/2002 (Standard IDE ATA/ATAPI controllers)
PCI standard host CPU bridge	Yes	SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)	machine.inf			Not Available
PCI\VEN_1014&DEV_0302&SUBSYS_00000000&REV_05\3&267A616A&0&00				PCI\IDE\IDECHANNEL\4&29582549&0&0				
RAGE XL PCI	No	DISPLAY	5.0.2195.5012	12/28/2001				
ATI Technologies Inc.		oem10.inf	Not Available	IDE\CDROMHL-DT-ST_DVD-ROM_GDR8081N_____0012_____\5&CBC355F&0&0.0.0				
PCI\VEN_1002&DEV_4752&SUBSYS_02A61014&REV_27\3&267A616A&0&18				Secondary IDE Channel	Yes	HDC	5.2.3790.0	10/1/2002 (Standard IDE ATA/ATAPI controllers)
Default Monitor	Yes	MONITOR	5.1.2001.0	6/6/2001				
(Standard monitor types)		monitor.inf	Not Available	PCI\IDE\IDECHANNEL\4&29582549&0&1				
DISPLAYDEFAULT_MONITOR\4&82045C7&0&80000000&00&03				VIA Rev 5 or later USB Universal Host Controller	Yes	USB	5.2.3790.0	10/1/2002 VIA Technologies usbport.inf
Other PCI Bridge Device	Not Available	UNKNOWN		PCI\VEN_1106&DEV_3038&SUBSYS_02A61014&REV_16\3&267A616A&0&2A				Not Available
Not Available	Not Available	Not Available	Not Available	VIA Rev 5 or later USB Universal Host Controller	Yes	USB	5.2.3790.0	10/1/2002 VIA Technologies usbport.inf
Available	Not Available			PCI\VEN_1106&DEV_3038&SUBSYS_02A61014&REV_16\3&267A616A&0&2B				Not Available
PCI\VEN_1014&DEV_010F&SUBSYS_01131014&REV_00\3&267A616A&0&20				VIA Tech Power Management controller	Yes	SYSTEM	5.2.3790.0	10/1/2002 VIA machine.inf
VIA Tech PCI to ISA bridge	Yes	SYSTEM	5.2.3790.0	10/1/2002 VIA machine.inf				Not Available
VIA				PCI\VEN_1106&DEV_0686&SUBSYS_00000000&REV_40\3&267A616A&0&28				
PCI\VEN_1106&DEV_0686&SUBSYS_00000000&REV_40\3&267A616A&0&28				ISAPNP Read Data Port	Yes	SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)
ISAPNP Read Data Port	Yes	SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)	machine.inf			Not Available
(Standard system devices)				ISAPNP\READDATAPORT\0				
ISAPNP\READDATAPORT\0				Motherboard resources	Yes	SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)
Motherboard resources	Yes	SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)	machine.inf			Not Available
(Standard system devices)				ACPI\PNP0C02\2				
ACPI\PNP0C02\2				Standard 101/102-Key or Microsoft Natural PS/2 Keyboard	Yes			
Standard 101/102-Key or Microsoft Natural PS/2 Keyboard	Yes			KEYBOARD	5.2.3790.0	10/1/2002 (Standard keyboards)	keyboard.inf	Not Available
KEYBOARD	5.2.3790.0	10/1/2002 (Standard keyboards)	keyboard.inf	Not Available	ACPI\PNP0303\4&7FD7688&0			
Not Available	ACPI\PNP0303\4&7FD7688&0			PS/2 Compatible Mouse	Yes	MOUSE	5.2.3790.0	10/1/2002 Microsoft msmouse.inf
PS/2 Compatible Mouse	Yes	MOUSE	5.2.3790.0	10/1/2002 Microsoft msmouse.inf	Not Available			
Microsoft msmouse.inf	Not Available			ACPI\PNP0F13\4&7FD7688&0				
ACPI\PNP0F13\4&7FD7688&0				Standard floppy disk controller	Yes	FDC	5.2.3790.0	10/1/2002 (Standard floppy disk controllers)
Standard floppy disk controller	Yes	FDC	5.2.3790.0	10/1/2002 (Standard floppy disk controllers)	fdc.inf			Not Available
(Standard floppy disk controllers)				ACPI\PNP0700\4&7FD7688&0				
ACPI\PNP0700\4&7FD7688&0				Floppy disk drive	Yes	FLOPPYDISK	5.2.3790.0	10/1/2002 (Standard floppy disk drives)
Floppy disk drive	Yes	FLOPPYDISK	5.2.3790.0	10/1/2002 (Standard floppy disk drives)	flpydisk.inf			Not Available
(Standard floppy disk drives)				FDC\GENERIC_FLOPPY_DRIVE\5&17D92A40&0&0				
FDC\GENERIC_FLOPPY_DRIVE\5&17D92A40&0&0				Communications Port	Yes	SYSTEM	5.2.3790.0	10/1/2002 (Standard port types)
Communications Port	Yes	SYSTEM	5.2.3790.0	10/1/2002 (Standard port types)	msports.inf			Not Available
(Standard port types)				Advanced programmable interrupt controller	Yes	SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)
Advanced programmable interrupt controller	Yes	SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)	machine.inf			Not Available
(Standard system devices)				Available	ACPI\PNP0003\4&7FD7688&0			
Available	ACPI\PNP0003\4&7FD7688&0			Direct memory access controller	Yes	SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)
Direct memory access controller	Yes	SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)	machine.inf			Not Available
(Standard system devices)				ACPI\PNP0200\4&7FD7688&0				
ACPI\PNP0200\4&7FD7688&0				System timer	Yes	SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)
System timer	Yes	SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)	machine.inf			Not Available
(Standard system devices)				ACPI\PNP0100\4&7FD7688&0				
ACPI\PNP0100\4&7FD7688&0				System CMOS/real time clock	Yes	SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)
System CMOS/real time clock	Yes	SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)	machine.inf			Not Available
(Standard system devices)				ACPI\PNP0B00\4&7FD7688&0				
ACPI\PNP0B00\4&7FD7688&0				System speaker	Yes	SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)
System speaker	Yes	SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)	machine.inf			Not Available
(Standard system devices)				ACPI\PNP0800\4&7FD7688&0				
ACPI\PNP0800\4&7FD7688&0								

Processor (Standard processor types)	Yes	PROCESSOR	5.2.3790.0	10/1/2002	(Standard processor types)	cpu.inf	Not Available		
ACPI\GENUINEINTEL_-X86_FAMILY_15_MODEL_2\10									
Processor (Standard processor types)	Yes	PROCESSOR	5.2.3790.0	10/1/2002	(Standard processor types)	cpu.inf	Not Available		
ACPI\GENUINEINTEL_-X86_FAMILY_15_MODEL_2\11									
Processor (Standard processor types)	Yes	PROCESSOR	5.2.3790.0	10/1/2002	(Standard processor types)	cpu.inf	Not Available		
ACPI\GENUINEINTEL_-X86_FAMILY_15_MODEL_2\12									
Processor (Standard processor types)	Yes	PROCESSOR	5.2.3790.0	10/1/2002	(Standard processor types)	cpu.inf	Not Available		
ACPI\GENUINEINTEL_-X86_FAMILY_15_MODEL_2\13									
Processor (Standard processor types)	Yes	PROCESSOR	5.2.3790.0	10/1/2002	(Standard processor types)	cpu.inf	Not Available		
ACPI\GENUINEINTEL_-X86_FAMILY_15_MODEL_2\14									
Processor (Standard processor types)	Yes	PROCESSOR	5.2.3790.0	10/1/2002	(Standard processor types)	cpu.inf	Not Available		
ACPI\GENUINEINTEL_-X86_FAMILY_15_MODEL_2\15									
Memory Module (Standard system devices)	Yes	MEMORY	5.2.3790.0	10/1/2002	(Standard system devices)	memory.inf	Not Available	Microsoft ACPI\PNP0C80\0	
Memory Module (Standard system devices)	Yes	MEMORY	5.2.3790.0	10/1/2002	(Standard system devices)	memory.inf	Not Available	Microsoft ACPI\PNP0C80\2	
PCI bus (Standard system devices)	Yes	SYSTEM	5.2.3790.0	10/1/2002	(Standard system devices)	machine.inf	Not Available	ACPI\PNP0A03\1	
PCI standard host CPU bridge (Standard system devices)	Yes	SYSTEM	5.2.3790.0	10/1/2002	(Standard system devices)	machine.inf	Not Available		
PCI\VEN_1014&DEV_0302&SUBSYS_00000000&REV_05\3&13C0B0C5&0&00									
LSI Logic PCI-X Ultra320 SCSI Host Adapter (Standard system devices)	Yes				(Standard system devices)				
SCSIADAPTER	5.2.3790.0	10/1/2002	LSI Logic Inc.					pnpscsi.inf	
PCI\VEN_1000&DEV_0030&SUBSYS_02921014&REV_07\3&13C0B0C5&0&18									
Disk drive (Standard disk drives)	Yes	DISKDRIVE	5.2.3790.0	10/1/2002	(Standard disk drives)	disk.inf	Not Available		
SCSI\DISK&VEN_IBM-ESXS&PROD_DTN036C3UCDY10FN&REV_S23\4&3CF91A&0&000									
SCSI Processor Device	Yes	SYSTEM	5.2.3790.0	10/1/2002					
IBM (Standard system devices)					(Standard system devices)				
SCSI\PROCESSOR&VEN_IBM&PROD_25P3495A_S320__1&REV_1\4&3CF91A&0&080									
IBM Dummy Device (Standard system devices)	Yes	SYSTEM	5.2.3790.0	10/1/2002	(Standard system devices)				
SCSI\BRIDGE&VEN_IBM&PROD_DUMMY_DEVICE&REV_4.80\4&3CF91A&0&100									
LSI Logic PCI-X Ultra320 SCSI Host Adapter (Standard system devices)	Yes				(Standard system devices)				
SCSIADAPTER	5.2.3790.0	10/1/2002	LSI Logic Inc.					pnpscsi.inf	
PCI\VEN_1000&DEV_0030&SUBSYS_02921014&REV_07\3&13C0B0C5&0&19									
IBM Dummy Device (Standard system devices)	Yes	SYSTEM	5.2.3790.0	10/1/2002	(Standard system devices)				
SCSI\BRIDGE&VEN_IBM&PROD_DUMMY_DEVICE&REV_4.80\4&12F6A9BF&0&100									
Broadcom NetXtreme Gigabit Ethernet (Standard system devices)	Yes	NET	6.34.0.0		(Standard system devices)				
2/17/2003 Broadcom (Standard system devices)					(Standard system devices)				
PCI\VEN_14E4&DEV_1648&SUBSYS_02A61014&REV_02\3&13C0B0C5&0&20									
Broadcom NetXtreme Gigabit Ethernet (Standard system devices)	Yes	NET	6.34.0.0		(Standard system devices)				
2/17/2003 Broadcom (Standard system devices)					(Standard system devices)				
PCI\VEN_14E4&DEV_1648&SUBSYS_02A61014&REV_02\3&13C0B0C5&0&21									
PCI bus (Standard system devices)	Yes	SYSTEM	5.2.3790.0	10/1/2002	(Standard system devices)	machine.inf	Not Available	ACPI\PNP0A03\2	
PCI standard host CPU bridge (Standard system devices)	Yes	SYSTEM	5.2.3790.0	10/1/2002	(Standard system devices)	machine.inf	Not Available		
PCI\VEN_1014&DEV_0302&SUBSYS_00000000&REV_05\3&1070020&0&00									
QLogic QLA23xx PCI Fibre Channel Adapter (Standard system devices)	No				(Standard system devices)				
SCSIADAPTER	8.2.1.0	3/28/2003	QLogic					oem11.inf	Not Available
PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&1070020&0&08									
Disk drive (Standard disk drives)	Yes	DISKDRIVE	5.2.3790.0	10/1/2002	(Standard disk drives)	disk.inf	Not Available		
SCSI\DISK&VEN_IBM&PROD_1742&REV_0530\4&3B4E3515&1&000									
Qlogic processor device	Yes	SYSTEM	5.2.3790.0	10/1/2002					
QLOGIC (Standard system devices)					(Standard system devices)				
SCSI\PROCESSOR&VEN_QLOGIC&PROD_PSEUDO_LUN&REV_4&3B4E3515&1&07F0									
PCI bus (Standard system devices)	Yes	SYSTEM	5.2.3790.0	10/1/2002	(Standard system devices)	machine.inf	Not Available	ACPI\PNP0A03\3	
PCI standard host CPU bridge (Standard system devices)	Yes	SYSTEM	5.2.3790.0	10/1/2002	(Standard system devices)	machine.inf	Not Available		
PCI\VEN_1014&DEV_0302&SUBSYS_00000000&REV_05\3&29E81982&0&00									
PCI bus (Standard system devices)	Yes	SYSTEM	5.2.3790.0	10/1/2002	(Standard system devices)	machine.inf	Not Available	ACPI\PNP0A03\4	
PCI standard host CPU bridge (Standard system devices)	Yes	SYSTEM	5.2.3790.0	10/1/2002	(Standard system devices)	machine.inf	Not Available		
PCI\VEN_1014&DEV_0302&SUBSYS_00000000&REV_05\3&172E68DD&0&00									
QLogic QLA23xx PCI Fibre Channel Adapter (Standard system devices)	No				(Standard system devices)				
SCSIADAPTER	8.2.0.0	10/22/2002	QLogic					oem3.inf	Not Available
PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&172E68DD&0&18									
Disk drive (Standard disk drives)	Yes	DISKDRIVE	5.2.3790.0	10/1/2002	(Standard disk drives)	disk.inf	Not Available		
SCSI\DISK&VEN_IBM&PROD_1742&REV_0530\4&26D408A5&0&000									
Disk drive (Standard disk drives)	Yes	DISKDRIVE	5.2.3790.0	10/1/2002	(Standard disk drives)	disk.inf	Not Available		
SCSI\DISK&VEN_IBM&PROD_1742&REV_0530\4&26D408A5&0&001									
Disk drive (Standard disk drives)	Yes	DISKDRIVE	5.2.3790.0	10/1/2002	(Standard disk drives)	disk.inf	Not Available		
SCSI\DISK&VEN_IBM&PROD_1742&REV_0530\4&26D408A5&0&012									
Disk drive (Standard disk drives)	Yes	DISKDRIVE	5.2.3790.0	10/1/2002	(Standard disk drives)	disk.inf	Not Available		
SCSI\DISK&VEN_IBM&PROD_1742&REV_0530\4&26D408A5&0&013									
Qlogic processor device	Yes	SYSTEM	5.2.3790.0	10/1/2002					
QLOGIC (Standard system devices)					(Standard system devices)				
SCSI\PROCESSOR&VEN_QLOGIC&PROD_PSEUDO_LUN&REV_4&26D408A5&0&07F0									
PCI bus (Standard system devices)	Yes	SYSTEM	5.2.3790.0	10/1/2002	(Standard system devices)	machine.inf	Not Available	ACPI\PNP0A03\5	
PCI standard host CPU bridge (Standard system devices)	Yes	SYSTEM	5.2.3790.0	10/1/2002	(Standard system devices)	machine.inf	Not Available		
PCI\VEN_1014&DEV_0302&SUBSYS_00000000&REV_05\3&474B838&0&00									
QLogic QLA23xx PCI Fibre Channel Adapter (Standard system devices)	No				(Standard system devices)				
SCSIADAPTER	8.2.0.0	10/22/2002	QLogic					oem3.inf	Not Available
PCI\VEN_1077&DEV_2312&SUBSYS_010C1077&REV_02\3&474B838&0&08									
Qlogic processor device	Yes	SYSTEM	5.2.3790.0	10/1/2002					
QLOGIC (Standard system devices)					(Standard system devices)				
SCSI\PROCESSOR&VEN_QLOGIC&PROD_PSEUDO_LUN&REV_4&4001865&0&07F0									
Motherboard resources (Standard system devices)	Yes	SYSTEM	5.2.3790.0	10/1/2002	(Standard system devices)	machine.inf	Not Available		
ACPI\PNP0C02\10									
Processor (Standard processor types)	Yes	PROCESSOR	5.2.3790.0	10/1/2002	(Standard processor types)	cpu.inf	Not Available		
ACPI\GENUINEINTEL_-X86_FAMILY_15_MODEL_2\16									



Processor (processor types)	Yes	PROCESSOR	5.2.3790.0	10/1/2002	(Standard
ACPI\GENUINEINTEL_-X86_FAMILY_15_MODEL_2\17	cpu.inf	Not Available			
Processor (processor types)	Yes	PROCESSOR	5.2.3790.0	10/1/2002	(Standard
ACPI\GENUINEINTEL_-X86_FAMILY_15_MODEL_2\18	cpu.inf	Not Available			
Processor (processor types)	Yes	PROCESSOR	5.2.3790.0	10/1/2002	(Standard
ACPI\GENUINEINTEL_-X86_FAMILY_15_MODEL_2\19	cpu.inf	Not Available			
Processor (processor types)	Yes	PROCESSOR	5.2.3790.0	10/1/2002	(Standard
ACPI\GENUINEINTEL_-X86_FAMILY_15_MODEL_2\20	cpu.inf	Not Available			
Processor (processor types)	Yes	PROCESSOR	5.2.3790.0	10/1/2002	(Standard
ACPI\GENUINEINTEL_-X86_FAMILY_15_MODEL_2\21	cpu.inf	Not Available			
Processor (processor types)	Yes	PROCESSOR	5.2.3790.0	10/1/2002	(Standard
ACPI\GENUINEINTEL_-X86_FAMILY_15_MODEL_2\22	cpu.inf	Not Available			
Processor (processor types)	Yes	PROCESSOR	5.2.3790.0	10/1/2002	(Standard
ACPI\GENUINEINTEL_-X86_FAMILY_15_MODEL_2\23	cpu.inf	Not Available			
Processor (processor types)	Yes	PROCESSOR	5.2.3790.0	10/1/2002	(Standard
ACPI\GENUINEINTEL_-X86_FAMILY_15_MODEL_2\24	cpu.inf	Not Available			
Processor (processor types)	Yes	PROCESSOR	5.2.3790.0	10/1/2002	(Standard
ACPI\GENUINEINTEL_-X86_FAMILY_15_MODEL_2\25	cpu.inf	Not Available			
Processor (processor types)	Yes	PROCESSOR	5.2.3790.0	10/1/2002	(Standard
ACPI\GENUINEINTEL_-X86_FAMILY_15_MODEL_2\26	cpu.inf	Not Available			
Processor (processor types)	Yes	PROCESSOR	5.2.3790.0	10/1/2002	(Standard
ACPI\GENUINEINTEL_-X86_FAMILY_15_MODEL_2\27	cpu.inf	Not Available			
Processor (processor types)	Yes	PROCESSOR	5.2.3790.0	10/1/2002	(Standard
ACPI\GENUINEINTEL_-X86_FAMILY_15_MODEL_2\28	cpu.inf	Not Available			
Processor (processor types)	Yes	PROCESSOR	5.2.3790.0	10/1/2002	(Standard
ACPI\GENUINEINTEL_-X86_FAMILY_15_MODEL_2\29	cpu.inf	Not Available			
Processor (processor types)	Yes	PROCESSOR	5.2.3790.0	10/1/2002	(Standard
ACPI\GENUINEINTEL_-X86_FAMILY_15_MODEL_2\30	cpu.inf	Not Available			
Processor (processor types)	Yes	PROCESSOR	5.2.3790.0	10/1/2002	(Standard
ACPI\GENUINEINTEL_-X86_FAMILY_15_MODEL_2\31	cpu.inf	Not Available			
Memory Module	Yes	MEMORY	5.2.3790.0	10/1/2002	Microsoft
memory.inf	Not Available	ACPI\PNP0C80\10			
Memory Module	Yes	MEMORY	5.2.3790.0	10/1/2002	Microsoft
memory.inf	Not Available	ACPI\PNP0C80\12			
PCI bus (Standard system devices)	Yes	SYSTEM	5.2.3790.0	10/1/2002	(Standard system
PCI standard host CPU bridge	Yes	SYSTEM	5.2.3790.0	10/1/2002	(Standard system devices)
PCI\VEN_1014&DEV_0302&SUBSYS_00000000&REV_05\3&A985F74&0&00	machine.inf	Not Available			ACPI\PNP0A03\10
Other PCI Bridge Device	Not Available	Not Available			Not Available
Not Available	Not Available	Not Available			Not Available
Available	Not Available				
PCI\VEN_1014&DEV_010F&SUBSYS_01131014&REV_00\3&A985F74&0&20	machine.inf	Not Available			
VIA Tech PCI to ISA bridge	Yes	SYSTEM	5.2.3790.0	10/1/2002	
VIA	machine.inf	Not Available			
PCI\VEN_1106&DEV_0686&SUBSYS_00000000&REV_40\3&A985F74&0&28	machine.inf	Not Available			
VIA Tech Power Management controller	Yes	SYSTEM	5.2.3790.0	10/1/2002	
10/1/2002 VIA	machine.inf	Not Available			
PCI\VEN_1106&DEV_3057&SUBSYS_02A61014&REV_40\3&A985F74&0&2C	machine.inf	Not Available			
PCI bus (Standard system devices)	Yes	SYSTEM	5.2.3790.0	10/1/2002	(Standard system
PCI standard host CPU bridge	Yes	SYSTEM	5.2.3790.0	10/1/2002	(Standard system devices)
PCI\VEN_1014&DEV_0302&SUBSYS_00000000&REV_05\3&1D521019&0&00	machine.inf	Not Available			ACPI\PNP0A03\11
LSI Logic PCI-X Ultra320 SCSI Host Adapter	Yes				
SCSIADAPTER	5.2.3790.0	10/1/2002	LSI Logic Inc.	pnpscsi.inf	Not Available
PCI\VEN_1000&DEV_0030&SUBSYS_02921014&REV_07\3&1D521019&0&18	machine.inf	Not Available			
SCSI Processor Device	Yes	SYSTEM	5.2.3790.0	10/1/2002	
IBM	scsidev.inf	Not Available			
SCSI\PROCESSOR&VEN_IBM&PROD_25P3495A_S320_1&REV_1\4&8B18983&0&080	machine.inf	Not Available			
IBM Dummy Device	Yes	SYSTEM	5.2.3790.0	10/1/2002	IBM
scsidev.inf	Not Available				
SCSI\BRIDGE&VEN_IBM&PROD_DUMMY_DEVICE&REV_4.80\4&8B18983&0&100	machine.inf	Not Available			
LSI Logic PCI-X Ultra320 SCSI Host Adapter	Yes				
SCSIADAPTER	5.2.3790.0	10/1/2002	LSI Logic Inc.	pnpscsi.inf	Not Available
PCI\VEN_1000&DEV_0030&SUBSYS_02921014&REV_07\3&1D521019&0&19	machine.inf	Not Available			
IBM Dummy Device	Yes	SYSTEM	5.2.3790.0	10/1/2002	IBM
scsidev.inf	Not Available				
SCSI\BRIDGE&VEN_IBM&PROD_DUMMY_DEVICE&REV_4.80\4&1B6B3A28&0&100	machine.inf	Not Available			
Broadcom NetXtreme Gigabit Ethernet	Yes	NET	6.34.0.0		
2/17/2003 Broadcom	oem1.inf	Not Available			
PCI\VEN_14E4&DEV_1648&SUBSYS_02A61014&REV_02\3&1D521019&0&20	machine.inf	Not Available			
Broadcom NetXtreme Gigabit Ethernet	Yes	NET	6.34.0.0		
2/17/2003 Broadcom	oem1.inf	Not Available			
PCI\VEN_14E4&DEV_1648&SUBSYS_02A61014&REV_02\3&1D521019&0&21	machine.inf	Not Available			
PCI bus (Standard system devices)	Yes	SYSTEM	5.2.3790.0	10/1/2002	(Standard system
PCI standard host CPU bridge	Yes	SYSTEM	5.2.3790.0	10/1/2002	(Standard system devices)
PCI\VEN_1014&DEV_0302&SUBSYS_00000000&REV_05\3&300BC0BE&0&00	machine.inf	Not Available			ACPI\PNP0A03\12
QLogic QLA23xx PCI Fibre Channel Adapter	No				
SCSIADAPTER	8.2.0.0	10/22/2002	QLogic	oem3.inf	Not Available
PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&300BC0BE&0&08	machine.inf	Not Available			
Disk drive (Standard disk drives)	Yes	DISKDRIVE	5.2.3790.0	10/1/2002	(Standard disk drives)
SCSI\DISK&VEN_IBM&PROD_1742&REV_0530\4&266D7871&0&000	disk.inf	Not Available			
Disk drive (Standard disk drives)	Yes	DISKDRIVE	5.2.3790.0	10/1/2002	(Standard disk drives)
SCSI\DISK&VEN_IBM&PROD_1742&REV_0530\4&266D7871&0&001	disk.inf	Not Available			
Disk drive (Standard disk drives)	Yes	DISKDRIVE	5.2.3790.0	10/1/2002	(Standard disk drives)
SCSI\DISK&VEN_IBM&PROD_1742&REV_0530\4&266D7871&0&012	disk.inf	Not Available			
Disk drive (Standard disk drives)	Yes	DISKDRIVE	5.2.3790.0	10/1/2002	(Standard disk drives)
SCSI\DISK&VEN_IBM&PROD_1742&REV_0530\4&266D7871&0&013	disk.inf	Not Available			
Qlogic processor device	Yes	SYSTEM	5.2.3790.0	10/1/2002	
QLOGIC	scsidev.inf	Not Available			
SCSI\PROCESSOR&VEN_QLOGIC&PROD_PSEUDO_LUN&REV_4&266D7871&0&07F0	machine.inf	Not Available			
QLogic QLA23xx PCI Fibre Channel Adapter	No				
SCSIADAPTER	8.2.0.0	10/22/2002	QLogic	oem3.inf	Not Available
PCI\VEN_1077&DEV_2312&SUBSYS_010C1077&REV_02\3&300BC0BE&0&10	machine.inf	Not Available			

Qlogic processor device	Yes	SYSTEM	5.2.3790.0	10/1/2002	SCSI\PROCESSOR&VEN_QLOGIC&PROD_PSEUDO_LUN&REV_4&29C1039C&0&07F0
QLOGIC scsidev.inf	Not Available				
SCSI\PROCESSOR&VEN_QLOGIC&PROD_PSEUDO_LUN&REV_4&D88F85E&0&07F0					Motherboard resources (Standard system devices)
PCI bus	Yes	SYSTEM	5.2.3790.0	10/1/2002	machine.inf Not Available
devices)	machine.inf	Not Available	ACPI\PNP0A03\13		ACPI\PNP0C02\11
PCI standard host CPU bridge	Yes	SYSTEM	5.2.3790.0	10/1/2002	PCI bus
(Standard system devices)	machine.inf	Not Available			Yes
PCI\VEN_1014&DEV_0302&SUBSYS_00000000&REV_05\3&72AA75C&0&00					SYSTEM
QLogic QLA23xx PCI Fibre Channel Adapter	No				5.2.3790.0
SCSIADAPTER	8.2.0.0	10/22/2002	QLogic	oem3.inf	10/1/2002
Not Available					(Standard system devices)
PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&72AA75C&0&20					machine.inf
Disk drive	Yes	DISKDRIVE	5.2.3790.0	10/1/2002	Not Available
(Standard system devices)	disk.inf	Not Available			ACPI\PNP0A03\80
SCSI\DISK&VEN_IBM&PROD_1742&REV_0530\4&36847BB2&0&011					PCI standard host CPU bridge
Disk drive	Yes	DISKDRIVE	5.2.3790.0	10/1/2002	Yes
(Standard system devices)	disk.inf	Not Available			SYSTEM
SCSI\DISK&VEN_IBM&PROD_1742&REV_0530\4&36847BB2&0&012					5.2.3790.0
Disk drive	Yes	DISKDRIVE	5.2.3790.0	10/1/2002	10/1/2002
(Standard system devices)	disk.inf	Not Available			Not Available
SCSI\DISK&VEN_IBM&PROD_1742&REV_0530\4&36847BB2&0&013					(Standard system devices)
Disk drive	Yes	DISKDRIVE	5.2.3790.0	10/1/2002	machine.inf
(Standard system devices)	disk.inf	Not Available			Not Available
SCSI\DISK&VEN_IBM&PROD_1742&REV_0530\4&36847BB2&0&014					PCI\VEN_1014&DEV_0302&SUBSYS_00000000&REV_03\3&389E99D&0&00
Disk drive	Yes	DISKDRIVE	5.2.3790.0	10/1/2002	QLogic QLA23xx PCI Fibre Channel Adapter
(Standard system devices)	disk.inf	Not Available			No
SCSI\DISK&VEN_IBM&PROD_1742&REV_0530\4&36847BB2&0&015					SCSIADAPTER
Disk drive	Yes	DISKDRIVE	5.2.3790.0	10/1/2002	8.2.0.0
(Standard system devices)	disk.inf	Not Available			10/22/2002
SCSI\DISK&VEN_IBM&PROD_1742&REV_0530\4&36847BB2&0&016					QLogic
Qlogic processor device	Yes	SYSTEM	5.2.3790.0	10/1/2002	oem3.inf
QLOGIC scsidev.inf	Not Available				Not Available
SCSI\PROCESSOR&VEN_QLOGIC&PROD_PSEUDO_LUN&REV_4&36847BB2&0&07F0					PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&F2FC708&0&08
PCI bus	Yes	SYSTEM	5.2.3790.0	10/1/2002	Disk drive
devices)	machine.inf	Not Available			Yes
PCI standard host CPU bridge	Yes	SYSTEM	5.2.3790.0	10/1/2002	DISKDRIVE
(Standard system devices)	machine.inf	Not Available			5.2.3790.0
PCI\VEN_1014&DEV_0302&SUBSYS_00000000&REV_05\3&19E45801&0&00					10/1/2002
PCI bus	Yes	SYSTEM	5.2.3790.0	10/1/2002	(Standard system devices)
devices)	machine.inf	Not Available			disk.inf
PCI standard host CPU bridge	Yes	SYSTEM	5.2.3790.0	10/1/2002	Not Available
(Standard system devices)	machine.inf	Not Available			SCSI\DISK&VEN_IBM&PROD_1742&REV_0530\4&16EDCAB3&0&000
PCI\VEN_1014&DEV_0302&SUBSYS_00000000&REV_05\3&2C9E08A6&0&00					Disk drive
QLogic QLA23xx PCI Fibre Channel Adapter	No				Yes
SCSIADAPTER	8.2.0.0	10/22/2002	QLogic	oem3.inf	DISKDRIVE
Not Available					5.2.3790.0
PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&2C9E08A6&0&08					10/1/2002
Disk drive	Yes	DISKDRIVE	5.2.3790.0	10/1/2002	(Standard system devices)
(Standard system devices)	disk.inf	Not Available			disk.inf
SCSI\DISK&VEN_IBM&PROD_1742&REV_0530\4&29C1039C&0&000					Not Available
Disk drive	Yes	DISKDRIVE	5.2.3790.0	10/1/2002	SCSI\DISK&VEN_IBM&PROD_1742&REV_0530\4&16EDCAB3&0&001
(Standard system devices)	disk.inf	Not Available			Disk drive
SCSI\DISK&VEN_IBM&PROD_1742&REV_0530\4&29C1039C&0&001					Yes
Disk drive	Yes	DISKDRIVE	5.2.3790.0	10/1/2002	DISKDRIVE
(Standard system devices)	disk.inf	Not Available			5.2.3790.0
SCSI\DISK&VEN_IBM&PROD_1742&REV_0530\4&29C1039C&0&012					10/1/2002
Disk drive	Yes	DISKDRIVE	5.2.3790.0	10/1/2002	(Standard system devices)
(Standard system devices)	disk.inf	Not Available			disk.inf
SCSI\DISK&VEN_IBM&PROD_1742&REV_0530\4&29C1039C&0&013					Not Available
Qlogic processor device	Yes	SYSTEM	5.2.3790.0	10/1/2002	SCSI\DISK&VEN_IBM&PROD_1742&REV_0530\4&16EDCAB3&0&012
QLOGIC scsidev.inf	Not Available				Disk drive
SCSI\PROCESSOR&VEN_QLOGIC&PROD_PSEUDO_LUN&REV_4&1A64A01B&0&07F0					Yes
Motherboard resources	Yes	SYSTEM	5.2.3790.0	10/1/2002	DISKDRIVE
(Standard system devices)	machine.inf	Not Available			5.2.3790.0
ACPI\PNP0C02\20					10/1/2002
PCI bus	Yes	SYSTEM	5.2.3790.0	10/1/2002	(Standard system devices)
devices)	machine.inf	Not Available			disk.inf
PCI standard host CPU bridge	Yes	SYSTEM	5.2.3790.0	10/1/2002	Not Available
(Standard system devices)	machine.inf	Not Available			SCSI\DISK&VEN_IBM&PROD_1742&REV_0530\4&16EDCAB3&0&013
PCI\VEN_1014&DEV_0302&SUBSYS_00000000&REV_03\3&32219A6E&0&00					Qlogic processor device
Intel 21154 PCI to PCI bridge	Yes	SYSTEM	5.2.3790.0	10/1/2002	Yes
Intel	machine.inf	Not Available			SYSTEM
PCI\VEN_8086&DEV_B154&SUBSYS_00000000&REV_00\3&32219A6E&0&10					5.2.3790.0
Intel(R) PRO/100 S Dual Port Server Adapter	Yes	NET			10/1/2002
7.0.26.0	3/4/2003	Intel	oem6.inf	Not Available	

PCI\VEN_8086&DEV_1229&SUBSYS_10158086&REV_0D\4&18FB0D9C&0&2010						STORAGE\VOLUME\1&30A96598&0&SIGNATUREE1E6C76OFFSET6BE9F03800LENGTHA6040D600
Intel(R) PRO/100 S Dual Port Server Adapter	Yes	NET				Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
7.0.26.0 3/4/2003 Intel oem6.inf	Not Available					volume.infNot Available
PCI\VEN_8086&DEV_1229&SUBSYS_10158086&REV_0D\4&18FB0D9C&0&2810						STORAGE\VOLUME\1&30A96598&0&SIGNATUREC3D52CFAOFFSET7E0000LENGTHB79AA5200
PCI bus Yes SYSTEM 5.2.3790.0 10/1/2002 (Standard system devices) machine.inf	Not Available					Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
PCI standard host CPU bridge (Standard system devices)	Yes	SYSTEM	5.2.3790.0	10/1/2002	ACPI\PNP0A03\89	volume.infNot Available
PCI\VEN_1014&DEV_0302&SUBSYS_00000000&REV_03\3&1F67E9C9&0&00						STORAGE\VOLUME\1&30A96598&0&SIGNATUREC3D52CFAOFFSETB7A28D000LENGTH65939E800
PCI bus Yes SYSTEM 5.2.3790.0 10/1/2002 (Standard system devices) machine.inf	Not Available					Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
PCI standard host CPU bridge (Standard system devices)	Yes	SYSTEM	5.2.3790.0	10/1/2002	ACPI\PNP0A03\8A	volume.infNot Available
PCI\VEN_1014&DEV_0302&SUBSYS_00000000&REV_03\3&3ACAD158&0&00						STORAGE\VOLUME\1&30A96598&0&SIGNATUREC3D52CFBOFFSET7E0000LENGTHB79AA5200
QLogic QLA23xx PCI Fibre Channel Adapter	No					Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
SCSIADAPTER 8.2.0.0 10/22/2002 QLogic oem3.inf	Not Available					volume.infNot Available
PCI\VEN_1077&DEV_2312&SUBSYS_010C1077&REV_02\3&3ACAD158&0&08						STORAGE\VOLUME\1&30A96598&0&SIGNATUREC3D52CFBOFFSETB7A28D000LENGTH65939E800
Qlogic processor device	Yes	SYSTEM	5.2.3790.0	10/1/2002		Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
QLOGIC scsidev.inf	Not Available					volume.infNot Available
SCSI\PROCESSOR&VEN_QLOGIC&PROD_PSEUDO_LUN&REV_4&CD65BEB&0&07F0						STORAGE\VOLUME\1&30A96598&0&SIGNATUREC3D52CFBOFFSET11D3633600LENGTH6457EADA00
QLogic QLA23xx PCI Fibre Channel Adapter	No					Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
SCSIADAPTER 8.2.1.0 3/28/2003 QLogic oem11.inf	Not Available					volume.infNot Available
PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&3ACAD158&0&10						STORAGE\VOLUME\1&30A96598&0&SIGNATUREC3D52CF9OFFSET7E0000LENGTHB79AA5200
Disk drive Yes DISKDRIVE 5.2.3790.0 10/1/2002 (Standard disk drives) disk.inf	Not Available					Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
SCSI\DISK&VEN_IBM&PROD_1742&REV_0530\4&28321759&0&000						volume.infNot Available
Disk drive Yes DISKDRIVE 5.2.3790.0 10/1/2002 (Standard disk drives) disk.inf	Not Available					STORAGE\VOLUME\1&30A96598&0&SIGNATUREC3D52CF9OFFSETB7A28D000LENGTH65939E800
SCSI\DISK&VEN_IBM&PROD_1742&REV_0530\4&28321759&0&001						Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
Disk drive Yes DISKDRIVE 5.2.3790.0 10/1/2002 (Standard disk drives) disk.inf	Not Available					volume.infNot Available
SCSI\DISK&VEN_IBM&PROD_1742&REV_0530\4&28321759&0&012						STORAGE\VOLUME\1&30A96598&0&SIGNATUREC3D52CFE0FFSET7E0000LENGTHB79AA5200
Disk drive Yes DISKDRIVE 5.2.3790.0 10/1/2002 (Standard disk drives) disk.inf	Not Available					Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
SCSI\DISK&VEN_IBM&PROD_1742&REV_0530\4&28321759&0&013						volume.infNot Available
Qlogic processor device	Yes	SYSTEM	5.2.3790.0	10/1/2002		STORAGE\VOLUME\1&30A96598&0&SIGNATUREC3D52CFE0FFSETB7A28D000LENGTH65939E800
QLOGIC scsidev.inf	Not Available					Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
SCSI\PROCESSOR&VEN_QLOGIC&PROD_PSEUDO_LUN&REV_4&28321759&0&07F0						volume.infNot Available
Motherboard resources (Standard system devices)	Yes	SYSTEM	5.2.3790.0	10/1/2002		STORAGE\VOLUME\1&30A96598&0&SIGNATUREC3D52CFE0FFSET11D3633600LENGTH6457EADA00
ACPI\PNP0C02\21	machine.inf	Not Available				Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
Logical Disk Manager	Yes	SYSTEM	5.2.3790.0	10/1/2002		volume.infNot Available
ROOT\DMIO\0000	machine.inf	Not Available				STORAGE\VOLUME\1&30A96598&0&SIGNATURE2920AE86OFFSET7E0000LENGTHB79AA5200
Volume Manager	Yes	SYSTEM	5.2.3790.0	10/1/2002		Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
ROOT\FTDISK\0000	machine.inf	Not Available				volume.infNot Available
Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002		STORAGE\VOLUME\1&30A96598&0&SIGNATURE2920AE86OFFSETB7A28D000LENGTH65939E800
STORAGE\VOLUME\1&30A96598&0&SIGNATURE8A6A8A6A0FFSET7E000LENGTH8796B9400						Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002		volume.infNot Available
STORAGE\VOLUME\1&30A96598&0&SIGNATUREE1E6C76OFFSET7E0000LENGTH6BE971BA00						STORAGE\VOLUME\1&30A96598&0&SIGNATURE2920AE86OFFSET11D3633600LENGTH6457EADA00
Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002		Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
STORAGE\VOLUME\1&30A96598&0&SIGNATUREE1E6C76OFFSET7E0000LENGTH6BE971BA00						volume.infNot Available
Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002		STORAGE\VOLUME\1&30A96598&0&SIGNATURE2920AE87OFFSET7E0000LENGTHB79AA5200
STORAGE\VOLUME\1&30A96598&0&SIGNATUREE1E6C76OFFSET7E0000LENGTH6BE971BA00						Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002		volume.infNot Available



STORAGE\VOLUME\1&30A96598&0&SIGNATURE2920AEBB	Beep	Not Available	LEGACYDRIVER	Not Available
0000LENGTHB6457EADA00	Not Available	Not Available	Not Available	Not Available
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft	Available	ROOT\LEGACY_BEEP\0000	Not Available	Not Available
volume.inf Not Available	CRC Disk Filter Driver	Not Available	LEGACYDRIVER	Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURE2920AEBB	Not Available	Not Available	Not Available	Not Available
0000LENGTHB79AA5200	Available	Not Available	ROOT\LEGACY_CRCDISK\0000	Not Available
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft	dmboot	Not Available	LEGACYDRIVER	Not Available
volume.inf Not Available	Not Available	Not Available	Not Available	Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURE2920AEBB	Available	ROOT\LEGACY_DMBOOT\0000	Not Available	Not Available
0000LENGTHB79AA5200	dmload	Not Available	LEGACYDRIVER	Not Available
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft	Not Available	Not Available	Not Available	Not Available
volume.inf Not Available	Available	ROOT\LEGACY_DMLOAD\0000	Not Available	Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURE2920AEBB	em	Not Available	LEGACYDRIVER	Not Available
0000LENGTHB6457EADA00	Not Available	Not Available	Not Available	Not Available
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft	Available	ROOT\LEGACY_EM\0000	Not Available	Not Available
volume.inf Not Available	Fips	Not Available	LEGACYDRIVER	Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURE2920AEBB	Not Available	Not Available	Not Available	Not Available
0000LENGTHB79AA5200	Available	ROOT\LEGACY_FIPS\0000	Not Available	Not Available
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft	Generic Packet Classifier	Not Available	LEGACYDRIVER	Not Available
volume.inf Not Available	Not Available	Not Available	Not Available	Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURE2920AEBB	Available	Not Available	ROOT\LEGACY_GPC\0000	Not Available
0000LENGTHB79AA5200	IPSEC driver	Not Available	LEGACYDRIVER	Not Available
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft	Available	Not Available	Not Available	Not Available
volume.inf Not Available	Not Available	ROOT\LEGACY_IPSEC\0000	Not Available	Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURE2920AEBB	ksecdd	Not Available	LEGACYDRIVER	Not Available
0000LENGTHB6457EADA00	Not Available	Not Available	Not Available	Not Available
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft	Available	ROOT\LEGACY_KSECDD\0000	Not Available	Not Available
volume.inf Not Available	mnmdd	Not Available	LEGACYDRIVER	Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURE2920AEBB	Not Available	Not Available	Not Available	Not Available
0000LENGTHB79AA5200	Available	ROOT\LEGACY_MNMDD\0000	Not Available	Not Available
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft	mountmgr	Not Available	LEGACYDRIVER	Not Available
volume.inf Not Available	Not Available	Not Available	Not Available	Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURE2920AEBB	Available	ROOT\LEGACY_MOUNTMGR\0000	Not Available	Not Available
0000LENGTHB79AA5200	NDIS System Driver	Not Available	LEGACYDRIVER	Not Available
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft	Available	Not Available	Not Available	Not Available
volume.inf Not Available	Not Available	ROOT\LEGACY_NDIS\0000	Not Available	Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURE2920AEBB	Remote Access NDIS TAPI Driver	Not Available	LEGACYDRIVER	Not Available
0000LENGTHB6457EADA00	LEGACYDRIVER	Not Available	Not Available	Not Available
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft	Available	Not Available	Not Available	Not Available
volume.inf Not Available	ROOT\LEGACY_NDISTAPI\0000	Not Available	Not Available	Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURE2920AEBB	NDIS Usermode I/O Protocol	Not Available	LEGACYDRIVER	Not Available
0000LENGTHB79AA5200	Not Available	Not Available	Not Available	Not Available
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft	Available	Not Available	ROOT\LEGACY_NDISUIO\0000	Not Available
volume.inf Not Available	NDProxy	Not Available	LEGACYDRIVER	Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURE2920AEBB	Not Available	Not Available	Not Available	Not Available
0000LENGTHB79AA5200	Available	ROOT\LEGACY_NDPROXY\0000	Not Available	Not Available
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft	NetBios over Tcpip	Not Available	LEGACYDRIVER	Not Available
volume.inf Not Available	Available	Not Available	Not Available	Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURE2920AEBB	Not Available	ROOT\LEGACY_NETBT\0000	Not Available	Not Available
0000LENGTHB6457EADA00	Null	Not Available	LEGACYDRIVER	Not Available
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft	Not Available	Not Available	Not Available	Not Available
volume.inf Not Available	Available	ROOT\LEGACY_NULL\0000	Not Available	Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURE2920AEBB	Partition Manager	Not Available	LEGACYDRIVER	Not Available
0000LENGTHB79AA5200	Available	Not Available	Not Available	Not Available
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft	Not Available	ROOT\LEGACY_PARTMGR\0000	Not Available	Not Available
volume.inf Not Available	PCIVIEW	Not Available	LEGACYDRIVER	Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURE2920AEBB	Not Available	Not Available	Not Available	Not Available
0000LENGTHB79AA5200	Available	ROOT\LEGACY_PCIVIEW\0000	Not Available	Not Available
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft	qldirect	Not Available	LEGACYDRIVER	Not Available
volume.inf Not Available	Not Available	Not Available	Not Available	Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURE2920AEBB	Available	ROOT\LEGACY_QLDIRECT\0000	Not Available	Not Available
0000LENGTHB64388A5A00	qlvika	Not Available	LEGACYDRIVER	Not Available
AFD Networking Support Environment	Not Available	Not Available	Not Available	Not Available
LEGACYDRIVER	Not Available	Not Available	Not Available	Not Available
Available	Not Available	Not Available	Not Available	Not Available
ROOT\LEGACY_AFD\0000	Available	ROOT\LEGACY_QLVIKA\0000	Not Available	Not Available
	Remote Access Auto Connection Driver	Not Available	LEGACYDRIVER	Not Available
	LEGACYDRIVER	Not Available	Not Available	Not Available
	Available	Not Available	Not Available	Not Available
	ROOT\LEGACY_RASACD\0000	Not Available	Not Available	Not Available

```

RDPCCD Not Available LEGACYDRIVER Not Available
Not Available Not Available Not Available Not
Available ROOT\LEGACY_RDPCCD\0000
RDPWD Not Available LEGACYDRIVER Not Available
Not Available Not Available Not Available Not
Available ROOT\LEGACY_RDPWD\0000
TCP/IP Protocol Driver Not Available LEGACYDRIVER
Not Available Not Available Not Available Not
Available Not Available ROOT\LEGACY_TCPIP\0000
TDTCP Not Available LEGACYDRIVER Not Available
Not Available Not Available Not Available Not
Available ROOT\LEGACY_TDTCP\0000
tperf Not Available LEGACYDRIVER Not Available
Not Available Not Available Not Available Not
Available ROOT\LEGACY_TTPERF\0000
VGA Display Controller. Not Available LEGACYDRIVER
Not Available Not Available Not Available Not
Available Not Available ROOT\LEGACY_VGASAVE\0000
volsnap Not Available LEGACYDRIVER Not Available
Not Available Not Available Not Available Not
Available ROOT\LEGACY_VOLSAP\0000
Remote Access IP ARP Driver Not Available LEGACYDRIVER
Not Available Not Available Not Available Not
Available Not Available ROOT\LEGACY_WANARP\0000
Audio Codecs Yes MEDIA 5.2.3790.0 10/1/2002 (Standard
system devices) wave.inf Not Available
ROOT\MEDIA\MS_MMCM
Legacy Audio Drivers Yes MEDIA 5.2.3790.0 10/1/2002 (Standard
system devices) wave.inf Not Available
ROOT\MEDIA\MS_MMDRV
Media Control Devices Yes MEDIA 5.2.3790.0 10/1/2002
(Standard system devices) wave.inf Not Available
ROOT\MEDIA\MS_MMMCI
Legacy Video Capture Devices Yes MEDIA 5.2.3790.0 10/1/2002
(Standard system devices) wave.inf Not Available
ROOT\MEDIA\MS_MMVCD
Video Codecs Yes MEDIA 5.2.3790.0 10/1/2002 (Standard
system devices) wave.inf Not Available
ROOT\MEDIA\MS_MMVID
WAN Miniport (L2TP) Yes NET 5.2.3790.0 10/1/2002 Microsoft
netrasa.inf Not Available
ROOT\MS_L2TPMINIPORT\0000
WAN Miniport (IP) Yes NET 5.2.3790.0 10/1/2002 Microsoft
netrasa.inf Not Available
ROOT\MS_NDISWANIP\0000
WAN Miniport (PPPOE) Yes NET 5.2.3790.0 10/1/2002
Microsoft netrasa.inf Not Available
ROOT\MS_PPPOEMINIPORT\0000
WAN Miniport (PPTP) Yes NET 5.2.3790.0 10/1/2002
Microsoft netrasa.inf Not Available
ROOT\MS_PPTPMINIPORT\0000
Direct Parallel Yes NET 5.2.3790.0 10/1/2002 Microsoft
netrasa.inf Not Available
ROOT\MS_PTMINIPORT\0000
Terminal Server Device Redirector Yes SYSTEM 5.2.3790.0
10/1/2002 (Standard system devices) machine.inf Not
Available ROOT\RDPDR\0000
Terminal Server Keyboard Driver Yes SYSTEM 5.2.3790.0
10/1/2002 (Standard system devices) machine.inf Not
Available ROOT\RDP_KBD\0000
Terminal Server Mouse Driver Yes SYSTEM 5.2.3790.0 10/1/2002
(Standard system devices) machine.inf Not Available
ROOT\RDP_MOU\0000
Plug and Play Software Device Enumerator Yes SYSTEM 5.2.3790.0
10/1/2002 (Standard system devices) machine.inf Not
Available ROOT\SYSTEM\0000
Microcode Update Device Yes SYSTEM 5.2.3790.0 10/1/2002
(Standard system devices) machine.inf Not Available
ROOT\SYSTEM\0001

```

[Environment Variables]

```

Variable Value User Name
ClusterLog C:\WINDOWS\Cluster\cluster.log <SYSTEM>
ComSpec %SystemRoot%\system32\cmd.exe <SYSTEM>
NUMBER_OF_PROCESSORS 32 <SYSTEM>
OS Windows_NT <SYSTEM>
Path
%SystemRoot%\system32;%SystemRoot%;%SystemRoot%\System32\Wbem;
C:\Program Files\Microsoft SQL Server\80\Tools\BINN;C:\Dot_net_kernrate
<SYSTEM>
PATHEXT
.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH <SYSTEM>
PROCESSOR_ARCHITECTURE x86 <SYSTEM>
PROCESSOR_IDENTIFIER x86 Family 15 Model 2 Stepping 5,
GenuineIntel <SYSTEM>
PROCESSOR_LEVEL 15 <SYSTEM>
PROCESSOR_REVISION 0205 <SYSTEM>
TEMP %SystemRoot%\TEMP <SYSTEM>
TMP %SystemRoot%\TEMP <SYSTEM>
windir %SystemRoot% <SYSTEM>
TEMP %USERPROFILE%\Local Settings\Temp NT
AUTHORITY\SYSTEM
TMP %USERPROFILE%\Local Settings\Temp NT
AUTHORITY\SYSTEM
TEMP %USERPROFILE%\Local Settings\Temp NT
AUTHORITY\LOCAL SERVICE
TMP %USERPROFILE%\Local Settings\Temp NT
AUTHORITY\LOCAL SERVICE
TEMP %USERPROFILE%\Local Settings\Temp NT
AUTHORITY\NETWORK SERVICE
TMP %USERPROFILE%\Local Settings\Temp NT
AUTHORITY\NETWORK SERVICE
TEMP %USERPROFILE%\Local Settings\Temp
IBMSERV4\Administrator
TMP %USERPROFILE%\Local Settings\Temp
IBMSERV4\Administrator

```

[Print Jobs]

```

Document Size Owner Notify Status Time Submitted
Start Time Until Time Elapsed Time Pages Printed Job ID
Priority Parameters Driver Print Processor Host Print Queue
Data Type Name

```

[Network Connections]

```

Local Name Remote Name Type Status User Name
Z: \\fsserv\hdrive Disk Current Connection
IBMSERV4\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 Not Available
Not Available Not Available
system Not Available 4 8 0 1413120
Not Available Not Available Not Available Not
Available
sms.exe Not Available 548 11 204800 1413120
7/15/2003 4:15 PM Not Available Not Available Not
Available
csrss.exe Not Available 604 13 Not Available
Not Available 7/15/2003 4:18 PM Not Available Not
Available Not Available

```

winlogon.exe	c:\windows\system32\winlogon.exe	628		dmadmin.exe	c:\windows\system32\dmadmin.exe	3324	8
13	204800 1413120 7/15/2003 4:18 PM 5.2.3790.0			204800 1413120 7/15/2003 5:45 PM 5.2.3790.0			
(srv03_rtm.030324-2048)	536.50 KB (549,376 bytes)	3/25/2003		(srv03_rtm.030324-2048)	220.00 KB (225,280 bytes)	3/25/2003	
7:00 AM				7:00 AM			
services.exe	c:\windows\system32\services.exe	672	9	wmiprvse.exe	Not Available	3808	8 Not
204800 1413120 7/15/2003 4:18 PM 5.2.3790.0				Available	Not Available	7/15/2003 5:49 PM	Not Available
(srv03_rtm.030324-2048)	102.00 KB (104,448 bytes)	3/25/2003		Not Available	Not Available		
7:00 AM				helpctr.exe	c:\windows\system32\helpctr\binaries\helpctr.exe	2476	8
lsass.exe	c:\windows\system32\lsass.exe	684	9	204800 1413120 7/15/2003 5:52 PM 5.2.3790.0			
1413120 7/15/2003 4:18 PM 5.2.3790.0 (srv03_rtm.030324-2048)				(srv03_rtm.030324-2048)	764.00 KB (782,336 bytes)	4/10/2003	
13.00 KB (13,312 bytes)	3/25/2003 7:00 AM			10:26 AM			
svchost.exe	c:\windows\system32\svchost.exe	836	8	helpsvc.exe	c:\windows\system32\helpctr\binaries\helpsvc.exe		
204800 1413120 7/15/2003 4:18 PM 5.2.3790.0				2396 8 204800 1413120 7/15/2003 5:52 PM 5.2.3790.0			
(srv03_rtm.030324-2048)	13.00 KB (13,312 bytes)	3/25/2003		(srv03_rtm.030324-2048)	720.00 KB (737,280 bytes)	4/10/2003	
7:00 AM				10:27 AM			
svchost.exe	c:\windows\system32\svchost.exe	904	8	[Loaded Modules]			
204800 1413120 7/15/2003 4:18 PM 5.2.3790.0				Name	Version	Size	File Date
(srv03_rtm.030324-2048)	13.00 KB (13,312 bytes)	3/25/2003		winlogon	5.2.3790.0 (srv03_rtm.030324-2048)	536.50 KB (549,376 bytes)	3/25/2003 7:00 AM
7:00 AM				Available	Not Available	7/15/2003 4:19 PM	Not Available
svchost.exe	Not Available	1060	8 Not	Not Available	Not Available		
svchost.exe	Not Available	1132	8 Not	svchost.exe	Not Available	1132	8 Not
svchost.exe	Not Available	1172	8	Available	Not Available	7/15/2003 4:19 PM	Not Available
204800 1413120 7/15/2003 4:19 PM 5.2.3790.0				Not Available	Not Available		
(srv03_rtm.030324-2048)	13.00 KB (13,312 bytes)	3/25/2003		svchost.exe	c:\windows\system32\svchost.exe	1172	8
7:00 AM				204800 1413120 7/15/2003 4:19 PM 5.2.3790.0			
explorer.exe	c:\windows\explorer.exe	1388	8	(srv03_rtm.030324-2048)	13.00 KB (13,312 bytes)	3/25/2003	
204800 1413120 7/15/2003 4:19 PM 6.00.3790.0				7:00 AM			
(srv03_rtm.030324-2048)	1,008.50 KB (1,032,704 bytes)	3/25/2003		explorer.exe	c:\windows\explorer.exe	1388	8
7:00 AM				204800 1413120 7/15/2003 4:19 PM 6.2.35.0			
pronomgr.exe	c:\program files\intel\ncs\proset\pronomgr.exe	1456	8	84.00 KB (86,016 bytes)	3/11/2003 3:24 PM		
204800 1413120 7/15/2003 4:19 PM 6.2.35.0				msdtc.exe	Not Available	1552	8 Not Available
Not Available	7/15/2003 4:19 PM	Not Available	Not Available	Not Available	7/15/2003 4:19 PM	Not Available	Not Available
svchost.exe	Not Available	304	8 Not	Available	Not Available		
Available	Not Available	7/15/2003 4:19 PM	Not Available	svchost.exe	Not Available	304	8 Not
Not Available	Not Available			Available	Not Available	7/15/2003 4:19 PM	Not Available
wmiprvse.exe	Not Available	1384	8 Not	Not Available	Not Available		
Available	Not Available	7/15/2003 4:19 PM	Not Available	wmiprvse.exe	Not Available	1384	8 Not
Not Available	Not Available			Available	Not Available	7/15/2003 4:19 PM	Not Available
cmd.exe	c:\windows\system32\cmd.exe	2708	8	Not Available	Not Available		
1413120 7/15/2003 4:32 PM 5.2.3790.0 (srv03_rtm.030324-2048)				cmd.exe	c:\windows\system32\cmd.exe	2708	8
374.00 KB (382,976 bytes)	3/25/2003 7:00 AM			1413120 7/15/2003 4:32 PM 5.2.3790.0 (srv03_rtm.030324-2048)			
sqlservr.exe	c:\program files\microsoft sql server\mssql\bin\sqlservr.exe	3404	13	374.00 KB (382,976 bytes)	3/25/2003 7:00 AM		
2000.080.0782.00	7.11 MB (7,458,856 bytes)	6/18/2003 9:32 AM		sqlservr.exe	c:\program files\microsoft sql server\mssql\bin\sqlservr.exe	3404	13
cmd.exe	c:\windows\system32\cmd.exe	2336	8	7/15/2003 4:35 PM 2000.080.0782.00			
1413120 7/15/2003 4:38 PM 5.2.3790.0 (srv03_rtm.030324-2048)				7.11 MB (7,458,856 bytes)	4/16/2003 10:10 AM		
374.00 KB (382,976 bytes)	3/25/2003 7:00 AM			cmd.exe	c:\windows\system32\cmd.exe	2336	8
osql.exe	c:\program files\microsoft sql server\80\tools\bin\osql.exe	1428	8	1413120 7/15/2003 4:38 PM 5.2.3790.0 (srv03_rtm.030324-2048)			
2000.080.0760.00	56.55 KB (57,904 bytes)	4/16/2003 10:10 AM		374.00 KB (382,976 bytes)	3/25/2003 7:00 AM		
javaw.exe	c:\program files\ibm_fasttjre\1.3\bin\javaw.exe	2644	8	osql.exe	c:\program files\microsoft sql server\80\tools\bin\osql.exe	1428	8
204800 1413120 7/15/2003 5:34 PM Not Available		20.07 KB		2000.080.0760.00	56.55 KB (57,904 bytes)	4/16/2003 10:10 AM	
(20,549 bytes)	4/16/2003 4:37 PM			javaw.exe	c:\program files\ibm_fasttjre\1.3\bin\javaw.exe	2644	8
mmc.exe	c:\windows\system32\mmc.exe	3688	8	204800 1413120 7/15/2003 5:34 PM Not Available		20.07 KB	
1413120 7/15/2003 5:45 PM 5.2.3790.0 (srv03_rtm.030324-2048)				(20,549 bytes)	4/16/2003 4:37 PM		
762.50 KB (780,800 bytes)	3/25/2003 7:00 AM			mmc.exe	c:\windows\system32\mmc.exe	3688	8
vds.exe	c:\windows\system32\vds.exe	3316	8	1413120 7/15/2003 5:45 PM 5.2.3790.0 (srv03_rtm.030324-2048)			
1413120 7/15/2003 5:45 PM 5.2.3790.0 (srv03_rtm.030324-2048)				762.50 KB (780,800 bytes)	3/25/2003 7:00 AM		
157.00 KB (160,768 bytes)	3/25/2003 7:00 AM			vds.exe	c:\windows\system32\vds.exe	3316	8
				1413120 7/15/2003 5:45 PM 5.2.3790.0 (srv03_rtm.030324-2048)			
				157.00 KB (160,768 bytes)	3/25/2003 7:00 AM		

regapi	5.2.3790.0 (srv03_rtm.030324-2048)	48.50 KB (49,664 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\regapi.dll
ws2_32	5.2.3790.0 (srv03_rtm.030324-2048)	87.50 KB (89,600 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\ws2_32.dll
ws2help	5.2.3790.0 (srv03_rtm.030324-2048)	19.50 KB (19,968 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\ws2help.dll
psapi	5.2.3790.0 (srv03_rtm.030324-2048)	21.50 KB (22,016 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\psapi.dll
version	5.2.3790.0 (srv03_rtm.030324-2048)	17.00 KB (17,408 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\version.dll
setupapi	5.2.3790.0 (srv03_rtm.030324-2048)	1,014.50 KB (1,038,848 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\setupapi.dll
msgina	5.2.3790.0 (srv03_rtm.030324-2048)	1.14 MB (1,191,936 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\msgina.dll
shsvcs	6.00.3790.0 (srv03_rtm.030324-2048)	121.50 KB (124,416 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\shsvcs.dll
shlwapi	6.00.3790.0 (srv03_rtm.030324-2048)	281.00 KB (287,744 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\shlwapi.dll
sfc	5.2.3790.0 (srv03_rtm.030324-2048)	4.50 KB (4,608 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\sfc.dll
sfc_os	5.2.3790.0 (srv03_rtm.030324-2048)	133.00 KB (136,192 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\sfc_os.dll
wintrust	5.131.3790.0 (srv03_rtm.030324-2048)	161.50 KB (165,376 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\wintrust.dll
ole32	5.2.3790.0 (srv03_rtm.030324-2048)	1.13 MB (1,187,328 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\ole32.dll
imagehlp	5.2.3790.0 (srv03_rtm.030324-2048)	142.50 KB (145,920 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\imagehlp.dll
comctl32	6.0 (srv03_rtm.030324-2048)	907.00 KB (928,768 bytes)	4/10/2003 5:01 AM	Microsoft Corporation	c:\windows\winsxs\x86_microsoft.windows.common-controls_6595b64144ccf1df_6.0.100.0_x-ww_8417450b\comctl32.dll
winscard	5.2.3790.0 (srv03_rtm.030324-2048)	98.50 KB (100,864 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\winscard.dll
wtsapi32	5.2.3790.0 (srv03_rtm.030324-2048)	17.50 KB (17,920 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\wtsapi32.dll
winmm	5.2.3790.0 (srv03_rtm.030324-2048)	166.00 KB (169,984 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\winmm.dll
sxs	5.2.3790.0 (srv03_rtm.030324-2048)	733.00 KB (750,592 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\sxs.dll
shell32	6.00.3790.0 (srv03_rtm.030324-2048)	7.79 MB (8,166,400 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\shell32.dll
wldap32	5.2.3790.0 (srv03_rtm.030324-2048)	158.00 KB (161,792 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\wldap32.dll
rsaenh	5.2.3790.0 (srv03_rtm.030324-2048)	176.83 KB (181,072 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\rsaenh.dll
cscdll	5.2.3790.0 (srv03_rtm.030324-2048)	99.00 KB (101,376 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\cscdll.dll
wlnotify	5.2.3790.0 (srv03_rtm.030324-2048)	87.50 KB (89,600 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\wlnotify.dll
winspool	5.2.3790.0 (srv03_rtm.030324-2048)	140.00 KB (143,360 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\winspool.drv
mpr	5.2.3790.0 (srv03_rtm.030324-2048)	56.00 KB (57,344 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\mpr.dll
comctl32	5.82 (srv03_rtm.030324-2048)	561.00 KB (574,464 bytes)	4/10/2003 5:01 AM	Microsoft Corporation	c:\windows\winsxs\x86_microsoft.windows.common-controls_6595b64144ccf1df_5.82.0.0_x-ww_8a69ba05\comctl32.dll
uxtheme	6.00.3790.0 (srv03_rtm.030324-2048)	196.00 KB (200,704 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\uxtheme.dll
mprapi	5.2.3790.0 (srv03_rtm.030324-2048)	81.00 KB (82,944 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\mprapi.dll
activeds	5.2.3790.0 (srv03_rtm.030324-2048)	189.00 KB (193,536 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\activeds.dll
adslidpc	5.2.3790.0 (srv03_rtm.030324-2048)	142.50 KB (145,920 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\adslidpc.dll
credui	5.2.3790.0 (srv03_rtm.030324-2048)	159.00 KB (162,816 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\credui.dll
atl	3.05.2283	83.00 KB (84,992 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\atl.dll
oleaut32	5.2.3790.0	486.00 KB (497,664 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\oleaut32.dll
rtutils	5.2.3790.0 (srv03_rtm.030324-2048)	32.00 KB (32,768 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\rtutils.dll
samlib	5.2.3790.0 (srv03_rtm.030324-2048)	49.00 KB (50,176 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\samlib.dll
cscui	5.2.3790.0 (srv03_rtm.030324-2048)	305.00 KB (312,320 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\cscui.dll
ntmarta	5.2.3790.0 (srv03_rtm.030324-2048)	114.00 KB (116,736 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\ntmarta.dll
clbcatq	2001.12.4720.0 (srv03_rtm.030324-2048)	481.00 KB (492,544 bytes)	4/10/2003 10:23 AM	Microsoft Corporation	c:\windows\system32\clbcatq.dll
comres	2001.12.4720.0 (srv03_rtm.030324-2048)	778.00 KB (796,672 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\comres.dll
wbemprox	5.2.3790.0 (srv03_rtm.030324-2048)	17.50 KB (17,920 bytes)	4/10/2003 10:23 AM	Microsoft Corporation	c:\windows\system32\wbem\wbemprox.dll
wbemcomn	5.2.3790.0 (srv03_rtm.030324-2048)	211.50 KB (216,576 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\wbem\wbemcomn.dll
wbemsvc	5.2.3790.0 (srv03_rtm.030324-2048)	42.50 KB (43,520 bytes)	4/10/2003 10:23 AM	Microsoft Corporation	c:\windows\system32\wbem\wbemsvc.dll
fastprox	5.2.3790.0 (srv03_rtm.030324-2048)	443.00 KB (453,632 bytes)	4/10/2003 10:23 AM	Microsoft Corporation	c:\windows\system32\wbem\fastprox.dll
msvcpx60	6.05.2144.0	388.00 KB (397,312 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\msvcpx60.dll



ntdsapi	5.2.3790.0 (srv03_rtm.030324-2048)	76.00 KB (77,824 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\ntdsapi.dll
dnsapi	5.2.3790.0 (srv03_rtm.030324-2048)	147.50 KB (151,040 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\dnsapi.dll
services	5.2.3790.0 (srv03_rtm.030324-2048)	102.00 KB (104,448 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\services.exe
scesrv	5.2.3790.0 (srv03_rtm.030324-2048)	316.50 KB (324,096 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\scesrv.dll
authz	5.2.3790.0 (srv03_rtm.030324-2048)	67.00 KB (68,608 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\authz.dll
umpnpgmr	5.2.3790.0 (srv03_rtm.030324-2048)	121.50 KB (124,416 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\umpnpgmr.dll
ncobjapi	5.2.3790.0 (srv03_rtm.030324-2048)	34.50 KB (35,328 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\ncobjapi.dll
eventlog	5.2.3790.0 (srv03_rtm.030324-2048)	60.50 KB (61,952 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\eventlog.dll
lsass	5.2.3790.0 (srv03_rtm.030324-2048)	13.00 KB (13,312 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\lsass.exe
lsasrv	5.2.3790.0 (srv03_rtm.030324-2048)	780.50 KB (799,232 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\lsasrv.dll
samsrv	5.2.3790.0 (srv03_rtm.030324-2048)	452.00 KB (462,848 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\samsrv.dll
cryptdll	5.2.3790.0 (srv03_rtm.030324-2048)	34.00 KB (34,816 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\cryptdll.dll
msprivs	5.2.3790.0 (srv03_rtm.030324-2048)	46.50 KB (47,616 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\msprivs.dll
kerberos	5.2.3790.0 (srv03_rtm.030324-2048)	332.50 KB (340,480 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\kerberos.dll
msv1_0	5.2.3790.0 (srv03_rtm.030324-2048)	127.00 KB (130,048 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\msv1_0.dll
netlogon	5.2.3790.0 (srv03_rtm.030324-2048)	409.00 KB (418,816 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\netlogon.dll
w32time	5.2.3790.0 (srv03_rtm.030324-2048)	216.00 KB (221,184 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\w32time.dll
iphlpapi	5.2.3790.0 (srv03_rtm.030324-2048)	82.50 KB (84,480 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\iphlpapi.dll
schannel	5.2.3790.0 (srv03_rtm.030324-2048)	149.50 KB (153,088 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\schannel.dll
wdigest	5.2.3790.0 (srv03_rtm.030324-2048)	61.00 KB (62,464 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\wdigest.dll
rassfm	5.2.3790.0 (srv03_rtm.030324-2048)	20.50 KB (20,992 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\rassfm.dll
kdcsvc	5.2.3790.0 (srv03_rtm.030324-2048)	221.00 KB (226,304 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\kdcsvc.dll
ntdsa	5.2.3790.0 (srv03_rtm.030324-2048)	1.45 MB (1,520,640 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\ntdsa.dll
ntdsatq	5.2.3790.0 (srv03_rtm.030324-2048)	32.00 KB (32,768 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\ntdsatq.dll
mswsock	5.2.3790.0 (srv03_rtm.030324-2048)	254.00 KB (260,096 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\mswsock.dll
esent	5.2.3790.0 (srv03_rtm.030324-2048)	1.01 MB (1,056,256 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\esent.dll
scecli	5.2.3790.0 (srv03_rtm.030324-2048)	179.50 KB (183,808 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\scecli.dll
wshtcpip	5.2.3790.0 (srv03_rtm.030324-2048)	18.00 KB (18,432 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\wshtcpip.dll
ipsecsvc	5.2.3790.0 (srv03_rtm.030324-2048)	162.50 KB (166,400 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\ipsecsvc.dll
oakley	5.2.3790.0 (srv03_rtm.030324-2048)	325.50 KB (333,312 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\oakley.dll
winipsec	5.2.3790.0 (srv03_rtm.030324-2048)	34.50 KB (35,328 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\winipsec.dll
dsenh	5.2.3790.0 (srv03_rtm.030324-2048)	131.33 KB (134,480 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\dsenh.dll
wlbcctl	5.2.3790.0 (srv03_rtm.030324-2048)	78.00 KB (79,872 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\wlbcctl.dll
svchost	5.2.3790.0 (srv03_rtm.030324-2048)	13.00 KB (13,312 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\svchost.exe
rpsvc	5.2.3790.0 (srv03_rtm.030324-2048)	276.50 KB (283,136 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\rpsvc.dll
termsrv	5.2.3790.0 (srv03_rtm.030324-2048)	216.50 KB (221,696 bytes)	4/10/2003 10:23 AM	Microsoft Corporation	c:\windows\system32\termsrv.dll
icaapi	5.2.3790.0 (srv03_rtm.030324-2048)	10.50 KB (10,752 bytes)	4/10/2003 10:23 AM	Microsoft Corporation	c:\windows\system32\icaapi.dll
mstlsapi	5.2.3790.0 (srv03_rtm.030324-2048)	104.50 KB (107,008 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\mstlsapi.dll
schedsvc	5.2.3790.0 (srv03_rtm.030324-2048)	176.00 KB (180,224 bytes)	4/10/2003 10:26 AM	Microsoft Corporation	c:\windows\system32\schedsvc.dll
msidle	6.00.3790.0 (srv03_rtm.030324-2048)	5.50 KB (5,632 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\msidle.dll
wkssvc	5.2.3790.0 (srv03_rtm.030324-2048)	125.00 KB (128,000 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\wkssvc.dll
wiarpc	5.2.3790.0 (srv03_rtm.030324-2048)	30.00 KB (30,720 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\wiarpc.dll
cryptsvc	5.2.3790.0 (srv03_rtm.030324-2048)	51.00 KB (52,224 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\cryptsvc.dll
certcli	5.2.3790.0 (srv03_rtm.030324-2048)	228.00 KB (233,472 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\certcli.dll

vssapi	5.2.3790.0 (srv03_rtm.030324-2048)	528.00 KB (540,672 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\vssapi.dll
dmserver	5.2.3790.0 (srv03_rtm.030324-2048)	24.00 KB (24,576 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\dmserver.dll
es	2001.12.4720.0 (srv03_rtm.030324-2048)	221.50 KB (226,816 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\es.dll
srvsvc	5.2.3790.0 (srv03_rtm.030324-2048)	89.00 KB (91,136 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\srvsvc.dll
seclogon	5.2.3790.0 (srv03_rtm.030324-2048)	16.50 KB (16,896 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\seclogon.dll
wmisvc	5.2.3790.0 (srv03_rtm.030324-2048)	131.00 KB (134,144 bytes)	4/10/2003 10:23 AM	Microsoft Corporation	c:\windows\system32\wmisvc.dll
sens	5.2.3790.0 (srv03_rtm.030324-2048)	35.50 KB (36,352 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\sens.dll
comsvcs	2001.12.4720.0 (srv03_rtm.030324-2048)	1.14 MB (1,199,616 bytes)	4/10/2003 10:23 AM	Microsoft Corporation	c:\windows\system32\comsvcs.dll
netman	5.2.3790.0 (srv03_rtm.030324-2048)	209.00 KB (214,016 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\netman.dll
rasapi32	5.2.3790.0 (srv03_rtm.030324-2048)	227.50 KB (232,960 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\rasapi32.dll
rasman	5.2.3790.0 (srv03_rtm.030324-2048)	56.50 KB (57,856 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\rasman.dll
tapi32	5.2.3790.0 (srv03_rtm.030324-2048)	175.00 KB (179,200 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\tapi32.dll
wzcsvc	5.2.3790.0 (srv03_rtm.030324-2048)	272.50 KB (279,040 bytes)	3/25/2003 7:15 AM	Microsoft Corporation	c:\windows\system32\wzcsvc.dll
wmi	5.2.3790.0 (srv03_rtm.030324-2048)	6.50 KB (6,656 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\wmi.dll
dhcpcsvc	5.2.3790.0 (srv03_rtm.030324-2048)	101.50 KB (103,936 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\dhcpcsvc.dll
wzcsapi	5.2.3790.0 (srv03_rtm.030324-2048)	24.50 KB (25,088 bytes)	3/25/2003 7:15 AM	Microsoft Corporation	c:\windows\system32\wzcsapi.dll
netshell	5.2.3790.0 (srv03_rtm.030324-2048)	1.67 MB (1,747,456 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\netshell.dll
clusapi	5.2.3790.0 (srv03_rtm.030324-2048)	56.00 KB (57,344 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\clusapi.dll
hnetcfg	5.2.3790.0 (srv03_rtm.030324-2048)	243.50 KB (249,344 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\hnetcfg.dll
wininet	6.00.3790.0 (srv03_rtm.030324-2048)	609.00 KB (623,616 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\wininet.dll
wbemcore	5.2.3790.0 (srv03_rtm.030324-2048)	457.00 KB (467,968 bytes)	4/10/2003 10:23 AM	Microsoft Corporation	c:\windows\system32\wbem\wbemcore.dll
esscli	5.2.3790.0 (srv03_rtm.030324-2048)	235.50 KB (241,152 bytes)	4/10/2003 10:23 AM	Microsoft Corporation	c:\windows\system32\wbem\esscli.dll
wmiutils	5.2.3790.0 (srv03_rtm.030324-2048)	90.50 KB (92,672 bytes)	4/10/2003 10:23 AM	Microsoft Corporation	c:\windows\system32\wbem\wmiutils.dll
repdrvfs	5.2.3790.0 (srv03_rtm.030324-2048)	165.00 KB (168,960 bytes)	4/10/2003 10:23 AM	Microsoft Corporation	c:\windows\system32\wbem\repdrvfs.dll
wmiprvsd	5.2.3790.0 (srv03_rtm.030324-2048)	405.50 KB (415,232 bytes)	4/10/2003 10:23 AM	Microsoft Corporation	c:\windows\system32\wbem\wmiprvsd.dll
wbemess	5.2.3790.0 (srv03_rtm.030324-2048)	256.50 KB (262,656 bytes)	4/10/2003 10:23 AM	Microsoft Corporation	c:\windows\system32\wbem\wbemess.dll
rasdlg	5.2.3790.0 (srv03_rtm.030324-2048)	642.00 KB (657,408 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\rasdlg.dll
ncprov	5.2.3790.0 (srv03_rtm.030324-2048)	43.00 KB (44,032 bytes)	4/10/2003 10:23 AM	Microsoft Corporation	c:\windows\system32\wbem\ncprov.dll
rasadhlp	5.2.3790.0 (srv03_rtm.030324-2048)	6.50 KB (6,656 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\rasadhlp.dll
netcfgx	5.2.3790.0 (srv03_rtm.030324-2048)	726.00 KB (743,424 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\netcfgx.dll
wbemcons	5.2.3790.0 (srv03_rtm.030324-2048)	69.00 KB (70,656 bytes)	4/10/2003 10:23 AM	Microsoft Corporation	c:\windows\system32\wbem\wbemcons.dll
pchsvc	5.2.3790.0 (srv03_rtm.030324-2048)	31.50 KB (32,256 bytes)	4/10/2003 10:27 AM	Microsoft Corporation	c:\windows\pchealth\helpctr\binaries\pchsvc.dll
explorer	6.00.3790.0 (srv03_rtm.030324-2048)	1,008.50 KB (1,032,704 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\explorer.exe
browseui	6.00.3790.0 (srv03_rtm.030324-2048)	1.01 MB (1,057,280 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\browseui.dll
shdocvw	6.00.3790.0 (srv03_rtm.030324-2048)	1.33 MB (1,393,664 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\shdocvw.dll
apphelp	5.2.3790.0 (srv03_rtm.030324-2048)	122.00 KB (124,928 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\apphelp.dll
themeui	6.00.3790.0 (srv03_rtm.030324-2048)	360.50 KB (369,152 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\themeui.dll
msimg32	5.2.3790.0 (srv03_rtm.030324-2048)	4.50 KB (4,608 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\msimg32.dll
linkinfo	5.2.3790.0 (srv03_rtm.030324-2048)	16.50 KB (16,896 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\linkinfo.dll
ntshrui	6.00.3790.0 (srv03_rtm.030324-2048)	136.00 KB (139,264 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\ntshrui.dll
urlmon	6.00.3790.0 (srv03_rtm.030324-2048)	501.50 KB (513,536 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\urlmon.dll
webcheck	6.00.3790.0 (srv03_rtm.030324-2048)	261.50 KB (267,776 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\webcheck.dll
wsock32	5.2.3790.0 (srv03_rtm.030324-2048)	22.00 KB (22,528 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\wsock32.dll
stobject	5.2.3790.0 (srv03_rtm.030324-2048)	117.50 KB (120,320 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\stobject.dll

batmeter	6.00.3790.0 (srv03_rtm.030324-2048)	28.50 KB (29,184 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\batmeter.dll
powrprof	6.00.3790.0 (srv03_rtm.030324-2048)	14.50 KB (14,848 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\powrprof.dll
printui	5.2.3790.0 (srv03_rtm.030324-2048)	536.50 KB (549,376 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\printui.dll
cfgmgr32	5.2.3790.0 (srv03_rtm.030324-2048)	17.50 KB (17,920 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\cfgmgr32.dll
drprov	5.2.3790.0 (srv03_rtm.030324-2048)	12.50 KB (12,800 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\drprov.dll
ntlanman	5.2.3790.0 (srv03_rtm.030324-2048)	41.00 KB (41,984 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\ntlanman.dll
netui0	5.2.3790.0 (srv03_rtm.030324-2048)	75.50 KB (77,312 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\netui0.dll
netui1	5.2.3790.0 (srv03_rtm.030324-2048)	184.00 KB (188,416 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\netui1.dll
davclnt	5.2.3790.0 (srv03_rtm.030324-2048)	23.50 KB (24,064 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\davclnt.dll
browsec	6.00.3790.0 (srv03_rtm.030324-2048)	62.00 KB (63,488 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\browsec.dll
shdoclc	6.00.3790.0 (srv03_rtm.030324-2048)	588.50 KB (602,624 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\shdoclc.dll
shimgvw	6.00.3790.0 (srv03_rtm.030324-2048)	418.00 KB (428,032 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\shimgvw.dll
gdipplus	5.2.3790.0 (srv03_rtm.030324-2048)	1.67 MB (1,748,992 bytes)	4/10/2003 5:01 AM	Microsoft Corporation	c:\windows\winsxs\x86_microsoft.windows.gdipplus_6595b64144ccfd1_1.0.10.0.0_x-ww_0d1f9f94\gdipplus.dll
mydocs	6.00.3790.0 (srv03_rtm.030324-2048)	88.00 KB (90,112 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\mydocs.dll
mprui	5.2.3790.0 (srv03_rtm.030324-2048)	49.00 KB (50,176 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\mprui.dll
netui2	5.2.3790.0 (srv03_rtm.030324-2048)	309.50 KB (316,928 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\netui2.dll
comdlg32	6.00.3790.0 (srv03_rtm.030324-2048)	261.00 KB (267,264 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\comdlg32.dll
netmsg	5.2.3790.0 (srv03_rtm.030324-2048)	178.00 KB (182,272 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\netmsg.dll
netplwiz	5.2.3790.0 (srv03_rtm.030324-2048)	843.00 KB (863,232 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\netplwiz.dll
actxprxy	6.00.3790.0 (srv03_rtm.030324-2048)	95.00 KB (97,280 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\actxprxy.dll
pronomgr	6.2.35.0	84.00 KB (86,016 bytes)	3/11/2003 3:24 PM	Intel(R) Corporation	c:\program files\intel\ncs\proset\pronomgr.exe
enupguir	6.2.35.0	340.00 KB (348,160 bytes)	3/11/2003 3:15 PM	Intel(R) Corporation	c:\program files\intel\ncs\proset\enupguir.dll
pnc802_3	6.2.35.0	84.00 KB (86,016 bytes)	3/11/2003 3:33 PM	Intel(R) Corporation	c:\program files\intel\ncs\proset\8023\pnc802_3.dll
enupcmrs	6.2.35.0	344.07 KB (352,325 bytes)	3/11/2003 3:20 PM	Intel(R) Corporation	c:\program files\intel\ncs\proset\8023\enupcmrs.dll
cmd	5.2.3790.0 (srv03_rtm.030324-2048)	374.00 KB (382,976 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\cmd.exe
sqlservr	2000.080.0782.00	7.11 MB (7,458,856 bytes)	6/18/2003 9:32 AM	Microsoft Corporation	c:\program files\microsoft sql server\mssql\binn\sqlservr.exe
opends60	2000.080.0194.00	24.06 KB (24,639 bytes)	4/25/2003 4:20 PM	Microsoft Corporation	c:\program files\microsoft sql server\mssql\binn\opends60.dll
ums	2000.080.0760.00	52.55 KB (53,808 bytes)	4/25/2003 4:20 PM	Microsoft Corporation	c:\program files\microsoft sql server\mssql\binn\ums.dll
sqlsort	2000.080.0760.00	576.56 KB (590,396 bytes)	4/25/2003 4:20 PM	Microsoft Corporation	c:\program files\microsoft sql server\mssql\binn\sqlsort.dll
msvcirt	7.0.3790.0 (srv03_rtm.030324-2048)	50.00 KB (51,200 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\msvcirt.dll
sqllevn70	2000.080.0760.00	28.00 KB (28,672 bytes)	4/25/2003 4:20 PM	Microsoft Corporation	c:\program files\microsoft sql server\mssql\binn\resources\1033\sqllevn70.rll
xolehlp	2001.12.4720.0 (srv03_rtm.030324-2048)	8.50 KB (8,704 bytes)	4/10/2003 10:23 AM	Microsoft Corporation	c:\windows\system32\xolehlp.dll
msdteprx	2001.12.4720.0 (srv03_rtm.030324-2048)	427.50 KB (437,760 bytes)	4/10/2003 10:23 AM	Microsoft Corporation	c:\windows\system32\msdteprx.dll
mtxclu	2001.12.4720.0 (srv03_rtm.030324-2048)	74.50 KB (76,288 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\mtxclu.dll
resutils	5.2.3790.0 (srv03_rtm.030324-2048)	59.00 KB (60,416 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\resutils.dll
mfc42u	6.05.3014.0	960.00 KB (983,040 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\mfc42u.dll
winnr	5.2.3790.0 (srv03_rtm.030324-2048)	15.00 KB (15,360 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\winnr.dll
ssnetlib	2000.080.0766.00	80.56 KB (82,492 bytes)	4/25/2003 4:20 PM	Microsoft Corporation	c:\program files\microsoft sql server\mssql\binn\ssnetlib.dll
ssnmpn70	2000.080.0534.00	24.56 KB (25,148 bytes)	4/25/2003 4:20 PM	Microsoft Corporation	c:\program files\microsoft sql server\mssql\binn\ssnmpn70.dll
security	5.2.3790.0 (srv03_rtm.030324-2048)	5.50 KB (5,632 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\security.dll
ssmslpcn	2000.080.0760.00	28.56 KB (29,244 bytes)	4/25/2003 4:20 PM	Microsoft Corporation	c:\program files\microsoft sql server\mssql\binn\ssmslpcn.dll
osql	2000.080.0760.00	56.55 KB (57,904 bytes)	4/16/2003 10:10 AM	Microsoft Corporation	c:\program files\microsoft sql server\80\tools\bin\osql.exe
odbc32	3.525.1022.0 (srv03_rtm.030324-2048)	232.00 KB (237,568 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\odbc32.dll
odbcint	3.525.1022.0 (srv03_rtm.030324-2048)	92.00 KB (94,208 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\odbcint.dll
sqlsrv32	2000.085.1022.0 (srv03_rtm.030324-2048)	404.00 KB (413,696 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\sqlsrv32.dll
sqlunirl	2000.080.0728.00	176.56 KB (180,800 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\sqlunirl.dll

sqlsrv32	2000.085.1022.00 (srv03_rtm.030324-2048)	88.00 KB (90,112 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\sqlsrv32.rll
odbccp32	3.525.1022.0 (srv03_rtm.030324-2048)	100.00 KB (102,400 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\odbccp32.dll
dbnetlib	2000.085.1022 (srv03_rtm.030324-2048)	76.00 KB (77,824 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\dbnetlib.dll
dbmslpcn	2000.080.0760.00	28.56 KB (29,244 bytes)	4/16/2003 10:09 AM	Microsoft Corporation	c:\windows\system32\dbmslpcn.dll
javaw	Not Available	20.07 KB (20,549 bytes)	4/16/2003 4:37 PM	Not Available	c:\program files\ibm_fasttjre\1.3\bin\javaw.exe
jvm	Not Available	756.00 KB (774,144 bytes)	4/16/2003 4:37 PM	Not Available	c:\program files\ibm_fasttjre\1.3\bin\hotspot\jvm.dll
hpi	Not Available	28.08 KB (28,753 bytes)	4/16/2003 4:37 PM	Not Available	c:\program files\ibm_fasttjre\1.3\bin\hpi.dll
verify	Not Available	52.07 KB (53,319 bytes)	4/16/2003 4:37 PM	Not Available	c:\program files\ibm_fasttjre\1.3\bin\verify.dll
java	Not Available	84.08 KB (86,093 bytes)	4/16/2003 4:37 PM	Not Available	c:\program files\ibm_fasttjre\1.3\bin\java.dll
zip	Not Available	52.08 KB (53,326 bytes)	4/16/2003 4:37 PM	Not Available	c:\program files\ibm_fasttjre\1.3\bin\zip.dll
awt	Not Available	984.07 KB (1,007,688 bytes)	4/16/2003 4:37 PM	Not Available	c:\program files\ibm_fasttjre\1.3\bin\awt.dll
imm32	5.2.3790.0 (srv03_rtm.030324-2048)	105.50 KB (108,032 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\imm32.dll
fontmanager	Not Available	236.09 KB (241,757 bytes)	4/16/2003 4:37 PM	Not Available	c:\program files\ibm_fasttjre\1.3\bin\fontmanager.dll
ddraw	5.2.3790.0 (srv03_rtm.030324-2048)	250.00 KB (256,000 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\ddraw.dll
dciman32	5.2.3790.0 (srv03_rtm.030324-2048)	8.50 KB (8,704 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\dciman32.dll
atio2kad	4.00.0110.20	1.04 MB (1,085,440 bytes)	11/22/2001 9:40 AM	ATI Technologies Inc.	c:\windows\system32\atio2kad.dll
net	Not Available	32.07 KB (32,841 bytes)	4/16/2003 4:37 PM	Not Available	c:\program files\ibm_fasttjre\1.3\bin\net.dll
mmc	5.2.3790.0 (srv03_rtm.030324-2048)	762.50 KB (780,800 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\mmc.exe
oleacc	4.2.5406.0 (srv03_rtm.030324-2048)	171.00 KB (175,104 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\oleacc.dll
mmcbase	5.2.3790.0 (srv03_rtm.030324-2048)	70.50 KB (72,192 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\mmcbase.dll
mmcmdmgr	5.2.3790.0 (srv03_rtm.030324-2048)	1.13 MB (1,182,720 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\mmcmdmgr.dll
msxml3	8.40.9419.0	1.28 MB (1,337,344 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\msxml3.dll
mycomput	5.2.3790.0 (srv03_rtm.030324-2048)	96.50 KB (98,816 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\mycomput.dll
ntmsmgr	5.2.3790.0 (srv03_rtm.030324-2048)	495.50 KB (507,392 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\ntmsmgr.dll
ntmsapi	5.2.3790.0 (srv03_rtm.030324-2048)	42.50 KB (43,520 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\ntmsapi.dll
dfrgsnap	5.2.3790.0 (srv03_rtm.030324-2048)	36.00 KB (36,864 bytes)	3/25/2003 7:00 AM	Microsoft Corp. and Executive Software International, Inc.	c:\windows\system32\dfrgsnap.dll
dfrgres	5.2.3790.0 (srv03_rtm.030324-2048)	50.50 KB (51,712 bytes)	3/25/2003 7:00 AM	Microsoft Corp. and Executive Software International, Inc.	c:\windows\system32\dfrgres.dll
dmdskmgr	5.2.3790.0 (srv03_rtm.030324-2048)	164.00 KB (167,936 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\dmdskmgr.dll
dmutil	5.2.3790.0 (srv03_rtm.030324-2048)	51.50 KB (52,736 bytes)	3/24/2003 8:48 PM	Microsoft Corporation	c:\windows\system32\dmutil.dll
dmskres	5.2.3790.0 (srv03_rtm.030324-2048)	115.50 KB (118,272 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\dmskres.dll
els	5.2.3790.0 (srv03_rtm.030324-2048)	178.00 KB (182,272 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\els.dll
riched32	5.2.3790.0 (srv03_rtm.030324-2048)	3.50 KB (3,584 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\riched32.dll
riched20	5.31.23.1218	406.00 KB (415,744 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\riched20.dll
filegmt	5.2.3790.0 (srv03_rtm.030324-2048)	327.50 KB (335,360 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\filegmt.dll
localsec	5.2.3790.0 (srv03_rtm.030324-2048)	223.50 KB (228,864 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\localsec.dll
adsnt	5.2.3790.0 (srv03_rtm.030324-2048)	260.00 KB (266,240 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\adsnt.dll
smlogcfg	5.2.3790.0 (srv03_rtm.030324-2048)	360.50 KB (369,152 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\smlogcfg.dll
pdh	5.2.3790.0 (srv03_rtm.030324-2048)	274.50 KB (281,088 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\pdh.dll
odbcbcp	2000.085.1022.00 (srv03_rtm.030324-2048)	24.00 KB (24,576 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\odbcbcp.dll
devmgr	5.2.3790.0 (srv03_rtm.030324-2048)	276.50 KB (283,136 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\devmgr.dll
mlang	6.00.3790.0 (srv03_rtm.030324-2048)	570.00 KB (583,680 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\mlang.dll
dmdlgs	5.2.3790.0 (srv03_rtm.030324-2048)	272.50 KB (279,040 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\dmdlgs.dll
dmview	5.2.3790.0 (srv03_rtm.030324-2048)	61.50 KB (62,976 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\dmview.ocx
vds_ps	5.2.3790.0 (srv03_rtm.030324-2048)	21.50 KB (22,016 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\vds_ps.dll
dmvdsitf	5.2.3790.0 (srv03_rtm.030324-2048)	92.50 KB (94,720 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\dmvdsitf.dll
vds	5.2.3790.0 (srv03_rtm.030324-2048)	157.00 KB (160,768 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\vds.exe
osuninst	5.2.3790.0 (srv03_rtm.030324-2048)	115.50 KB (118,272 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\osuninst.dll
cabinet	5.2.3790.0 (srv03_rtm.030324-2048)	61.00 KB (62,464 bytes)	3/25/2003 7:00 AM	Microsoft Corporation	c:\windows\system32\cabinet.dll

vdsutil	5.2.3790.0 (srv03_rtm.030324-2048)	38.50 KB (39,424 bytes)	3/25/2003 7:00 AM	Microsoft Corporation
c:\windows\system32\vdsutil.dll				
vdsbas	5.2.3790.0 (srv03_rtm.030324-2048)	103.50 KB (105,984 bytes)	3/25/2003 7:00 AM	Microsoft Corporation
c:\windows\system32\vdsbas.dll				
fmifs	5.2.3790.0 (srv03_rtm.030324-2048)	17.00 KB (17,408 bytes)	3/25/2003 7:00 AM	Microsoft Corporation
c:\windows\system32\fmifs.dll				
ulib	5.2.3790.0 (srv03_rtm.030324-2048)	265.50 KB (271,872 bytes)	3/25/2003 7:00 AM	Microsoft Corporation
c:\windows\system32\ulib.dll				
ifsutil	5.2.3790.0 (srv03_rtm.030324-2048)	72.00 KB (73,728 bytes)	3/25/2003 7:00 AM	Microsoft Corporation
c:\windows\system32\ifsutil.dll				
vdscydr	5.2.3790.0 (srv03_rtm.030324-2048)	307.50 KB (314,880 bytes)	3/25/2003 7:00 AM	Microsoft Corporation
c:\windows\system32\vdscydr.dll				
dmintf	5.2.3790.0 (srv03_rtm.030324-2048)	20.00 KB (20,480 bytes)	3/25/2003 7:00 AM	Microsoft Corporation
c:\windows\system32\dmintf.dll				
dmadmin	5.2.3790.0 (srv03_rtm.030324-2048)	220.00 KB (225,280 bytes)	3/25/2003 7:00 AM	Microsoft Corporation
c:\windows\system32\dmadmin.exe				
helpctr	5.2.3790.0 (srv03_rtm.030324-2048)	764.00 KB (782,336 bytes)	4/10/2003 10:26 AM	Microsoft Corporation
c:\windows\pchealth\helpctr\binaries\helpctr.exe				
hcappres	5.2.3790.0 (srv03_rtm.030324-2048)	6.50 KB (6,656 bytes)	4/10/2003 10:26 AM	Microsoft Corporation
c:\windows\pchealth\helpctr\binaries\hcappres.dll				
itss	5.2.3790.0 (srv03_rtm.030324-2048)	119.50 KB (122,368 bytes)	3/25/2003 7:00 AM	Microsoft Corporation
c:\windows\system32\itss.dll				
pchshell	5.2.3790.0 (srv03_rtm.030324-2048)	100.50 KB (102,912 bytes)	4/10/2003 10:27 AM	Microsoft Corporation
c:\windows\pchealth\helpctr\binaries\pchshell.dll				
mshtml	6.00.3790.0 (srv03_rtm.030324-2048)	2.78 MB (2,916,352 bytes)	3/25/2003 7:00 AM	Microsoft Corporation
c:\windows\system32\mshtml.dll				
msimtf	5.2.3790.0 (srv03_rtm.030324-2048)	149.00 KB (152,576 bytes)	3/25/2003 7:00 AM	Microsoft Corporation
c:\windows\system32\msimtf.dll				
msctf	5.2.3790.0 (srv03_rtm.030324-2048)	287.00 KB (293,888 bytes)	3/25/2003 7:00 AM	Microsoft Corporation
c:\windows\system32\msctf.dll				
jscrip	5.6.0.8515 436.00 KB (446,464 bytes)		3/25/2003 7:00 AM	Microsoft Corporation
c:\windows\system32\jscrip.dll				
msls31	3.10.349.0 147.00 KB (150,528 bytes)		3/25/2003 7:00 AM	Microsoft Corporation
c:\windows\system32\msls31.dll				
mshtml	6.00.3790.0 (srv03_rtm.030324-2048)	443.50 KB (454,144 bytes)	3/25/2003 7:00 AM	Microsoft Corporation
c:\windows\system32\mshtml.dll				
vbscript	5.6.0.8515 404.00 KB (413,696 bytes)		3/25/2003 7:00 AM	Microsoft Corporation
c:\windows\system32\vbscript.dll				
mfc42	6.05.3014.0	960.00 KB (983,040 bytes)	3/25/2003 7:00 AM	Microsoft Corporation
c:\windows\system32\mfc42.dll				
msinfo	5.2.3790.0 (srv03_rtm.030324-2048)	358.50 KB (367,104 bytes)	4/10/2003 10:27 AM	Microsoft Corporation
c:\windows\pchealth\helpctr\binaries\msinfo.dll				
helpsvc	5.2.3790.0 (srv03_rtm.030324-2048)	720.00 KB (737,280 bytes)	4/10/2003 10:27 AM	Microsoft Corporation
c:\windows\pchealth\helpctr\binaries\helpsvc.exe				

[Services]

Display Name	Name	State	Start Mode	Service Type
Path	Error Control	Start Name	Tag	ID

Alerter	Alerter	Stopped	Disabled	Share Process
c:\windows\system32\svchost.exe -k localservice Normal NT				
AUTHORITY\LocalService 0				
Application Layer Gateway Service		ALG	Stopped	Manual
Own Process	c:\windows\system32\alg.exe	Normal	NT	
AUTHORITY\LocalService 0				
Application Management	AppMgmt	Stopped	Manual	Share
Process	c:\windows\system32\svchost.exe -k netsvcs	Normal		
LocalSystem 0				
Windows Audio	AudioSrv	Stopped	Disabled	Share Process
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0				
Background Intelligent Transfer Service	BITS	Stopped	Manual	
Share Process	c:\windows\system32\svchost.exe -k netsvcs	Normal		
LocalSystem 0				
Computer Browser	Browser	Stopped	Manual	Share Process
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0				
Indexing Service	CiSvc	Stopped	Disabled	Share Process
c:\windows\system32\cisvc.exe Normal LocalSystem 0				
ClipBook	ClipSrv	Stopped	Disabled	Own Process
c:\windows\system32\clipsrv.exe Normal LocalSystem 0				
COM+ System Application	COMSysApp	Stopped	Manual	
Own Process	c:\windows\system32\dlhhost.exe			
/processid:{02d4b3f1-fd88-11d1-960d-00805fc79235} Normal LocalSystem 0				
Cryptographic Services	CryptSvc	Running	Auto	Share
Process c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0				
Distributed File System	Dfs	Stopped	Disabled	Own
Process	c:\windows\system32\dfsrv.exe	Normal	LocalSystem	0
DHCP Client	Dhcp	Running	Auto	Share Process
c:\windows\system32\svchost.exe -k networkservice Normal NT				
AUTHORITY\NetworkService 0				
Logical Disk Manager Administrative Service			dmadmin	Running
Manual	Share Process	c:\windows\system32\dmadmin.exe /com		
Normal LocalSystem 0				
Logical Disk Managerdmsrv	Running	Auto	Share Process	
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0				
DNS Client	Dnscache	Running	Auto	Share Process
c:\windows\system32\svchost.exe -k networkservice Normal NT				
AUTHORITY\NetworkService 0				
Error Reporting Service	ERSvc	Stopped	Manual	Share
Process	c:\windows\system32\svchost.exe -k winerr	Ignore		
LocalSystem 0				
Event Log	Eventlog	Running	Auto	Share Process
c:\windows\system32\services.exe Normal LocalSystem 0				
COM+ Event System	EventSystem	Running	Manual	Share
Process	c:\windows\system32\svchost.exe -k netsvcs	Normal		
LocalSystem 0				
Help and Support	helpsvc	Running	Manual	Share Process
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0				
Human Interface Device Access	HidServ	Stopped	Disabled	Share
Process	c:\windows\system32\svchost.exe -k netsvcs	Normal		
LocalSystem 0				
HTTP SSLHTTPFilter		Stopped	Manual	Share Process
c:\windows\system32\lsass.exe Normal LocalSystem 0				
IMAPI CD-Burning COM Service	ImapiService	Stopped		
Disabled	Own Process	c:\windows\system32\imapi.exe	Normal	
LocalSystem 0				
InterSite Messaging	IsmServ	Stopped	Disabled	Own Process
c:\windows\system32\ismserv.exe Normal LocalSystem 0				
Kerberos Key Distribution Center	kdc	Stopped	Disabled	
Share Process	c:\windows\system32\lsass.exe	Normal		
LocalSystem 0				
Server	lanmanserver	Running	Auto	Share Process
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0				
Workstation	lanmanworkstation	Running	Auto	Share
Process	c:\windows\system32\svchost.exe -k netsvcs	Normal		
LocalSystem 0				

License Logging	LicenseService	Stopped	Disabled	Own	Remote Procedure Call (RPC)	RpcSs	Running	Auto	Share
Process	c:\windows\system32\llsrv.exe	Normal	NT		Process	c:\windows\system32\svchost -k rpcss		Normal	
AUTHORITY\NetworkService	0				LocalSystem	0			
TCP/IP NetBIOS Helper	LmHosts	Running	Auto	Share	Resultant Set of Policy Provider	RSOPProv	Stopped	Manual	Share
Process	c:\windows\system32\svchost.exe -k localservice			Normal	Process	c:\windows\system32\rsopprov.exe		Normal	
NT AUTHORITY\LocalService	0				LocalSystem	0			
Messenger	Messenger	Stopped	Disabled	Share	Special Administration Console Helper	sacsrvr	Stopped	Manual	
Process	c:\windows\system32\svchost.exe -k netsvcs			LocalSystem	Share Process	c:\windows\system32\svchost.exe -k netsvcs		Normal	
NetMeeting	Remote Desktop Sharing	Stopped	Disabled	Own	LocalSystem	0			
Process	c:\windows\system32\mnmsrv.exe			Normal	Security Accounts Manager	SamSs	Running	Auto	Share
LocalSystem	0				Process	c:\windows\system32\lsass.exe		Normal	LocalSystem
Distributed Transaction Coordinator	MSDTC	Running	Auto		Smart Card	SCardSvr	Stopped	Manual	Share
Own Process	c:\windows\system32\msdtc.exe			Normal	Process	c:\windows\system32\scardsvr.exe		Ignore	NT
AUTHORITY\NetworkService	0				AUTHORITY\LocalService	0			
Windows Installer	MSIServer	Stopped	Manual	Share	Task Scheduler	Schedule	Running	Auto	Share
Process	c:\windows\system32\msiexec.exe /v			LocalSystem	Process	c:\windows\system32\svchost.exe -k netsvcs		Normal	LocalSystem
MSSQLSERVER	MSSQLSERVER	Stopped	Manual	Own	Secondary Logon	seclogon	Running	Auto	Share
Process	c:\program files\microsoft\sql\bin\sqlservr.exe			Normal	Process	c:\windows\system32\svchost.exe -k netsvcs		Ignore	LocalSystem
LocalSystem	0				System Event Notification	SENS	Running	Auto	Share
MSSQLServerADHelper	MSSQLServerADHelper			Stopped	Process	c:\windows\system32\svchost.exe -k netsvcs		Normal	
Manual	Own Process				LocalSystem	0			
server\80\tools\bin\sqladhlp.exe		Normal		LocalSystem	Shell Hardware Detection	ShellHWDetection	Running	Auto	
Network DDE	NetDDE	Stopped	Disabled	Share	Share Process	c:\windows\system32\svchost.exe -k netsvcs		Ignore	
Process	c:\windows\system32\netdde.exe			LocalSystem	LocalSystem	0			
Network DDE DSDM	NetDDEdsdm	Stopped	Disabled	Share	Print Spooler	Spooler	Stopped	Disabled	Own
Process	c:\windows\system32\netdde.exe			LocalSystem	Process	c:\windows\system32\spoolsv.exe		Normal	LocalSystem
Net Logon	Netlogon	Stopped	Manual	Share	SQLSERVERAGENT	SQLSERVERAGENT		Stopped	
Process	c:\windows\system32\lsass.exe			LocalSystem	Manual	Own Process			c:\program files\microsoft\sql\bin\sqlagent.exe
Network Connections	Netman	Running	Manual	Share	Normal	LocalSystem		0	
Process	c:\windows\system32\svchost.exe -k netsvcs			LocalSystem	Windows Image Acquisition (WIA)	stisvc	Stopped	Disabled	
Intel NCS NetService	NetSvc	Stopped	Manual	Own	Share Process	c:\windows\system32\svchost.exe -k imgsvc		Normal	
Process	c:\program files\intel\ncs\sync\netsvc.exe			LocalSystem	NT AUTHORITY\LocalService	0			
Network Location Awareness (NLA)	Nla	Running	Manual		Microsoft Software Shadow Copy Provider	swprv	Stopped	Manual	
Share Process	c:\windows\system32\svchost.exe -k netsvcs			Normal	Own Process	c:\windows\system32\svchost.exe -k swprv		Normal	
LocalSystem	0				LocalSystem	0			
File Replication	NtFrs	Stopped	Manual	Own	Performance Logs and Alerts	SysmonLog		Stopped	Manual
Process	c:\windows\system32\ntfrs.exe			LocalSystem	Own Process	c:\windows\system32\smlogsvc.exe		Normal	
NT LM Security Support Provider	NtLmSsp	Stopped	Manual		NT Authority\NetworkService	0			
Share Process	c:\windows\system32\lsass.exe			Normal	Telephony TapiSrv	Stopped	Manual	Share	Process
LocalSystem	0				Process	c:\windows\system32\svchost.exe -k tapisrv		Normal	LocalSystem
Removable Storage	NtmsSvc	Stopped	Manual	Share	Terminal Services	TermService	Running	Manual	Share
Process	c:\windows\system32\svchost.exe -k netsvcs			LocalSystem	Process	c:\windows\system32\svchost.exe -k termsvc		Normal	
Plug and Play	PlugPlay	Running	Auto	Share	LocalSystem	0			
Process	c:\windows\system32\services.exe			LocalSystem	Themes	Themes	Stopped	Disabled	Share
IPSEC Services	PolicyAgent	Running	Auto	Share	Process	c:\windows\system32\svchost.exe -k netsvcs		Normal	LocalSystem
Process	c:\windows\system32\lsass.exe			LocalSystem	0				
Protected Storage	ProtectedStorage	Stopped	Disabled	Share	Telnet	TlntSvr	Stopped	Disabled	Own
Process	c:\windows\system32\lsass.exe			LocalSystem	Process	c:\windows\system32\tlntsvr.exe		Normal	NT AUTHORITY\LocalService
Remote Access Auto Connection Manager	RasAuto	Stopped	Manual		0				
Share Process	c:\windows\system32\svchost.exe -k netsvcs			Normal	Distributed Link Tracking Server	TrkSvr	Stopped	Disabled	Share
LocalSystem	0				Process	c:\windows\system32\svchost.exe -k netsvcs		Normal	
Remote Access Connection Manager	RasMan	Stopped	Manual		LocalSystem	0			
Share Process	c:\windows\system32\svchost.exe -k netsvcs			Normal	Distributed Link Tracking Client	TrkWks	Stopped	Disabled	Share
LocalSystem	0				Process	c:\windows\system32\svchost.exe -k netsvcs		Normal	
Remote Desktop Help Session Manager	RDSessMgr			Stopped	LocalSystem	0			
Manual	Own Process				Terminal Services Session Directory	Tssdis	Stopped	Disabled	
Normal	LocalSystem			0	Own Process	c:\windows\system32\tssdis.exe		Normal	
Routing and Remote Access	RemoteAccess	Stopped	Disabled		LocalSystem	0			
Share Process	c:\windows\system32\svchost.exe -k netsvcs			Normal	Upload Manager	uploadmgr	Stopped	Disabled	Share
LocalSystem	0				Process	c:\windows\system32\svchost.exe -k netsvcs		Normal	LocalSystem
Remote Registry	RemoteRegistry	Running	Auto	Share	Uninterruptible Power Supply	UPS	Stopped	Manual	Own
Process	c:\windows\system32\svchost.exe -k regsvc			Normal	Process	c:\windows\system32\ups.exe		Normal	NT
AUTHORITY\LocalService	0				AUTHORITY\LocalService	0			
Remote Procedure Call (RPC) Locator	RpcLocator			Stopped	Virtual Disk Service	vds	Running	Manual	Own
Manual	Own Process				Process	c:\windows\system32\vds.exe		Normal	LocalSystem
NT AUTHORITY\NetworkService	0				Volume Shadow Copy	VSS	Stopped	Manual	Own
					Process	c:\windows\system32\vssvc.exe		Normal	LocalSystem
					Windows Time	W32Time	Stopped	Manual	Share
					Process	c:\windows\system32\svchost.exe -k netsvcs		Normal	LocalSystem
									0

```

WebClient WebClient Stopped Disabled Share Process
c:\windows\system32\svchost.exe -k localservice Normal NT
AUTHORITY\LocalService 0
WinHTTP Web Proxy Auto-Discovery Service WinHttpAutoProxySvc
Stopped Manual Share Process c:\windows\system32\svchost.exe
-k localservice Normal NT AUTHORITY\LocalService 0
Windows Management Instrumentation winmgmt Running Auto
Share Process c:\windows\system32\svchost.exe -k netsvcs Ignore
LocalSystem 0
Portable Media Serial Number Service WmdmPmSN Stopped
Manual Share Process c:\windows\system32\svchost.exe -k netsvcs
Normal LocalSystem 0
Windows Management Instrumentation Driver Extensions Wmi
Stopped Manual Share Process c:\windows\system32\svchost.exe
-k netsvcs Normal LocalSystem 0
WMI Performance Adapter WmiApSrv Stopped Manual
Own Process c:\windows\system32\wbem\wmiaprv.exe Normal
LocalSystem 0
Automatic Updates wuauerv Stopped Manual Share Process
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0
Wireless Configuration WZCSVC Stopped Manual Share
Process c:\windows\system32\svchost.exe -k netsvcs 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
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\System Tools	All Users:Accessories\System Tools	All Users
Administrative Tools	All Users:Administrative Tools	All Users
Debugging Tools for Windows	All Users:Debugging Tools for Windows	All Users
Intel Network Adapters	All Users:Intel Network Adapters	All Users
Microsoft SQL Server	All Users:Microsoft SQL Server	All Users
Startup	All Users:Startup	All Users
Accessories	NT AUTHORITY\SYSTEM:Accessories	NT AUTHORITY\SYSTEM
Accessories\Accessibility	NT AUTHORITY\SYSTEM:Accessories\Accessibility	NT AUTHORITY\SYSTEM
Accessories\Entertainment	NT AUTHORITY\SYSTEM:Accessories\Entertainment	NT AUTHORITY\SYSTEM
Startup	NT AUTHORITY\SYSTEM:Startup	NT AUTHORITY\SYSTEM
Accessories	IBMSERV4\Administrator:Accessories	IBMSERV4\Administrator
Accessories\Accessibility	IBMSERV4\Administrator:Accessories\Accessibility	IBMSERV4\Administrator
Accessories\Entertainment	IBMSERV4\Administrator:Accessories\Entertainment	IBMSERV4\Administrator
QLogic Corporation	IBMSERV4\Administrator:QLogic Corporation	IBMSERV4\Administrator

```

QLogic Corporation\SANblade Control VIXIBMSERV4\Administrator:QLogic
Corporation\SANblade Control VIX IBMSERV4\Administrator
Startup IBMSERV4\Administrator:Startup
IBMSERV4\Administrator

```

[Startup Programs]

Program	Command	User Name	Location	Startup
desktop	desktop.ini	NT AUTHORITY\SYSTEM		Startup
desktop	desktop.ini	IBMSERV4\Administrator		Startup
desktop	desktop.ini	DEFAULT		Startup
desktop	desktop.ini	All Users	Common	Startup
synctime	synctime.cmd	All Users	Common	Startup
PRONoMgr.exe	c:\program files\intel\ncs\proset\pronomgr.exe	All Users	HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run	
KernelFaultCheck	%systemroot%\system32\dumprep 0 -k	All Users	HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run	

[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
WordPad Document	"%programfiles%\windows nt\accessories\wordpad.exe"
Windows Media Services DRM Storage object	Not Available
Bitmap Image	mspaint.exe

[Windows Error Reporting]

Time	Type	Details
------	------	---------

[Internet Settings]

[Internet Explorer]

[ Following are sub-categories of this main category ]

[Summary]

Item	Value
Version	6.0.3790.0
Build	63790
Application Path	C:\Program Files\Internet Explorer
Language	English (United States)
Active Printer	Not Available
Cipher Strength	128-bit
Content Advisor	Disabled
IEAK Install	No

[File Versions]

File	Version	Size	Date	Path	Company
actxprxy.dll	6.0.3790.0	95 KB	3/25/2003 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
advpack.dll	6.0.3790.0	94 KB	3/25/2003 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
ascctrls.ocx	6.0.3790.0	90 KB	3/25/2003 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
browserc.dll	6.0.3790.0	62 KB	3/25/2003 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
browseui.dll	6.0.3790.0	1,033 KB	3/25/2003 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation

cdfview.dll	6.0.3790.0	144 KB	3/25/2003 8:00:00 AM
C:\WINDOWS\system32	Microsoft Corporation		
comctl32.dll	5.82.3790.0	561 KB	3/25/2003 8:00:00 AM
C:\WINDOWS\system32	Microsoft Corporation		
dxtrans.dll	6.3.3790.0	198 KB	3/25/2003 8:00:00 AM
C:\WINDOWS\system32	Microsoft Corporation		
dxtmsft.dll	6.3.3790.0	344 KB	3/25/2003 8:00:00 AM
C:\WINDOWS\system32	Microsoft Corporation		
iecont.dll	<File Missing>	Not Available	Not Available
Not Available	Not Available		
iecontlc.dll	<File Missing>	Not Available	Not Available
Not Available	Not Available		
iedkcs32.dll	16.0.3790.0	300 KB	3/25/2003 8:00:00 AM
C:\WINDOWS\system32	Microsoft Corporation		
iepeers.dll	6.0.3790.0	230 KB	3/25/2003 8:00:00 AM
C:\WINDOWS\system32	Microsoft Corporation		
iesetup.dll	6.0.3790.0	59 KB	3/25/2003 8:00:00 AM
C:\WINDOWS\system32	Microsoft Corporation		
ieunit.inf	Not Available	20 KB	3/25/2003 8:00:00 AM
C:\WINDOWS\system32	Not Available		
iexplore.exe	6.0.3790.0	90 KB	3/25/2003 8:00:00 AM
C:\Program Files\Internet Explorer	Microsoft Corporation		
imgutil.dll	5.2.3790.0	35 KB	3/25/2003 8:00:00 AM
C:\WINDOWS\system32	Microsoft Corporation		
inetctl.cpl	6.0.3790.0	303 KB	3/25/2003 8:00:00 AM
C:\WINDOWS\system32	Microsoft Corporation		
inetctl.dll	6.0.3790.0	109 KB	3/25/2003 8:00:00 AM
C:\WINDOWS\system32	Microsoft Corporation		
inseng.dll	6.0.3790.0	72 KB	3/25/2003 8:00:00 AM
C:\WINDOWS\system32	Microsoft Corporation		
mlang.dll	6.0.3790.0	570 KB	3/25/2003 8:00:00 AM
C:\WINDOWS\system32	Microsoft Corporation		
msencode.dll	2002.10.4.0	112 KB	3/25/2003 8:00:00 AM
C:\WINDOWS\system32	Not Available		
mshta.exe	6.0.3790.0	26 KB	3/25/2003 8:00:00 AM
C:\WINDOWS\system32	Microsoft Corporation		
mshtml.dll	6.0.3790.0	2,848 KB	3/25/2003 8:00:00 AM
C:\WINDOWS\system32	Microsoft Corporation		
mshtml.tlb	6.0.3790.0	1,319 KB	3/25/2003 8:00:00 AM
C:\WINDOWS\system32	Microsoft Corporation		
mshtml.ed	6.0.3790.0	444 KB	3/25/2003 8:00:00 AM
C:\WINDOWS\system32	Microsoft Corporation		
mshtmlr.dll	6.0.3790.0	55 KB	3/25/2003 8:00:00 AM
C:\WINDOWS\system32	Microsoft Corporation		
msident.dll	6.0.3790.0	47 KB	3/25/2003 8:00:00 AM
C:\WINDOWS\system32	Microsoft Corporation		
msidntld.dll	6.0.3790.0	15 KB	3/25/2003 8:00:00 AM
C:\WINDOWS\system32	Microsoft Corporation		
msieftp.dll	6.0.3790.0	230 KB	3/25/2003 8:00:00 AM
C:\WINDOWS\system32	Microsoft Corporation		
msrating.dll	6.0.3790.0	132 KB	3/25/2003 8:00:00 AM
C:\WINDOWS\system32	Microsoft Corporation		
mstime.dll	6.0.3790.0	491 KB	3/25/2003 8:00:00 AM
C:\WINDOWS\system32	Microsoft Corporation		
occache.dll	6.0.3790.0	89 KB	3/25/2003 8:00:00 AM
C:\WINDOWS\system32	Microsoft Corporation		
proctexe.ocx	6.3.3790.0	78 KB	3/25/2003 8:00:00 AM
C:\WINDOWS\system32	Intel Corporation		
sendmail.dll	6.0.3790.0	52 KB	3/25/2003 8:00:00 AM
C:\WINDOWS\system32	Microsoft Corporation		
shdoclc.dll	6.0.3790.0	589 KB	3/25/2003 8:00:00 AM
C:\WINDOWS\system32	Microsoft Corporation		
shdocvw.dll	6.0.3790.0	1,361 KB	3/25/2003 8:00:00 AM
C:\WINDOWS\system32	Microsoft Corporation		
shfolder.dll	6.0.3790.0	23 KB	3/25/2003 8:00:00 AM
C:\WINDOWS\system32	Microsoft Corporation		
shlwapi.dll	6.0.3790.0	281 KB	3/25/2003 8:00:00 AM
C:\WINDOWS\system32	Microsoft Corporation		

tdc.ocx	1.3.0.3130	58 KB	3/25/2003 8:00:00 AM
C:\WINDOWS\system32	Microsoft Corporation		
url.dll	6.0.3790.0	36 KB	3/25/2003 8:00:00 AM
C:\WINDOWS\system32	Microsoft Corporation		
urlmon.dll	6.0.3790.0	502 KB	3/25/2003 8:00:00 AM
C:\WINDOWS\system32	Microsoft Corporation		
webcheck.dll	6.0.3790.0	262 KB	3/25/2003 8:00:00 AM
C:\WINDOWS\system32	Microsoft Corporation		
wininet.dll	6.0.3790.0	609 KB	3/25/2003 8:00:00 AM
C:\WINDOWS\system32	Microsoft Corporation		

[Connectivity]

Item	Value
Connection Preference	Never dial

LAN Settings

AutoConfigProxy	Not Available
AutoProxyDetectMode	Disabled
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\NetworkService\Local Settings\Temporary Internet Files
Total Disk Space	Not Available
Available Disk Space	Not Available
Maximum Cache Size	Not Available
Available Cache Size	Not Available

[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
My Computer	Custom
Local intranet	Medium-low
Trusted sites	Medium
Internet	High
Restricted sites	High

## Disk Controller Configuration Parameters

### TotalStorage FC2-133 Host Bus Adapter 1

```
// Logical configuration information from Storage Subsystem Flute1log.
// Saved on July 15, 2003
// Firmware package version for Storage Subsystem Flute1log = 05.30.12.00
// NVSRAM package version for Storage Subsystem Flute1log =
NV1742R800NT004
```

```
// Deletes the existing configuration.
show "Deleting the existing configuration.";
set storagesubsystem resetConfiguration = true;
```

```
// Storage Subsystem global logical configuration script commands
show "Setting the Storage Subsystem user label to Flute1log.";
set storagesubsystem userLabel="Flute1log";
```

```
show "Setting the Storage Subsystem media scan rate to disabled.";
set storagesubsystem mediaScanRate=disabled;
```

```
// Remove default logical drive, if exists, script command. NOTE: Default
logical drive name is always = "" (blank).
on error continue;
show "Deleting the default logical drive created during the removal of the
existing configuration.";
delete logicaldrive[""];;
```

```
// Copies the hot spare settings
// NOTE: These statements are wrapped in on-error continue and on-error stop
statements to
// account for minor differences in capacity from the drive of the Storage
Subsystem on which the
// configuration was saved to that of the drives on which the configuration will
be copied.
```

```
show "Creating RAID 1 Logical Drive flute1log on new Array 1.";
// This command creates the Array and the initial Logical Drive on that array.
// NOTE: For Arrays that use all available capacity, the last Logical Drive on
this array is
// created using all remaining capacity by omitting the capacity= logical drive
creation parameter.
create logicaldrive drives[0,1 1,14 1,13 1,12 1,11 1,10 1,9 1,8 1,7 1,6 1,5 1,4
1,3 1,2 1,1 0,14 0,13 0,12 0,11 0,10 0,9 0,8 0,7 0,6 0,5 0,4 0,3 0,2]
RAIDLevel=1 userLabel="flute1log" owner=A segmentSize=64
mapping=default;
show "Setting additional attributes for Logical Drive flute1log.";
// Configuration settings that can not be set during Logical Drive creation.
set logicaldrive["flute1log"] cacheFlushModifier=10;
set logicaldrive["flute1log"] cacheWithoutBatteryEnabled=false;
set logicaldrive["flute1log"] mirrorEnabled=true;
set logicaldrive["flute1log"] readCacheEnabled=false;
set logicaldrive["flute1log"] writeCacheEnabled=true;
set logicaldrive["flute1log"] mediaScanEnabled=false;
set logicaldrive["flute1log"] redundancyCheckEnabled=false;
```

```
set logicaldrive["flute1log"] readAheadMultiplier=1;
set logicaldrive["flute1log"] modificationPriority=high;
show "Setting the Storage Subsystem cache block size to 4.";
set storagesubsystem cacheBlockSize=4;
```

```
show "Setting the Storage Subsystem to begin cache flush at 65% full.";
set storagesubsystem cacheFlushStart=65;
```

```
show "Setting the Storage Subsystem to end cache flush at 15% full.";
set storagesubsystem cacheFlushStop=15;
```

### TotalStorage FC2-133 Host Bus Adapter 2

```
// Logical configuration information from Storage Subsystem Flute_2.
// Saved on July 15, 2003
// Firmware package version for Storage Subsystem Flute_2 = 05.30.09.01
// NVSRAM package version for Storage Subsystem Flute_2 =
NV1742R800NT004
```

```
// Deletes the existing configuration.
show "Deleting the existing configuration.";
set storagesubsystem resetConfiguration = true;
```

```
// Storage Subsystem global logical configuration script commands
show "Setting the Storage Subsystem user label to Flute_2.";
set storagesubsystem userLabel="Flute_2";
```

```
show "Setting the Storage Subsystem media scan rate to disabled.";
set storagesubsystem mediaScanRate=disabled;
```

```
// Remove default logical drive, if exists, script command. NOTE: Default
logical drive name is always = "" (blank).
on error continue;
show "Deleting the default logical drive created during the removal of the
existing configuration.";
delete logicaldrive[""];;
```

```
// Copies the hot spare settings
// NOTE: These statements are wrapped in on-error continue and on-error stop
statements to
// account for minor differences in capacity from the drive of the Storage
Subsystem on which the
// configuration was saved to that of the drives on which the configuration will
be copied.
```

```
show "Creating RAID 0 Logical Drive Flute2_lun1 on new Array 1.";
// This command creates the Array and the initial Logical Drive on that array.
// NOTE: For Arrays that use all available capacity, the last Logical Drive on
this array is
// created using all remaining capacity by omitting the capacity= logical drive
creation parameter.
create logicaldrive drives[20,9 20,8 20,7 20,6 20,5 20,4 20,3 20,2 20,1 21,11
21,10 21,9 21,8 21,7 21,6 21,5 21,4 21,3 21,2 21,1 20,14 20,13 20,12 20,11
20,10 21,14 21,13 21,12] RAIDLevel=0 userLabel="Flute2_lun1" owner=A
segmentSize=64 mapping=default;
show "Setting additional attributes for Logical Drive Flute2_lun1.";
// Configuration settings that can not be set during Logical Drive creation.
set logicaldrive["Flute2_lun1"] cacheFlushModifier=10;
set logicaldrive["Flute2_lun1"] cacheWithoutBatteryEnabled=false;
set logicaldrive["Flute2_lun1"] mirrorEnabled=false;
set logicaldrive["Flute2_lun1"] readCacheEnabled=true;
set logicaldrive["Flute2_lun1"] writeCacheEnabled=false;
```

```

set logicaldrive["Flute2_lun1"] mediaScanEnabled=false;
set logicaldrive["Flute2_lun1"] redundancyCheckEnabled=false;
set logicaldrive["Flute2_lun1"] readAheadMultiplier=1;
set logicaldrive["Flute2_lun1"] modificationPriority=high;

show "Creating RAID 0 Logical Drive Flute2_lun2 on new Array 2.";
// This command creates the Array and the initial Logical Drive on that array.
// NOTE: For Arrays that use all available capacity, the last Logical Drive on
this array is
// created using all remaining capacity by omitting the capacity= logical drive
creation parameter.
create logicaldrive drives[23,7 23,6 23,5 23,4 23,3 23,2 23,1 23,14 23,13 23,12
23,11 23,10 23,9 23,8 22,13 22,12 22,11 22,10 22,9 22,8 22,7 22,6 22,5 22,4
22,3 22,2 22,1 22,14] RAIDLevel=0 userLabel="Flute2_lun2" owner=A
segmentSize=64 mapping=default;
show "Setting additional attributes for Logical Drive Flute2_lun2.";
// Configuration settings that can not be set during Logical Drive creation.
set logicaldrive["Flute2_lun2"] cacheFlushModifier=10;
set logicaldrive["Flute2_lun2"] cacheWithoutBatteryEnabled=false;
set logicaldrive["Flute2_lun2"] mirrorEnabled=false;
set logicaldrive["Flute2_lun2"] readCacheEnabled=true;
set logicaldrive["Flute2_lun2"] writeCacheEnabled=false;
set logicaldrive["Flute2_lun2"] mediaScanEnabled=false;
set logicaldrive["Flute2_lun2"] redundancyCheckEnabled=false;
set logicaldrive["Flute2_lun2"] readAheadMultiplier=1;
set logicaldrive["Flute2_lun2"] modificationPriority=high;

show "Creating RAID 0 Logical Drive Flute2_lun3 on new Array 3.";
// This command creates the Array and the initial Logical Drive on that array.
// NOTE: For Arrays that use all available capacity, the last Logical Drive on
this array is
// created using all remaining capacity by omitting the capacity= logical drive
creation parameter.
create logicaldrive drives[25,14 25,13 25,12 25,11 25,10 25,9 25,8 25,7 25,6
25,5 25,4 25,3 25,2 25,1 24,3 24,2 24,1 24,14 24,13 24,12 24,11 24,10 24,9
24,8 24,7 24,6 24,5 24,4] RAIDLevel=0 userLabel="Flute2_lun3" owner=B
segmentSize=64 mapping=default;
show "Setting additional attributes for Logical Drive Flute2_lun3.";
// Configuration settings that can not be set during Logical Drive creation.
set logicaldrive["Flute2_lun3"] cacheFlushModifier=10;
set logicaldrive["Flute2_lun3"] cacheWithoutBatteryEnabled=false;
set logicaldrive["Flute2_lun3"] mirrorEnabled=false;
set logicaldrive["Flute2_lun3"] readCacheEnabled=true;
set logicaldrive["Flute2_lun3"] writeCacheEnabled=false;
set logicaldrive["Flute2_lun3"] mediaScanEnabled=false;
set logicaldrive["Flute2_lun3"] redundancyCheckEnabled=false;
set logicaldrive["Flute2_lun3"] readAheadMultiplier=1;
set logicaldrive["Flute2_lun3"] modificationPriority=high;

show "Creating RAID 0 Logical Drive Flute2_lun4 on new Array 4.";
// This command creates the Array and the initial Logical Drive on that array.
// NOTE: For Arrays that use all available capacity, the last Logical Drive on
this array is
// created using all remaining capacity by omitting the capacity= logical drive
creation parameter.
create logicaldrive drives[27,1 27,14 27,13 27,12 27,11 27,10 27,9 27,8 27,7
27,6 27,5 27,4 27,3 27,2 26,5 26,4 26,3 26,2 26,1 26,14 26,13 26,12 26,11
26,10 26,9 26,8 26,7 26,6] RAIDLevel=0 userLabel="Flute2_lun4" owner=B
segmentSize=64 mapping=default;
show "Setting additional attributes for Logical Drive Flute2_lun4.";
// Configuration settings that can not be set during Logical Drive creation.
set logicaldrive["Flute2_lun4"] cacheFlushModifier=10;
set logicaldrive["Flute2_lun4"] cacheWithoutBatteryEnabled=false;
set logicaldrive["Flute2_lun4"] mirrorEnabled=false;
set logicaldrive["Flute2_lun4"] readCacheEnabled=true;
set logicaldrive["Flute2_lun4"] writeCacheEnabled=false;
set logicaldrive["Flute2_lun4"] mediaScanEnabled=false;
set logicaldrive["Flute2_lun4"] redundancyCheckEnabled=false;

```

```

set logicaldrive["Flute2_lun4"] readAheadMultiplier=1;
set logicaldrive["Flute2_lun4"] modificationPriority=high;
show "Setting the Storage Subsystem cache block size to 4.";
set storagesubsystem cacheBlockSize=4;

```

```

show "Setting the Storage Subsystem to begin cache flush at 80% full.";
set storagesubsystem cacheFlushStart=80;

```

```

show "Setting the Storage Subsystem to end cache flush at 80% full.";
set storagesubsystem cacheFlushStop=80;

```

## TotalStorage FC2-133 Host Bus Adapter 3

```

// Logical configuration information from Storage Subsystem Flute_3.
// Saved on July 15, 2003
// Firmware package version for Storage Subsystem Flute_3 = 05.30.09.01
// NVSRAM package version for Storage Subsystem Flute_3 =
NV1742R800NT004

```

```

// Deletes the existing configuration.
show "Deleting the existing configuration.";
set storagesubsystem resetConfiguration = true;

```

```

// Storage Subsystem global logical configuration script commands
show "Setting the Storage Subsystem user label to Flute_3.";
set storagesubsystem userLabel="Flute_3";

```

```

show "Setting the Storage Subsystem media scan rate to disabled.";
set storagesubsystem mediaScanRate=disabled;

```

```

// Remove default logical drive, if exists, script command. NOTE: Default
logical drive name is always = "" (blank).
on error continue;
show "Deleting the default logical drive created during the removal of the
existing configuration.";
delete logicaldrive[""];

```

```

// Copies the hot spare settings
// NOTE: These statements are wrapped in on-error continue and on-error stop
statements to
// account for minor differences in capacity from the drive of the Storage
Subsystem on which the
// configuration was saved to that of the drives on which the configuration will
be copied.

```

```

show "Creating RAID 0 Logical Drive Flute3_lun1 on new Array 1.";
// This command creates the Array and the initial Logical Drive on that array.
// NOTE: For Arrays that use all available capacity, the last Logical Drive on
this array is
// created using all remaining capacity by omitting the capacity= logical drive
creation parameter.
create logicaldrive drives[31,14 31,13 31,12 31,11 31,10 31,9 31,8 31,7 31,6
31,5 31,4 31,3 31,2 31,1 30,7 30,6 30,5 30,4 30,3 30,2 30,1 30,14 30,13 30,12
30,11 30,10 30,9 30,8] RAIDLevel=0 userLabel="Flute3_lun1" owner=A
segmentSize=64 mapping=default;
show "Setting additional attributes for Logical Drive Flute3_lun1.";
// Configuration settings that can not be set during Logical Drive creation.
set logicaldrive["Flute3_lun1"] cacheFlushModifier=10;
set logicaldrive["Flute3_lun1"] cacheWithoutBatteryEnabled=false;
set logicaldrive["Flute3_lun1"] mirrorEnabled=false;
set logicaldrive["Flute3_lun1"] readCacheEnabled=true;
set logicaldrive["Flute3_lun1"] writeCacheEnabled=false;

```

```

set logicaldrive["Flute3_lun1"] mediaScanEnabled=false;
set logicaldrive["Flute3_lun1"] redundancyCheckEnabled=false;
set logicaldrive["Flute3_lun1"] readAheadMultiplier=1;
set logicaldrive["Flute3_lun1"] modificationPriority=high;

show "Creating RAID 0 Logical Drive Flute3_lun2 on new Array 2.";
// This command creates the Array and the initial Logical Drive on that array.
// NOTE: For Arrays that use all available capacity, the last Logical Drive on
this array is
// created using all remaining capacity by omitting the capacity= logical drive
creation parameter.
create logicaldrive drives[32,11 32,10 32,9 32,8 32,7 32,6 32,5 32,4 32,3 32,2
32,1 33,13 33,12 33,11 33,10 33,9 33,8 33,7 33,6 33,5 33,4 33,3 33,2 33,1
32,14 32,13 32,12 33,14] RAIDLevel=0 userLabel="Flute3_lun2" owner=A
segmentSize=64 mapping=default;
show "Setting additional attributes for Logical Drive Flute3_lun2.";
// Configuration settings that can not be set during Logical Drive creation.
set logicaldrive["Flute3_lun2"] cacheFlushModifier=10;
set logicaldrive["Flute3_lun2"] cacheWithoutBatteryEnabled=false;
set logicaldrive["Flute3_lun2"] mirrorEnabled=false;
set logicaldrive["Flute3_lun2"] readCacheEnabled=true;
set logicaldrive["Flute3_lun2"] writeCacheEnabled=false;
set logicaldrive["Flute3_lun2"] mediaScanEnabled=false;
set logicaldrive["Flute3_lun2"] redundancyCheckEnabled=false;
set logicaldrive["Flute3_lun2"] readAheadMultiplier=1;
set logicaldrive["Flute3_lun2"] modificationPriority=high;

show "Creating RAID 0 Logical Drive Flute3_lun3 on new Array 3.";
// This command creates the Array and the initial Logical Drive on that array.
// NOTE: For Arrays that use all available capacity, the last Logical Drive on
this array is
// created using all remaining capacity by omitting the capacity= logical drive
creation parameter.
create logicaldrive drives[35,1 35,14 35,13 35,12 35,11 35,10 35,9 35,8 35,7
35,6 35,5 35,4 35,3 35,2 34,5 34,4 34,3 34,2 34,1 34,14 34,13 34,12 34,11
34,10 34,9 34,8 34,7 34,6] RAIDLevel=0 userLabel="Flute3_lun3" owner=B
segmentSize=64 mapping=default;
show "Setting additional attributes for Logical Drive Flute3_lun3.";
// Configuration settings that can not be set during Logical Drive creation.
set logicaldrive["Flute3_lun3"] cacheFlushModifier=10;
set logicaldrive["Flute3_lun3"] cacheWithoutBatteryEnabled=false;
set logicaldrive["Flute3_lun3"] mirrorEnabled=false;
set logicaldrive["Flute3_lun3"] readCacheEnabled=true;
set logicaldrive["Flute3_lun3"] writeCacheEnabled=false;
set logicaldrive["Flute3_lun3"] mediaScanEnabled=false;
set logicaldrive["Flute3_lun3"] redundancyCheckEnabled=false;
set logicaldrive["Flute3_lun3"] readAheadMultiplier=1;
set logicaldrive["Flute3_lun3"] modificationPriority=high;

show "Creating RAID 0 Logical Drive Flute3_lun4 on new Array 4.";
// This command creates the Array and the initial Logical Drive on that array.
// NOTE: For Arrays that use all available capacity, the last Logical Drive on
this array is
// created using all remaining capacity by omitting the capacity= logical drive
creation parameter.
create logicaldrive drives[36,3 36,2 36,1 36,14 36,13 36,12 36,11 36,10 36,9
36,8 36,7 36,6 36,5 36,4 37,9 37,8 37,7 37,6 37,5 37,4 37,3 37,2 37,1 37,14
37,13 37,12 37,11 37,10] RAIDLevel=0 userLabel="Flute3_lun4" owner=B
segmentSize=64 mapping=default;
show "Setting additional attributes for Logical Drive Flute3_lun4.";
// Configuration settings that can not be set during Logical Drive creation.
set logicaldrive["Flute3_lun4"] cacheFlushModifier=10;
set logicaldrive["Flute3_lun4"] cacheWithoutBatteryEnabled=false;
set logicaldrive["Flute3_lun4"] mirrorEnabled=false;
set logicaldrive["Flute3_lun4"] readCacheEnabled=true;
set logicaldrive["Flute3_lun4"] writeCacheEnabled=false;
set logicaldrive["Flute3_lun4"] mediaScanEnabled=false;
set logicaldrive["Flute3_lun4"] redundancyCheckEnabled=false;

```

```

set logicaldrive["Flute3_lun4"] readAheadMultiplier=1;
set logicaldrive["Flute3_lun4"] modificationPriority=high;
show "Setting the Storage Subsystem cache block size to 4.";
set storagesubsystem cacheBlockSize=4;

```

```

show "Setting the Storage Subsystem to begin cache flush at 80% full.";
set storagesubsystem cacheFlushStart=80;

```

```

show "Setting the Storage Subsystem to end cache flush at 80% full.";
set storagesubsystem cacheFlushStop=80;

```

## TotalStorage FC2-133 Host Bus Adapter 4

```

// Logical configuration information from Storage Subsystem Flute_4.
// Saved on July 15, 2003
// Firmware package version for Storage Subsystem Flute_4 = 05.30.09.01
// NVSRAM package version for Storage Subsystem Flute_4 =
NV1742R800NT004

```

```

// Deletes the existing configuration.
show "Deleting the existing configuration.";
set storagesubsystem resetConfiguration = true;

```

```

// Storage Subsystem global logical configuration script commands
show "Setting the Storage Subsystem user label to Flute_4.";
set storagesubsystem userLabel="Flute_4";

```

```

show "Setting the Storage Subsystem media scan rate to disabled.";
set storagesubsystem mediaScanRate=disabled;

```

```

// Remove default logical drive, if exists, script command. NOTE: Default
logical drive name is always = "" (blank).
on error continue;
show "Deleting the default logical drive created during the removal of the
existing configuration.";
delete logicaldrive[""];

```

```

// Copies the hot spare settings
// NOTE: These statements are wrapped in on-error continue and on-error stop
statements to
// account for minor differences in capacity from the drive of the Storage
Subsystem on which the
// configuration was saved to that of the drives on which the configuration will
be copied.

```

```

show "Creating RAID 0 Logical Drive Flute4_lun1 on new Array 1.";
// This command creates the Array and the initial Logical Drive on that array.
// NOTE: For Arrays that use all available capacity, the last Logical Drive on
this array is
// created using all remaining capacity by omitting the capacity= logical drive
creation parameter.
create logicaldrive drives[40,3 40,2 40,1 41,5 41,4 41,3 41,2 41,1 40,14 40,13
40,12 40,11 40,10 40,9 40,8 40,7 40,6 40,5 40,4 41,14 41,13 41,12 41,11 41,10
41,9 41,8 41,7 41,6] RAIDLevel=0 userLabel="Flute4_lun1" owner=A
segmentSize=64 mapping=default;
show "Setting additional attributes for Logical Drive Flute4_lun1.";
// Configuration settings that can not be set during Logical Drive creation.
set logicaldrive["Flute4_lun1"] cacheFlushModifier=10;
set logicaldrive["Flute4_lun1"] cacheWithoutBatteryEnabled=false;
set logicaldrive["Flute4_lun1"] mirrorEnabled=false;
set logicaldrive["Flute4_lun1"] readCacheEnabled=true;
set logicaldrive["Flute4_lun1"] writeCacheEnabled=false;

```

```
set logicaldrive["Flute4_lun1"] mediaScanEnabled=false;
set logicaldrive["Flute4_lun1"] redundancyCheckEnabled=false;
set logicaldrive["Flute4_lun1"] readAheadMultiplier=1;
set logicaldrive["Flute4_lun1"] modificationPriority=high;
```

```
show "Creating RAID 0 Logical Drive Flute4_lun2 on new Array 2.";
// This command creates the Array and the initial Logical Drive on that array.
// NOTE: For Arrays that use all available capacity, the last Logical Drive on
this array is
// created using all remaining capacity by omitting the capacity= logical drive
creation parameter.
```

```
create logicaldrive drives[42,1 43,14 43,13 43,12 43,11 43,10 43,9 43,8 43,7
43,6 43,5 43,4 43,3 43,2 43,1 42,14 42,13 42,12 42,11 42,10 42,9 42,8 42,7
42,6 42,5 42,4 42,3 42,2] RAIDLevel=0 userLabel="Flute4_lun2" owner=A
segmentSize=64 mapping=default;
```

```
show "Setting additional attributes for Logical Drive Flute4_lun2.";
// Configuration settings that can not be set during Logical Drive creation.
set logicaldrive["Flute4_lun2"] cacheFlushModifier=10;
set logicaldrive["Flute4_lun2"] cacheWithoutBatteryEnabled=false;
set logicaldrive["Flute4_lun2"] mirrorEnabled=false;
set logicaldrive["Flute4_lun2"] readCacheEnabled=true;
set logicaldrive["Flute4_lun2"] writeCacheEnabled=false;
set logicaldrive["Flute4_lun2"] mediaScanEnabled=false;
set logicaldrive["Flute4_lun2"] redundancyCheckEnabled=false;
set logicaldrive["Flute4_lun2"] readAheadMultiplier=1;
set logicaldrive["Flute4_lun2"] modificationPriority=high;
```

```
show "Creating RAID 0 Logical Drive Flute4_lun3 on new Array 3.";
// This command creates the Array and the initial Logical Drive on that array.
// NOTE: For Arrays that use all available capacity, the last Logical Drive on
this array is
// created using all remaining capacity by omitting the capacity= logical drive
creation parameter.
```

```
create logicaldrive drives[44,7 44,6 44,5 44,4 44,3 44,2 44,1 44,14 44,13 44,12
44,11 44,10 44,9 44,8 45,11 45,10 45,9 45,8 45,7 45,6 45,5 45,4 45,3 45,2 45,1
45,14 45,13 45,12] RAIDLevel=0 userLabel="Flute4_lun3" owner=B
segmentSize=64 mapping=default;
```

```
show "Setting additional attributes for Logical Drive Flute4_lun3.";
// Configuration settings that can not be set during Logical Drive creation.
set logicaldrive["Flute4_lun3"] cacheFlushModifier=10;
set logicaldrive["Flute4_lun3"] cacheWithoutBatteryEnabled=false;
set logicaldrive["Flute4_lun3"] mirrorEnabled=false;
set logicaldrive["Flute4_lun3"] readCacheEnabled=true;
set logicaldrive["Flute4_lun3"] writeCacheEnabled=false;
set logicaldrive["Flute4_lun3"] mediaScanEnabled=false;
set logicaldrive["Flute4_lun3"] redundancyCheckEnabled=false;
set logicaldrive["Flute4_lun3"] readAheadMultiplier=1;
set logicaldrive["Flute4_lun3"] modificationPriority=high;
```

```
show "Creating RAID 0 Logical Drive Flute4_lun4 on new Array 4.";
// This command creates the Array and the initial Logical Drive on that array.
// NOTE: For Arrays that use all available capacity, the last Logical Drive on
this array is
// created using all remaining capacity by omitting the capacity= logical drive
creation parameter.
```

```
create logicaldrive drives[46,9 46,8 46,7 46,6 46,5 46,4 46,3 46,2 46,1 46,14
46,13 46,12 46,11 46,10 47,13 47,12 47,11 47,10 47,9 47,8 47,7 47,6 47,5 47,4
47,3 47,2 47,1 47,14] RAIDLevel=0 userLabel="Flute4_lun4" owner=B
segmentSize=64 mapping=default;
```

```
show "Setting additional attributes for Logical Drive Flute4_lun4.";
// Configuration settings that can not be set during Logical Drive creation.
set logicaldrive["Flute4_lun4"] cacheFlushModifier=10;
set logicaldrive["Flute4_lun4"] cacheWithoutBatteryEnabled=false;
set logicaldrive["Flute4_lun4"] mirrorEnabled=false;
set logicaldrive["Flute4_lun4"] readCacheEnabled=true;
set logicaldrive["Flute4_lun4"] writeCacheEnabled=false;
set logicaldrive["Flute4_lun4"] mediaScanEnabled=false;
set logicaldrive["Flute4_lun4"] redundancyCheckEnabled=false;
```

```
set logicaldrive["Flute4_lun4"] readAheadMultiplier=1;
set logicaldrive["Flute4_lun4"] modificationPriority=high;
show "Setting the Storage Subsystem cache block size to 4.";
set storagesubsystem cacheBlockSize=4;
```

```
show "Setting the Storage Subsystem to begin cache flush at 80% full.";
set storagesubsystem cacheFlushStart=80;
```

```
show "Setting the Storage Subsystem to end cache flush at 80% full.";
set storagesubsystem cacheFlushStop=80;
```

## TotalStorage FC2-133 Host Bus Adapter 5

```
// Logical configuration information from Storage Subsystem Flute_5.
// Saved on July 15, 2003
// Firmware package version for Storage Subsystem Flute_5 = 05.30.09.01
// NVSRAM package version for Storage Subsystem Flute_5 =
NV1742R800NT004
```

```
// Deletes the existing configuration.
show "Deleting the existing configuration.";
set storagesubsystem resetConfiguration = true;
```

```
// Storage Subsystem global logical configuration script commands
show "Setting the Storage Subsystem user label to Flute_5.";
set storagesubsystem userLabel="Flute_5";
```

```
show "Setting the Storage Subsystem media scan rate to disabled.";
set storagesubsystem mediaScanRate=disabled;
```

```
// Remove default logical drive, if exists, script command. NOTE: Default
logical drive name is always = "" (blank).
on error continue;
show "Deleting the default logical drive created during the removal of the
existing configuration.";
delete logicaldrive[""];
```

```
// Copies the hot spare settings
// NOTE: These statements are wrapped in on-error continue and on-error stop
statements to
// account for minor differences in capacity from the drive of the Storage
Subsystem on which the
// configuration was saved to that of the drives on which the configuration will
be copied.
```

```
show "Creating RAID 0 Logical Drive Flute5_lun1 on new Array 1.";
// This command creates the Array and the initial Logical Drive on that array.
// NOTE: For Arrays that use all available capacity, the last Logical Drive on
this array is
// created using all remaining capacity by omitting the capacity= logical drive
creation parameter.
```

```
create logicaldrive drives[50,14 50,13 50,12 50,11 50,10 50,9 50,8 50,7 50,6
50,5 50,4 50,3 50,2 50,1 51,7 51,6 51,5 51,4 51,3 51,2 51,1 51,14 51,13 51,12
51,11 51,10 51,9 51,8] RAIDLevel=0 userLabel="Flute5_lun1" owner=A
segmentSize=64 mapping=default;
```

```
show "Setting additional attributes for Logical Drive Flute5_lun1.";
// Configuration settings that can not be set during Logical Drive creation.
set logicaldrive["Flute5_lun1"] cacheFlushModifier=10;
set logicaldrive["Flute5_lun1"] cacheWithoutBatteryEnabled=false;
set logicaldrive["Flute5_lun1"] mirrorEnabled=false;
set logicaldrive["Flute5_lun1"] readCacheEnabled=true;
```

```

set logicaldrive["Flute5_lun1"] writeCacheEnabled=false;
set logicaldrive["Flute5_lun1"] mediaScanEnabled=false;
set logicaldrive["Flute5_lun1"] redundancyCheckEnabled=false;
set logicaldrive["Flute5_lun1"] readAheadMultiplier=1;
set logicaldrive["Flute5_lun1"] modificationPriority=high;

show "Creating RAID 0 Logical Drive Flute5_lun2 on new Array 2.";
// This command creates the Array and the initial Logical Drive on that array.
// NOTE: For Arrays that use all available capacity, the last Logical Drive on
this array is
// created using all remaining capacity by omitting the capacity= logical drive
creation parameter.
create logicaldrive drives[52,3 52,2 52,1 53,5 53,4 53,3 53,2 53,1 52,14 52,13
52,12 52,11 52,10 52,9 52,8 52,7 52,6 52,5 52,4 53,14 53,13 53,12 53,11 53,10
53,9 53,8 53,7 53,6] RAIDLevel=0 userLabel="Flute5_lun2" owner=A
segmentSize=64 mapping=default;
show "Setting additional attributes for Logical Drive Flute5_lun2.";
// Configuration settings that can not be set during Logical Drive creation.
set logicaldrive["Flute5_lun2"] cacheFlushModifier=10;
set logicaldrive["Flute5_lun2"] cacheWithoutBatteryEnabled=false;
set logicaldrive["Flute5_lun2"] mirrorEnabled=false;
set logicaldrive["Flute5_lun2"] readCacheEnabled=true;
set logicaldrive["Flute5_lun2"] writeCacheEnabled=false;
set logicaldrive["Flute5_lun2"] mediaScanEnabled=false;
set logicaldrive["Flute5_lun2"] redundancyCheckEnabled=false;
set logicaldrive["Flute5_lun2"] readAheadMultiplier=1;
set logicaldrive["Flute5_lun2"] modificationPriority=high;

```

```

show "Creating RAID 0 Logical Drive Flute5_lun3 on new Array 3.";
// This command creates the Array and the initial Logical Drive on that array.
// NOTE: For Arrays that use all available capacity, the last Logical Drive on
this array is
// created using all remaining capacity by omitting the capacity= logical drive
creation parameter.
create logicaldrive drives[54,1 54,14 54,13 54,12 54,11 54,10 54,9 54,8 54,7
54,6 54,5 54,4 54,3 54,2 55,11 55,10 55,9 55,8 55,7 55,6 55,5 55,4 55,3 55,2
55,1 55,14 55,13 55,12] RAIDLevel=0 userLabel="Flute5_lun3" owner=B
segmentSize=64 mapping=default;
show "Setting additional attributes for Logical Drive Flute5_lun3.";
// Configuration settings that can not be set during Logical Drive creation.
set logicaldrive["Flute5_lun3"] cacheFlushModifier=10;
set logicaldrive["Flute5_lun3"] cacheWithoutBatteryEnabled=false;
set logicaldrive["Flute5_lun3"] mirrorEnabled=false;
set logicaldrive["Flute5_lun3"] readCacheEnabled=true;
set logicaldrive["Flute5_lun3"] writeCacheEnabled=false;
set logicaldrive["Flute5_lun3"] mediaScanEnabled=false;
set logicaldrive["Flute5_lun3"] redundancyCheckEnabled=false;
set logicaldrive["Flute5_lun3"] readAheadMultiplier=1;
set logicaldrive["Flute5_lun3"] modificationPriority=high;

```

```

show "Creating RAID 0 Logical Drive Flute5_lun4 on new Array 4.";
// This command creates the Array and the initial Logical Drive on that array.
// NOTE: For Arrays that use all available capacity, the last Logical Drive on
this array is
// created using all remaining capacity by omitting the capacity= logical drive
creation parameter.
create logicaldrive drives[56,9 56,8 56,7 56,6 56,5 56,4 56,3 56,2 56,1 56,14
56,13 56,12 56,11 56,10 57,13 57,12 57,11 57,10 57,9 57,8 57,7 57,6 57,5 57,4
57,3 57,2 57,1 57,14] RAIDLevel=0 userLabel="Flute5_lun4" owner=B
segmentSize=64 mapping=default;
show "Setting additional attributes for Logical Drive Flute5_lun4.";
// Configuration settings that can not be set during Logical Drive creation.
set logicaldrive["Flute5_lun4"] cacheFlushModifier=10;
set logicaldrive["Flute5_lun4"] cacheWithoutBatteryEnabled=false;
set logicaldrive["Flute5_lun4"] mirrorEnabled=false;
set logicaldrive["Flute5_lun4"] readCacheEnabled=true;
set logicaldrive["Flute5_lun4"] writeCacheEnabled=false;
set logicaldrive["Flute5_lun4"] mediaScanEnabled=false;

```

```

set logicaldrive["Flute5_lun4"] redundancyCheckEnabled=false;
set logicaldrive["Flute5_lun4"] readAheadMultiplier=1;
set logicaldrive["Flute5_lun4"] modificationPriority=high;
show "Setting the Storage Subsystem cache block size to 4.";
set storagesubsystem cacheBlockSize=4;

show "Setting the Storage Subsystem to begin cache flush at 80% full.";
set storagesubsystem cacheFlushStart=80;

show "Setting the Storage Subsystem to end cache flush at 80% full.";
set storagesubsystem cacheFlushStop=80;

```

## TotalStorage FC2-133 Host Bus Adapter 6

```

// Logical configuration information from Storage Subsystem Flute6.
// Saved on July 15, 2003
// Firmware package version for Storage Subsystem Flute6 = 05.30.09.01
// NVSRAM package version for Storage Subsystem Flute6 =
CNV1742R821NT007

```

```

// Deletes the existing configuration.
show "Deleting the existing configuration.";
set storagesubsystem resetConfiguration = true;

```

```

// Storage Subsystem global logical configuration script commands
show "Setting the Storage Subsystem user label to Flute6.";
set storagesubsystem userLabel="Flute6";

```

```

show "Setting the Storage Subsystem media scan rate to disabled.";
set storagesubsystem mediaScanRate=disabled;

```

```

// Remove default logical drive, if exists, script command. NOTE: Default
logical drive name is always = "" (blank).
on error continue;
show "Deleting the default logical drive created during the removal of the
existing configuration.";
delete logicaldrive[""];

```

```

// Copies the hot spare settings
// NOTE: These statements are wrapped in on-error continue and on-error stop
statements to
// account for minor differences in capacity from the drive of the Storage
Subsystem on which the
// configuration was saved to that of the drives on which the configuration will
be copied.

```

```

show "Creating RAID 0 Logical Drive Flute6_lun1 on new Array 1.";
// This command creates the Array and the initial Logical Drive on that array.
// NOTE: For Arrays that use all available capacity, the last Logical Drive on
this array is
// created using all remaining capacity by omitting the capacity= logical drive
creation parameter.
create logicaldrive drives[61,3 61,2 61,1 61,14 61,13 61,12 61,11 61,10 61,9
61,8 61,7 61,6 61,5 61,4 60,13 60,12 60,11 60,10 60,9 60,8 60,7 60,6 60,5 60,4
60,3 60,2 60,1 60,14] RAIDLevel=0 userLabel="Flute6_lun1" owner=A
segmentSize=64 mapping=default;
show "Setting additional attributes for Logical Drive Flute6_lun1.";
// Configuration settings that can not be set during Logical Drive creation.
set logicaldrive["Flute6_lun1"] cacheFlushModifier=10;
set logicaldrive["Flute6_lun1"] cacheWithoutBatteryEnabled=false;
set logicaldrive["Flute6_lun1"] mirrorEnabled=false;
set logicaldrive["Flute6_lun1"] readCacheEnabled=true;

```

```

set logicaldrive["Flute6_lun1"] writeCacheEnabled=false;
set logicaldrive["Flute6_lun1"] mediaScanEnabled=false;
set logicaldrive["Flute6_lun1"] redundancyCheckEnabled=false;
set logicaldrive["Flute6_lun1"] readAheadMultiplier=1;
set logicaldrive["Flute6_lun1"] modificationPriority=high;

```

```

show "Creating RAID 0 Logical Drive Flute6_lun2 on new Array 2.";
// This command creates the Array and the initial Logical Drive on that array.
// NOTE: For Arrays that use all available capacity, the last Logical Drive on
this array is
// created using all remaining capacity by omitting the capacity= logical drive
creation parameter.

```

```

create logicaldrive drives[62,5 62,4 62,3 62,2 62,1 63,7 63,6 63,5 63,4 63,3
63,2 63,1 62,14 62,13 62,12 62,11 62,10 62,9 62,8 62,7 62,6 63,14 63,13 63,12
63,11 63,10 63,9 63,8] RAIDLevel=0 userLabel="Flute6_lun2" owner=A
segmentSize=64 mapping=default;
show "Setting additional attributes for Logical Drive Flute6_lun2.";
// Configuration settings that can not be set during Logical Drive creation.
set logicaldrive["Flute6_lun2"] cacheFlushModifier=10;
set logicaldrive["Flute6_lun2"] cacheWithoutBatteryEnabled=false;
set logicaldrive["Flute6_lun2"] mirrorEnabled=false;
set logicaldrive["Flute6_lun2"] readCacheEnabled=true;
set logicaldrive["Flute6_lun2"] writeCacheEnabled=false;
set logicaldrive["Flute6_lun2"] mediaScanEnabled=false;
set logicaldrive["Flute6_lun2"] redundancyCheckEnabled=false;
set logicaldrive["Flute6_lun2"] readAheadMultiplier=1;
set logicaldrive["Flute6_lun2"] modificationPriority=high;

```

```

show "Creating RAID 0 Logical Drive Flute6_lun3 on new Array 3.";
// This command creates the Array and the initial Logical Drive on that array.
// NOTE: For Arrays that use all available capacity, the last Logical Drive on
this array is
// created using all remaining capacity by omitting the capacity= logical drive
creation parameter.

```

```

create logicaldrive drives[64,1 64,14 64,13 64,12 64,11 64,10 64,9 64,8 64,7
64,6 64,5 64,4 64,3 64,2 65,11 65,10 65,9 65,8 65,7 65,6 65,5 65,4 65,3 65,2
65,1 65,14 65,13 65,12] RAIDLevel=0 userLabel="Flute6_lun3" owner=B
segmentSize=64 mapping=default;
show "Setting additional attributes for Logical Drive Flute6_lun3.";
// Configuration settings that can not be set during Logical Drive creation.
set logicaldrive["Flute6_lun3"] cacheFlushModifier=10;
set logicaldrive["Flute6_lun3"] cacheWithoutBatteryEnabled=false;
set logicaldrive["Flute6_lun3"] mirrorEnabled=false;
set logicaldrive["Flute6_lun3"] readCacheEnabled=true;
set logicaldrive["Flute6_lun3"] writeCacheEnabled=false;
set logicaldrive["Flute6_lun3"] mediaScanEnabled=false;
set logicaldrive["Flute6_lun3"] redundancyCheckEnabled=false;
set logicaldrive["Flute6_lun3"] readAheadMultiplier=1;
set logicaldrive["Flute6_lun3"] modificationPriority=high;

```

```

show "Creating RAID 0 Logical Drive Flute6_lun4 on new Array 4.";
// This command creates the Array and the initial Logical Drive on that array.
// NOTE: For Arrays that use all available capacity, the last Logical Drive on
this array is
// created using all remaining capacity by omitting the capacity= logical drive
creation parameter.

```

```

create logicaldrive drives[66,14 66,13 66,12 66,11 66,10 66,9 66,8 66,7 66,6
66,5 66,4 66,3 66,2 66,1 67,9 67,8 67,7 67,6 67,5 67,4 67,3 67,2 67,1 67,14
67,13 67,12 67,11 67,10] RAIDLevel=0 userLabel="Flute6_lun4" owner=B
segmentSize=64 mapping=default;
show "Setting additional attributes for Logical Drive Flute6_lun4.";
// Configuration settings that can not be set during Logical Drive creation.
set logicaldrive["Flute6_lun4"] cacheFlushModifier=10;
set logicaldrive["Flute6_lun4"] cacheWithoutBatteryEnabled=false;
set logicaldrive["Flute6_lun4"] mirrorEnabled=false;
set logicaldrive["Flute6_lun4"] readCacheEnabled=true;
set logicaldrive["Flute6_lun4"] writeCacheEnabled=false;
set logicaldrive["Flute6_lun4"] mediaScanEnabled=false;

```

```

set logicaldrive["Flute6_lun4"] redundancyCheckEnabled=false;
set logicaldrive["Flute6_lun4"] readAheadMultiplier=1;
set logicaldrive["Flute6_lun4"] modificationPriority=high;
show "Setting the Storage Subsystem cache block size to 4.";
set storagesubsystem cacheBlockSize=4;

```

```

show "Setting the Storage Subsystem to begin cache flush at 80% full.";
set storagesubsystem cacheFlushStart=80;

```

```

show "Setting the Storage Subsystem to end cache flush at 80% full.";
set storagesubsystem cacheFlushStop=80;

```

## Microsoft Windows 2000 Server

### Client System Information Report

The system configuration is identical for all eight clients.

System Information report written at: 07/14/2003 05:33:17 PM  
[System Information]

[ Following are sub-categories of this main category ]

[System Summary]

Item	Value
OS Name	Microsoft Windows 2000 Server
Version	5.0.2195 Service Pack 3 Build 2195
OS Manufacturer	Microsoft Corporation
System Name	FCLIENT10
System Manufacturer	IBM
System Model	-[8647xxx]-
System Type	X86-based PC
Processor	x86 Family 15 Model 2 Stepping 7 GenuineIntel ~2400 Mhz
Processor	x86 Family 15 Model 2 Stepping 7 GenuineIntel ~2400 Mhz
Processor	x86 Family 15 Model 2 Stepping 7 GenuineIntel ~2400 Mhz
Processor	x86 Family 15 Model 2 Stepping 7 GenuineIntel ~2400 Mhz
BIOS Version	)Phoenix - Award WorkstationBIOS v6.00PG
Windows Directory	C:\WINNT
System Directory	C:\WINNT\System32
Boot Device	\Device\Harddisk0\Partition1
Locale	United States
User Name	FCLIENT10\Administrator
Time Zone	Eastern Daylight Time
Total Physical Memory	1,572,316 KB
Available Physical Memory	1,340,852 KB
Total Virtual Memory	4,563,440 KB
Available Virtual Memory	4,248,096 KB
Page File Space	2,991,124 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
4	Direct memory access controller	OK
2	Standard floppy disk 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
0x0D00-0x40B7	PCI bus	OK
0x40B9-0xBFFF	PCI bus	OK
0x7000-0xBFFF - 2553	Intel(R) E7000 Series Hub Interface B PCI-to-PCI Bridge	OK
0x7000-0xBFFF	Intel(R) P64H2 PCI to PCI Bridge - 1460	OK
0x7000-0xBFFF	PCI standard PCI-to-PCI bridge	OK
0x7000-0xBFFF	Intel(R) PRO/100 S Dual Port Server Adapter	OK
0xA000-0xBFFF	Intel(R) P64H2 PCI to PCI Bridge - 1460	OK
0xA000-0xBFFF	DEC 21154 PCI to PCI bridge	OK
0xA000-0xBFFF	Adaptec ANA62044 64-bit 4 port PCI Fast Ethernet Adapter	OK
0xB000-0xB0FF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
0xA400-0xA4FF	Adaptec ANA62044 64-bit 4 port PCI Fast Ethernet Adapter #2	OK
0xA800-0xA8FF	Adaptec ANA62044 64-bit 4 port PCI Fast Ethernet Adapter #3	OK
0xAC00-0xACFF	Adaptec ANA62044 64-bit 4 port PCI Fast Ethernet Adapter #4	OK
0x7400-0x743F	Intel(R) PRO/100 S Dual Port Server Adapter #2	OK
0x8000-0x8FFF	PCI standard PCI-to-PCI bridge	OK
0x8000-0x8FFF	Intel(R) PRO/100 S Dual Port Server Adapter #3	OK
0x8400-0x843F	Intel(R) PRO/100 S Dual Port Server Adapter #4	OK
0x9000-0x90FF	LSI Logic 1020/1030 Ultra320 SCSI Adapter	OK
0x9400-0x94FF	LSI Logic 1020/1030 Ultra320 SCSI Adapter	OK
0xD400-0xD41F - 24C2	Intel(R) 82801DB/DBM USB Universal Host Controller	OK
0xD000-0xD01F - 24C4	Intel(R) 82801DB/DBM USB Universal Host Controller	OK
0xC000-0xC0FF	RAGE XL PCI	OK
0x03B0-0x03BB	RAGE XL PCI	OK
0x03C0-0x03DF	RAGE XL PCI	OK
0x0A79-0x0A79	ISAPNP Read Data Port	OK
0x0279-0x0279	ISAPNP Read Data Port	OK
0x0274-0x0277	ISAPNP Read Data Port	OK
0x0010-0x001F	Motherboard resources	OK
0x0022-0x003F	Motherboard resources	OK
0x0044-0x005F	Motherboard resources	OK
0x0062-0x0063	Motherboard resources	OK
0x0065-0x006F	Motherboard resources	OK
0x0074-0x007F	Motherboard resources	OK
0x0091-0x0093	Motherboard resources	OK
0x00A2-0x00BF	Motherboard resources	OK
0x00E0-0x00EF	Motherboard resources	OK
0x04D0-0x04D1	Motherboard resources	OK
0x0800-0x087F	Motherboard resources	OK
0x0020-0x0021	Programmable interrupt controller	OK
0x00A0-0x00A1	Programmable interrupt controller	OK
0x0080-0x0090	Direct memory access controller	OK
0x0094-0x009F	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
0x03F2-0x03F5	Standard floppy disk controller	OK
0x03F7-0x03F7	Standard floppy disk controller	OK
0x03F8-0x03FF	Communications Port (COM1)	OK
0x02F8-0x02FF	Communications Port (COM2)	OK
0x0378-0x037F	Printer Port (LPT1)	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
0xF000-0xF00F	Intel(R) 82801DB Ultra ATA Storage Controller - 24CB	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
0x5000-0x501F	Intel(R) 82801DB/DBM SMBus Controller - 24C3	OK
0xDC00-0xDCFF	SoundMAX Integrated Digital Audio	OK
0xE000-0xE03F	SoundMAX Integrated Digital Audio	OK
0x4000-0x40BF	Motherboard resources	OK
0x40B8-0x40B8	Not Available	OK

[IRQs]

IRQ Number	Device
9	Microsoft ACPI-Compliant System
48	Broadcom NetXtreme Gigabit Ethernet
52	QLogic QLA23xx PCI Fibre Channel Adapter
56	Adaptec ANA62044 64-bit 4 port PCI Fast Ethernet Adapter
57	Adaptec ANA62044 64-bit 4 port PCI Fast Ethernet Adapter #2
58	Adaptec ANA62044 64-bit 4 port PCI Fast Ethernet Adapter #3
59	Adaptec ANA62044 64-bit 4 port PCI Fast Ethernet Adapter #4
24	Intel(R) PRO/100 S Dual Port Server Adapter
25	Intel(R) PRO/100 S Dual Port Server Adapter #2
28	Intel(R) PRO/100 S Dual Port Server Adapter #3
29	Intel(R) PRO/100 S Dual Port Server Adapter #4
32	LSI Logic 1020/1030 Ultra320 SCSI Adapter
33	LSI Logic 1020/1030 Ultra320 SCSI Adapter
16	Intel(R) 82801DB/DBM USB Universal Host Controller - 24C2
19	Intel(R) 82801DB/DBM USB Universal Host Controller - 24C4
23	Intel PCI to USB Enhanced Host Controller
22	RAGE XL PCI
8	System CMOS/real time clock
13	Numeric data processor
6	Standard floppy disk controller
4	Communications Port (COM1)
3	Communications Port (COM2)
12	PS/2 Compatible Mouse
1	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard
14	Primary IDE Channel
15	Secondary IDE Channel
5	Intel(R) 82801DB/DBM SMBus Controller - 24C3
17	SoundMAX Integrated Digital Audio

[Memory]

Range	Device	Status
0xCA000-0xCDFFF	System board	OK
0xCE200-0xCFFFF	System board	OK
0xF0000-0xFBFFF	System board	OK
0xFC000-0xFFFFF	System board	OK
0x5FFF0000-0x5FFFFFFF	System board	OK
0x0000-0x9FFFF	System board	OK
0x100000-0x5FFFFFFF	System board	OK
0xFEC0000-0xFECFFFFF	System board	OK

0xFEE00000-0xFEEFFFFFFF System board OK  
 0xFFB00000-0xFFB7FFFF System board OK  
 0xFF000000-0xFFFFFFFFFF System board OK  
 0xE0000-0xEFFFFF System board OK  
 0xFFB80000-0xFFBFFFFFFF Intel(r) 82802 Firmware Hub Device OK  
 0xA0000-0xBFFFFF PCI bus OK  
 0xA0000-0xBFFFFF RAGE XL PCI OK  
 0xC0000-0xDFFFFF PCI bus OK  
 0x60000000-0xFEBFFFFFFF PCI bus OK  
 0xEC000000-0xEFFFFFFF Intel(R) E7000 Series Processor to AGP Controller - 2552 OK  
 0xE8000000-0xEBFFFFFFF Intel(R) E7000 Series Processor to AGP Controller - 2552 OK  
 0xF0000000-0xF6FFFFFFF Intel(R) E7000 Series Hub Interface B PCI-to-PCI Bridge - 2553 OK  
 0xF0000000-0xF6FFFFFFF Intel(R) P64H2 PCI to PCI Bridge - 1460 OK  
 0xF6001000-0xF6001FFF Intel(R) P64H2 I/O Advanced Programmable Interrupt Controller - 1461 OK  
 0xF3000000-0xF300FFFF Broadcom NetXtreme Gigabit Ethernet OK  
 0xF3010000-0xF3010FFF QLogic QLA23xx PCI Fibre Channel Adapter OK  
 0xF1000000-0xF2FFFFFFF DEC 21154 PCI to PCI bridge OK  
 0xF2000000-0xF207FFFF Adaptec ANA62044 64-bit 4 port PCI Fast Ethernet Adapter OK  
 0xF2080000-0xF20FFFFFFF Adaptec ANA62044 64-bit 4 port PCI Fast Ethernet Adapter #2 OK  
 0xF2100000-0xF217FFFF Adaptec ANA62044 64-bit 4 port PCI Fast Ethernet Adapter #3 OK  
 0xF2180000-0xF21FFFFFFF Adaptec ANA62044 64-bit 4 port PCI Fast Ethernet Adapter #4 OK  
 0xF6000000-0xF6000FFF Intel(R) P64H2 I/O Advanced Programmable Interrupt Controller - 1461 OK  
 0xF4000000-0xF5FFFFFFF Intel(R) P64H2 PCI to PCI Bridge - 1460 OK  
 0xF5100000-0xF51FFFFFFF PCI standard PCI-to-PCI bridge OK  
 0xF5100000-0xF51FFFFFFF Intel(R) PRO/100 S Dual Port Server Adapter OK  
 0xF5141000-0xF5141FFF Intel(R) PRO/100 S Dual Port Server Adapter OK  
 0xF5140000-0xF5140FFF Intel(R) PRO/100 S Dual Port Server Adapter #2 OK  
 0xF5120000-0xF513FFFF Intel(R) PRO/100 S Dual Port Server Adapter #2 OK  
 0xF5000000-0xF50FFFFFFF PCI standard PCI-to-PCI bridge OK  
 0xF5000000-0xF50FFFFFFF Intel(R) PRO/100 S Dual Port Server Adapter #3 OK  
 0xF5041000-0xF5041FFF Intel(R) PRO/100 S Dual Port Server Adapter #3 OK  
 0xF5040000-0xF5040FFF Intel(R) PRO/100 S Dual Port Server Adapter #4 OK  
 0xF5020000-0xF503FFFF Intel(R) PRO/100 S Dual Port Server Adapter #4 OK  
 0xF5210000-0xF521FFFF LSI Logic 1020/1030 Ultra320 SCSI Adapter OK  
 0xF5200000-0xF520FFFF LSI Logic 1020/1030 Ultra320 SCSI Adapter OK  
 0xF5220000-0xF522FFFF LSI Logic 1020/1030 Ultra320 SCSI Adapter OK  
 0xF5230000-0xF523FFFF LSI Logic 1020/1030 Ultra320 SCSI Adapter OK  
 0xFA000000-0xFA0003FF Intel PCI to USB Enhanced Host Controller OK  
 0xF7000000-0xF7FFFFFFF RAGE XL PCI OK  
 0xF9000000-0xF9000FFF RAGE XL PCI OK

0xFEBFFC00-0xFEBFFFFFFF Intel(R) 82801DB Ultra ATA Storage Controller - 24CB OK  
 0xFA001000-0xFA0011FF SoundMAX Integrated Digital Audio OK  
 0xFA002000-0xFA0020FF SoundMAX Integrated Digital Audio 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
OK	C:\WINNT\System32\MSG723.ACM	Microsoft Corporation	4.4.3385	106.77 KB
OK	C:\WINNT\System32\IAC25_32.AX	Intel Corporation	2.05.53	195.00 KB
OK	C:\WINNT\System32\IAC25_32.AX	Indeo® audio software	12/7/1999 7:00:00 AM	(199,680 bytes)
OK	C:\WINNT\System32\IHACM.ACM	Microsoft Corporation	4.4.3385	33.27 KB
OK	C:\WINNT\System32\MSGSM32.ACM	Microsoft Corporation	5.00.2134.1	22.27 KB (22,800 bytes)
OK	C:\WINNT\System32\TSSOFT32.ACM	DSP GROUP, INC.	1.01	9.27 KB (9,488 bytes)
OK	C:\WINNT\System32\MSG711.ACM	Microsoft Corporation	5.00.2134.1	10.27 KB (10,512 bytes)
OK	C:\WINNT\System32\IMAADP32.ACM	Microsoft Corporation	5.00.2134.1	16.27 KB (16,656 bytes)
OK	C:\WINNT\System32\MSADP32.ACM	Microsoft Corporation	5.00.2134.1	14.77 KB (15,120 bytes)

[Video Codecs]

Codec Version	Manufacturer Size	Description Creation Date	Status	File
OK	C:\WINNT\System32\MSYUV.DLL	Microsoft Corporation	5.00.2134.1	14.77 KB (15,120 bytes)
OK	C:\WINNT\System32\IR50_32.DLL	Intel Corporation	R.5.10.15.2.55	737.50 KB (755,200 bytes)
OK	C:\WINNT\System32\MSH261.DRV	Microsoft Corporation	4.4.3385	167.77 KB (167,696 bytes)
OK	C:\WINNT\System32\MSH263.DRV	Microsoft Corporation	4.4.3385	252.27 KB (258,320 bytes)
OK	C:\WINNT\System32\MSVIDC32.DLL	Microsoft Corporation	5.00.2134.1	27.27 KB (27,920 bytes)
OK	C:\WINNT\System32\IR32_32.DLL	Intel(R) Corporation	Not Available	194.50 KB (199,168 bytes)



c:\winnt\system32\iccvid.dll Radius Inc. OK  
 C:\WINNT\System32\ICCVID.DLL 1.10.0.6 108.00 KB (110,592 bytes) 12/7/1999 7:00:00 AM  
 c:\winnt\system32\msrle32.dll Microsoft Corporation  
 OK C:\WINNT\System32\MSRLE32.DLL 5.00.2134.1  
 10.77 KB (11,024 bytes) 12/7/1999 7:00:00 AM

[CD-ROM]

Item	Value
Drive	Z:
Description	CD-ROM Drive
Media Loaded	False
Media Type	CD-ROM
Name	HL-DT-ST CD-ROM GCR-8480B
Manufacturer	(Standard CD-ROM drives)
Status	OK
Transfer Rate	Not Available
SCSI Target ID	0
PNP Device ID	
IDE\CDROMHL-DT-ST_CD-ROM_GCR-8480B_____1.02____\5&74A80B&0&0.0.0	

[Sound Device]

Item	Value
Name	SoundMAX Integrated Digital Audio
Status	OK
PNP Device ID	
PCI\VEN_8086&DEV_24C5&SUBSYS_027A1014&REV_02\3&13C0B0C5&0&FD	
IRQ Number	17
I/O Port	0xDC00-0xDCFF
I/O Port	0xE000-0xE03F
Memory Range	0xFA001000-0xFA0011FF
Memory Range	0xFA002000-0xFA0020FF
Driver	c:\winnt\system32\drivers\smwdm.sys (552932, PRE-RELEASE)

[Display]

Item	Value
Name	RAGE XL PCI
PNP Device ID	
PCI\VEN_1002&DEV_4752&SUBSYS_02401014&REV_27\4&1A671D0C&0&08F0	
Adapter Type	ATI RAGE XL PCI (B41), ATI Technologies Inc. compatible
Adapter Description	RAGE XL PCI
Adapter RAM	8.00 MB (8,388,608 bytes)
Installed Drivers	ati2drad.dll
Driver Version	5.00.2195.5005
INF File	oem1.inf (ati2mpad section)
Color Planes	1
Color Table Entries	65536
Resolution	1024 x 768 x 85 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&323CF337&0
NumberOfFunctionKeys	12

[Pointing Device]

Item	Value
Hardware Type	PS/2 Compatible Mouse
Number of Buttons	2
Status	OK
PNP Device ID	ACPI\PNP0F13\4&323CF337&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	7/14/2003 11:39:10 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_L2TPMINIPOINT\0000
Last Reset	7/14/2003 11:39:10 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 (52112, 5.00.2195.4052)

Name [00000002] WAN Miniport (PPTP)  
Adapter Type Wide Area Network (WAN)  
Product Name WAN Miniport (PPTP)  
Installed True  
PNP Device ID ROOT\MS\_PPTP\MINIPOINT\0000  
Last Reset 7/14/2003 11:39:10 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 (47888, 5.00.2195.4080)

Name [00000003] Direct Parallel  
Adapter Type Not Available  
Product Name Direct Parallel  
Installed True  
PNP Device ID ROOT\MS\_PT\MINIPOINT\0000  
Last Reset 7/14/2003 11:39:10 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 7/14/2003 11:39:10 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 (93104, 5.00.2195.5241)

Name [00000005] Broadcom NetXtreme Gigabit Ethernet  
Adapter Type Ethernet 802.3  
Product Name Broadcom NetXtreme Gigabit Ethernet  
Installed True

PNP Device ID  
PCI\VEN\_14E4&DEV\_16A7&SUBSYS\_026F1014&REV\_02\5&121CC7C2  
&0&08E810  
Last Reset 7/14/2003 11:39:10 AM  
Index 5  
Service Name b57w2k  
IP Address 192.168.125.10  
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 00:10:DC:76:E2:A7  
Service Name b57w2k  
IRQ Number 48  
Driver c:\winnt\system32\drivers\b57w2k.sys (79336, 2.90.0.0)

Name [00000006] Intel(R) PRO/100 S Dual Port Server Adapter  
Adapter Type Ethernet 802.3  
Product Name Intel(R) PRO/100 S Dual Port Server Adapter  
Installed True  
PNP Device ID  
PCI\VEN\_8086&DEV\_1229&SUBSYS\_10158086&REV\_0D\6&11588895&0  
&2008F810  
Last Reset 7/14/2003 11:39:10 AM  
Index 6  
Service Name E100B  
IP Address 192.168.115.99  
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 00:02:B3:AC:6A:5C  
Service Name E100B  
IRQ Number 24  
I/O Port 0x7000-0xBFFF  
Driver c:\winnt\system32\drivers\e100bnt5.sys (141584, 6.04.14.0000)

Name [00000007] Intel(R) PRO/100 S Dual Port Server Adapter  
Adapter Type Ethernet 802.3  
Product Name Intel(R) PRO/100 S Dual Port Server Adapter  
Installed True  
PNP Device ID  
PCI\VEN\_8086&DEV\_1229&SUBSYS\_10158086&REV\_0D\6&11588895&0  
&2808F810  
Last Reset 7/14/2003 11:39:10 AM  
Index 7  
Service Name E100B  
IP Address 192.168.114.99  
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 00:02:B3:AC:6A:5D  
Service Name E100B  
IRQ Number 25  
I/O Port 0x7400-0x743F  
Driver c:\winnt\system32\drivers\e100bnt5.sys (141584, 6.04.14.0000)

Name [00000008] Intel(R) PRO/100 S Dual Port Server Adapter  
Adapter Type Ethernet 802.3  
Product Name Intel(R) PRO/100 S Dual Port Server Adapter  
Installed True

PNP Device ID  
 PCI\VEN\_8086&DEV\_1229&SUBSYS\_10158086&REV\_0D\6&34E7E7F4&0&2010F810  
 Last Reset 7/14/2003 11:39:10 AM  
 Index 8  
 Service Name E100B  
 IP Address 192.168.117.99  
 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 00:02:B3:AC:67:F2  
 Service Name E100B  
 IRQ Number 28  
 I/O Port 0x8000-0x8FFF  
 Driver c:\winnt\system32\drivers\le100bnt5.sys (141584, 6.04.14.0000)

Name [00000009] Intel(R) PRO/100 S Dual Port Server Adapter  
 Adapter Type Ethernet 802.3  
 Product Name Intel(R) PRO/100 S Dual Port Server Adapter  
 Installed True  
 PNP Device ID  
 PCI\VEN\_8086&DEV\_1229&SUBSYS\_10158086&REV\_0D\6&34E7E7F4&0&2810F810  
 Last Reset 7/14/2003 11:39:10 AM  
 Index 9  
 Service Name E100B  
 IP Address 192.168.116.99  
 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 00:02:B3:AC:67:F3  
 Service Name E100B  
 IRQ Number 29  
 I/O Port 0x8400-0x843F  
 Driver c:\winnt\system32\drivers\le100bnt5.sys (141584, 6.04.14.0000)

Name [00000014] Adaptec ANA62044 64-bit 4 port PCI Fast Ethernet Adapter  
 Adapter Type Ethernet 802.3  
 Product Name Adaptec ANA62044 64-bit 4 port PCI Fast Ethernet Adapter  
 Installed True  
 PNP Device ID  
 PCI\VEN\_9004&DEV\_6915&SUBSYS\_00199004&REV\_03\6&1CD10E71&0&2018E810  
 Last Reset 7/14/2003 11:39:10 AM  
 Index 14  
 Service Name ADPTSF  
 IP Address 192.168.113.99  
 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 00:00:D1:EF:69:61  
 Service Name ADPTSF  
 IRQ Number 56  
 I/O Port 0xA000-0xBFFF  
 Driver c:\winnt\system32\drivers\adptsf50.sys (48384, 5.20.0.24)

Name [00000015] Adaptec ANA62044 64-bit 4 port PCI Fast Ethernet Adapter  
 Adapter Type Ethernet 802.3  
 Product Name Adaptec ANA62044 64-bit 4 port PCI Fast Ethernet Adapter  
 Installed True  
 PNP Device ID  
 PCI\VEN\_9004&DEV\_6915&SUBSYS\_00199004&REV\_03\6&1CD10E71&0&2818E810  
 Last Reset 7/14/2003 11:39:10 AM  
 Index 15  
 Service Name ADPTSF  
 IP Address 192.168.112.99  
 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 00:00:D1:EF:69:62  
 Service Name ADPTSF  
 IRQ Number 57  
 I/O Port 0xA400-0xA4FF  
 Driver c:\winnt\system32\drivers\adptsf50.sys (48384, 5.20.0.24)

Name [00000016] Adaptec ANA62044 64-bit 4 port PCI Fast Ethernet Adapter  
 Adapter Type Ethernet 802.3  
 Product Name Adaptec ANA62044 64-bit 4 port PCI Fast Ethernet Adapter  
 Installed True  
 PNP Device ID  
 PCI\VEN\_9004&DEV\_6915&SUBSYS\_00199004&REV\_03\6&1CD10E71&0&3018E810  
 Last Reset 7/14/2003 11:39:10 AM  
 Index 16  
 Service Name ADPTSF  
 IP Address 192.168.111.99  
 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 00:00:D1:EF:69:63  
 Service Name ADPTSF  
 IRQ Number 58  
 I/O Port 0xA800-0xA8FF  
 Driver c:\winnt\system32\drivers\adptsf50.sys (48384, 5.20.0.24)

Name [00000017] Adaptec ANA62044 64-bit 4 port PCI Fast Ethernet Adapter  
 Adapter Type Ethernet 802.3  
 Product Name Adaptec ANA62044 64-bit 4 port PCI Fast Ethernet Adapter  
 Installed True  
 PNP Device ID  
 PCI\VEN\_9004&DEV\_6915&SUBSYS\_00199004&REV\_03\6&1CD10E71&0&3818E810  
 Last Reset 7/14/2003 11:39:10 AM  
 Index 17  
 Service Name ADPTSF  
 IP Address 192.168.110.99  
 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 00:00:D1:EF:69:64  
 Service Name ADPTSF  
 IRQ Number 59  
 I/O Port 0xAC00-0xACFF  
 Driver c:\winnt\system32\drivers\adptsf50.sys (48384, 5.20.0.24)

[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\_{0BCE0365-C341-4AD3-8978-DD0996C8BA12}]  
 SEQPACKET 14  
 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\_{0BCE0365-C341-4AD3-8978-DD0996C8BA12}]  
 DATAGRAM 14  
 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\_{DC5C9BD6-34CB-4009-B5E2-6141DC7599A4}]  
 SEQPACKET 11  
 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\_{DC5C9BD6-34CB-4009-B5E2-6141DC7599A4}]  
 DATAGRAM 11  
 ConnectionlessService True  
 GuaranteesDelivery False  
 GuaranteesSequencing False  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize20 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\_{4A2A9C9A-DDD8-4FED-BF81-B957B71EB0BE}]  
 SEQPACKET 4  
 ConnectionlessService False  
 GuaranteesDelivery True  
 GuaranteesSequencing True  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize20 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\_{4A2A9C9A-DDD8-4FED-BF81-B957B71EB0BE}]  
 DATAGRAM 4  
 ConnectionlessService True  
 GuaranteesDelivery False  
 GuaranteesSequencing False  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize20 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\_{04626D60-9758-487A-8A9E-33BA956B6C94}]  
 SEQPACKET 3  
 ConnectionlessService False  
 GuaranteesDelivery True  
 GuaranteesSequencing True  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize20 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\_{04626D60-9758-487A-8A9E-33BA956B6C94}]  
 DATAGRAM 3  
 ConnectionlessService True  
 GuaranteesDelivery False  
 GuaranteesSequencing False  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize20 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\_{A330D6A6-3682-45AA-A21E-C98C4527F207}]  
 SEQPACKET 13  
 ConnectionlessService False  
 GuaranteesDelivery True  
 GuaranteesSequencing True  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize20 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\_{A330D6A6-3682-45AA-A21E-C98C4527F207}]  
 DATAGRAM 13  
 ConnectionlessService True  
 GuaranteesDelivery False  
 GuaranteesSequencing False  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes

MessageOriented True  
 MinimumAddressSize20 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\_{F37FFD50-F485-4FB7-A01E-55B193228DED}]  
 SEQPACKET 6  
 ConnectionlessService False  
 GuaranteesDelivery True  
 GuaranteesSequencing True  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize20 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\_{F37FFD50-F485-4FB7-A01E-55B193228DED}]  
 DATAGRAM 6  
 ConnectionlessService True  
 GuaranteesDelivery False  
 GuaranteesSequencing False  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize20 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\_{A3F279E1-7BCC-4CA2-B33C-18E41C7FF498}]  
 SEQPACKET 12  
 ConnectionlessService False  
 GuaranteesDelivery True  
 GuaranteesSequencing True  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize20 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\_{A3F279E1-7BCC-4CA2-B33C-18E41C7FF498}]  
 DATAGRAM 12  
 ConnectionlessService True  
 GuaranteesDelivery False  
 GuaranteesSequencing False  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize20 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\_{70347C08-B8F8-4918-93D1-FFF1CE9DB212}]  
 SEQPACKET 5  
 ConnectionlessService False  
 GuaranteesDelivery True  
 GuaranteesSequencing True  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize20 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\_{70347C08-B8F8-4918-93D1-FFF1CE9DB212}]  
 DATAGRAM 5  
 ConnectionlessService True  
 GuaranteesDelivery False  
 GuaranteesSequencing False  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize20 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\_{4DA8936B-544A-43FE-B2F7-D71CAF394BCD}]  
 SEQPACKET 0  
 ConnectionlessService False

GuaranteesDelivery True  
 GuaranteesSequencing True  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize20 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\_{4DA8936B-544A-43FE-B2F7-D71CAF394BCD}]  
 DATAGRAM 0  
 ConnectionlessService True  
 GuaranteesDelivery False  
 GuaranteesSequencing False  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize20 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\_{E4642D43-676A-4938-B902-1D94A9E14871}]  
 SEQPACKET 1  
 ConnectionlessService False  
 GuaranteesDelivery True  
 GuaranteesSequencing True  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize20 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\_{E4642D43-676A-4938-B902-1D94A9E14871}]  
 DATAGRAM 1  
 ConnectionlessService True  
 GuaranteesDelivery False  
 GuaranteesSequencing False  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize20 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\_{7A1A1C49-720E-4C27-ADCC-2C8703574F7C}]  
 SEQPACKET 2  
 ConnectionlessService False  
 GuaranteesDelivery True  
 GuaranteesSequencing True  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize20 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\_{7A1A1C49-720E-4C27-ADCC-2C8703574F7C}]  
 DATAGRAM 2  
 ConnectionlessService True  
 GuaranteesDelivery False  
 GuaranteesSequencing False  
 MaximumAddressSize 20 bytes  
 MaximumMessageSize 64000 bytes  
 MessageOriented True  
 MinimumAddressSize20 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	5.00.2195.4874
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  
 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 (62512, 5.00.2195.5080)

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 (62512, 5.00.2195.5080)

[Parallel]

Item Value  
 Name LPT1  
 PNP Device ID ACPI\PNP0400\4&323CF337&0

[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 (18,194,284,544 bytes)  
 Free Space 11.83 GB (12,698,902,528 bytes)  
 Volume Name C\_Drive  
 Volume Serial Number 5C29133F  
 Partition Disk #0, Partition #0  
 Partition Size 16.94 GB (18,194,287,104 bytes)  
 Starting Offset 32256 bytes  
 Drive Description Disk drive  
 Drive Manufacturer (Standard disk drives)  
 Drive Model IBM-ESXS ST318432LC FN SCSI Disk Device  
 Drive BytesPerSector 512  
 Drive MediaLoaded True  
 Drive MediaType Fixed hard disk media  
 Drive Partitions 1  
 Drive SCSI Bus 0  
 Drive SCSI LogicalUnit 0  
 Drive SCSI Port 2  
 Drive SCSTargetId 4  
 Drive SectorsPerTrack 63  
 Drive Size 18194319360 bytes  
 Drive TotalCylinders 2212  
 Drive TotalSectors 35535780  
 Drive TotalTracks 564060  
 Drive TracksPerCylinder 255



[SCSI]

Item Value  
 Name QLogic QLA23xx PCI Fibre Channel Adapter  
 Caption QLogic QLA23xx PCI Fibre Channel Adapter  
 Driver ql2300  
 Status OK  
 PNP Device ID  
 PCI\VEN\_1077&DEV\_2312&SUBSYS\_010C1077&REV\_02\5&121CC7C2&0&10E810  
 Device ID  
 PCI\VEN\_1077&DEV\_2312&SUBSYS\_010C1077&REV\_02\5&121CC7C2&0&10E810  
 Device Map Not Available  
 Index Not Available  
 Max Number Controlled Not Available  
 IRQ Number 52  
 I/O Port 0xB000-0xB0FF  
 Driver c:\winnt\system32\drivers\ql2300.sys (441733, 8.2.0 Beta 10 (W2K VI))

Name LSI Logic 1020/1030 Ultra320 SCSI Adapter  
 Caption LSI Logic 1020/1030 Ultra320 SCSI Adapter  
 Driver SYMMPI  
 Status OK  
 PNP Device ID  
 PCI\VEN\_1000&DEV\_0030&SUBSYS\_10001014&REV\_07\5&21593F33&0&18F810  
 Device ID  
 PCI\VEN\_1000&DEV\_0030&SUBSYS\_10001014&REV\_07\5&21593F33&0&18F810  
 Device Map Not Available  
 Index Not Available  
 Max Number Controlled Not Available  
 IRQ Number 32  
 I/O Port 0x9000-0x90FF  
 Driver c:\winnt\system32\drivers\symmpi.sys (38512, 1.08.22.00)

Name LSI Logic 1020/1030 Ultra320 SCSI Adapter  
 Caption LSI Logic 1020/1030 Ultra320 SCSI Adapter  
 Driver SYMMPI  
 Status OK  
 PNP Device ID  
 PCI\VEN\_1000&DEV\_0030&SUBSYS\_10001014&REV\_07\5&21593F33&0&19F810  
 Device ID  
 PCI\VEN\_1000&DEV\_0030&SUBSYS\_10001014&REV\_07\5&21593F33&0&19F810  
 Device Map Not Available  
 Index Not Available  
 Max Number Controlled Not Available  
 IRQ Number 33  
 I/O Port 0x9400-0x94FF  
 Driver c:\winnt\system32\drivers\symmpi.sys (38512, 1.08.22.00)

[Printing]

Name Port Name Server Name  
 No printing information

[Problem Devices]

Device	PNP Device ID	Error Code
Not Available	ACPI\IBM37D6\2&DABA3FF&0	28
Not Available	ACPI\ASF0001\2&DABA3FF&0	28

[USB]

Device	PNP Device ID
Intel(R) 82801DB/DBM USB Universal Host Controller - 24C2	PCI\VEN_8086&DEV_24C2&SUBSYS_027A1014&REV_02\3&13C0B0C5&0&E8
USB Root Hub	USB\ROOT_HUB\4&BF47197&0
Intel(R) 82801DB/DBM USB Universal Host Controller - 24C4	PCI\VEN_8086&DEV_24C4&SUBSYS_027A1014&REV_02\3&13C0B0C5&0&E9
USB Root Hub	USB\ROOT_HUB\4&31A3CFF4&0
Intel PCI to USB Enhanced Host Controller	PCI\VEN_8086&DEV_24CD&SUBSYS_027A1014&REV_02\3&13C0B0C5&0&EF
USB 2.0 Root Hub	USB\ROOT_HUB20\4&E58626F&0

[Software Environment]

[ Following are sub-categories of this main category ]

[Drivers]

Name	Description	File	Type	Started	Start Mode
abiosdsk	Abiosdsk	Not Available	Kernel Driver	False	False
abp480n5	abp480n5	Not Available	Kernel Driver	False	False
acpi	Microsoft ACPI Driver	c:\winnt\system32\drivers\acpi.sys	Kernel Driver	True	Normal
acpiec	ACPIEC	c:\winnt\system32\drivers\acpiec.sys	Kernel Driver	False	False
adptsf	Adaptec DuraLAN PCI Ethernet/Fast Ethernet driver for Windows 2000	c:\winnt\system32\drivers\adptsf50.sys	Kernel Driver	True	Normal
adpu160m	adpu160m	Not Available	Kernel Driver	False	False
aeaudio	aeaudio	c:\winnt\system32\drivers\aeaudio.sys	Kernel Driver	True	Normal
afd	AFD Networking Support Environment	c:\winnt\system32\drivers\afd.sys	Kernel Driver	True	Auto
agp440	Intel AGP Bus Filter	c:\winnt\system32\drivers\agp440.sys	Kernel Driver	True	Normal
aha154x	Aha154x	Not Available	Kernel Driver	False	False
aic116x	aic116x	Not Available	Kernel Driver	False	False
aic78u2	aic78u2	Not Available	Kernel Driver	False	False
aic78xx	aic78xx	Not Available	Kernel Driver	False	False
ami0nt	ami0nt	Not Available	Kernel Driver	False	False
amsint	amsint	Not Available	Kernel Driver	False	False
asc	asc	Not Available	Kernel Driver	False	False
asc3350p	asc3350p	Not Available	Kernel Driver	False	False
asc3550	asc3550	Not Available	Kernel Driver	False	False
asynmac	RAS Asynchronous Media Driver	c:\winnt\system32\drivers\asynmac.sys	Kernel Driver	False	False

atapi	Standard IDE/ESDI Hard Disk Controller					
c:\winnt\system32\drivers\atapi.sys	Kernel Driver	True				
Boot	Running	OK	Normal	False	True	
atdisk	Atdisk	Not Available	Kernel Driver	False	False	
Disabled	Stopped	OK	Ignore	False	False	
ati2mpad	ati2mpad	c:\winnt\system32\drivers\ati2mpad.sys	Kernel Driver	False	False	
Driver	True	Manual	Running	OK	Ignore	
True						
atirage3	atirage3	c:\winnt\system32\drivers\atimpab.sys	Kernel Driver	False	False	
Driver	False	Manual	Stopped	OK	Ignore	
False						
atmarpc	ATM ARP Client Protocol					
c:\winnt\system32\drivers\atmarpc.sys	Kernel Driver	False				
Manual	Stopped	OK	Normal	False	False	
audstub	Audio Stub Driver	c:\winnt\system32\drivers\audstub.sys	Kernel Driver	True	Manual	
Kernel Driver	True	Manual	Running	OK	Normal	
False	True					
b57w2k	Broadcom NetXtreme Gigabit Ethernet					
c:\winnt\system32\drivers\b57w2k.sys	Kernel Driver	True				
Manual	Running	OK	Normal	False	True	
beep	Beep	c:\winnt\system32\drivers\beep.sys	Kernel Driver	True	System	
Driver	True	System	Running	OK	Normal	
True						
buslogic	BusLogic	Not Available	Kernel Driver	False	False	
Disabled	Stopped	OK	Normal	False	False	
cd20xrnt	cd20xrnt	Not Available	Kernel Driver	False	False	
Disabled	Stopped	OK	Normal	False	False	
cdaudio	Cdaudio	c:\winnt\system32\drivers\cdaudio.sys	Kernel Driver	False	System	
Driver	False	System	Stopped	OK	Ignore	
False						
cdfs	Cdfs	c:\winnt\system32\drivers\cdfs.sys	File System Driver	True	Disabled	
Driver	True	Disabled	Running	OK	Normal	
True						
cdrom	CD-ROM Driver	c:\winnt\system32\drivers\cdrom.sys	Kernel Driver	True	System	
Kernel Driver	True	System	Running	OK	Normal	
False	True					
changer	Changer	Not Available	Kernel Driver	False	False	
System	Stopped	OK	Ignore	False	False	
cpqarray	Cpqarray	Not Available	Kernel Driver	False	False	
Disabled	Stopped	OK	Normal	False	False	
cpqarry2	cpqarry2	Not Available	Kernel Driver	False	False	
Disabled	Stopped	OK	Normal	False	False	
cpqcalm	cpqcalm	Not Available	Kernel Driver	False	False	
Disabled	Stopped	OK	Normal	False	False	
cpqfws2e	cpqfws2e	Not Available	Kernel Driver	False	False	
Disabled	Stopped	OK	Normal	False	False	
dac960nt	dac960nt	Not Available	Kernel Driver	False	False	
Disabled	Stopped	OK	Normal	False	False	
deckzpsx	deckzpsx	Not Available	Kernel Driver	False	False	
Disabled	Stopped	OK	Normal	False	False	
dfsdriver	DfsDriver	c:\winnt\system32\drivers\dfs.sys	File System Driver	True	Boot	
True	Boot	Running	OK	Normal	False	
True						
disk	Disk Driver	c:\winnt\system32\drivers\disk.sys	Kernel Driver	True	Boot	
Kernel Driver	True	Boot	Running	OK	Normal	
False	True					
diskperf	Diskperf	c:\winnt\system32\drivers\diskperf.sys	Kernel Driver	True	Boot	
Driver	True	Boot	Running	OK	Normal	
True						
dmboot	dmboot	c:\winnt\system32\drivers\dmboot.sys	Kernel Driver	False	Disabled	
Driver	False	Disabled	Stopped	OK	Normal	
False						
dmio	Logical Disk Manager Driver					
c:\winnt\system32\drivers\dmio.sys	Kernel Driver	True				
Boot	Running	OK	Normal	False	True	
dmload	dmload	c:\winnt\system32\drivers\dmload.sys	Kernel Driver	True	Boot	
Driver	True	Boot	Running	OK	Normal	
True						
dmusic	Microsoft DirectMusic SW Synth (WDM)					
c:\winnt\system32\drivers\dmusic.sys	Kernel Driver	False				
Manual	Stopped	OK	Normal	False	False	
e100b	Intel(R) PRO Adapter Driver					
c:\winnt\system32\drivers\e100bnt5.sys	Kernel Driver	True				
Manual	Running	OK	Normal	False	True	
efs	EFS	c:\winnt\system32\drivers\efs.sys	File System Driver	True	Disabled	
True	Disabled	Running	OK	Normal	False	
fastfat	Fastfat	c:\winnt\system32\drivers\fastfat.sys	File System Driver	True	Disabled	
Driver	True	Disabled	Running	OK	Normal	
True						
fd16_700	Fd16_700	Not Available	Kernel Driver	False	False	
Disabled	Stopped	OK	Normal	False	False	
fdc	Floppy Disk Controller Driver					
c:\winnt\system32\drivers\fdc.sys	Kernel Driver	True	Manual	Running	OK	
Kernel Driver	True	Manual	Running	OK	Normal	
False	True					
fips	Fips	c:\winnt\system32\drivers\fips.sys	Kernel Driver	Driver	True	
Driver	True	Auto	Running	OK	Normal	
True						
fireport	fireport	Not Available	Kernel Driver	False	False	
Disabled	Stopped	OK	Normal	False	False	
flashpnt	flashpnt	Not Available	Kernel Driver	False	False	
Disabled	Stopped	OK	Normal	False	False	
flpydisk	Floppy Disk Driver	c:\winnt\system32\drivers\flpydisk.sys	Kernel Driver	True	Manual	
Kernel Driver	True	Manual	Running	OK	Normal	
False	True					
ftdisk	Volume Manager Driver					
c:\winnt\system32\drivers\ftdisk.sys	Kernel Driver	True				
Boot	Running	OK	Normal	False	True	
gpc	Generic Packet Classifier					
c:\winnt\system32\drivers\msgpc.sys	Kernel Driver	True				
Manual	Running	OK	Normal	False	True	
i8042prt	i8042 Keyboard and PS/2 Mouse Port Driver					
c:\winnt\system32\drivers\i8042prt.sys	Kernel Driver	True				
System	Running	OK	Normal	False	True	
ini910u	ini910u	Not Available	Kernel Driver	False	False	
Disabled	Stopped	OK	Normal	False	False	
intelide	IntelIde	Not Available	Kernel Driver	False	False	
Disabled	Stopped	OK	Normal	False	False	
ipfilterdriver	IP Traffic Filter Driver					
c:\winnt\system32\drivers\ipfltdrv.sys	Kernel Driver	False				
Manual	Stopped	OK	Normal	False	False	
ipinip	IP in IP Tunnel Driver					
c:\winnt\system32\drivers\ipinip.sys	Kernel Driver	False	Manual	Stopped	OK	
Kernel Driver	False	Manual	Stopped	OK	Normal	
False	False					
ipnat	IP Network Address Translator					
c:\winnt\system32\drivers\ipnat.sys	Kernel Driver	False	Manual	Stopped	OK	
Kernel Driver	False	Manual	Stopped	OK	Normal	
False	False					
ipsec	IPSEC driver					
c:\winnt\system32\drivers\ipsec.sys	Kernel Driver	True	Manual	Running	OK	
Kernel Driver	True	Manual	Running	OK	Normal	
False	True					
ipsraidn	ipsraidn	Not Available	Kernel Driver	False	False	
Disabled	Stopped	OK	Normal	False	False	
irenum	IR Enumerator Service					
c:\winnt\system32\drivers\irenum.sys	Kernel Driver	False				
Manual	Stopped	OK	Normal	False	False	
isapnp	PnP ISA/EISA Bus Driver					
c:\winnt\system32\drivers\isapnp.sys	Kernel Driver	True				
Boot	Running	OK	Critical	False	True	
kbdclass	Keyboard Class Driver					
c:\winnt\system32\drivers\kbdclass.sys	Kernel Driver	True				
System	Running	OK	Normal	False	True	
kmixer	Microsoft Kernel Wave Audio Mixer					
c:\winnt\system32\drivers\kmixer.sys	Kernel Driver	False				
Manual	Stopped	OK	Normal	False	False	
ksecdd	KSecDD	c:\winnt\system32\drivers\ksecdd.sys	Kernel Driver	True	Boot	
Driver	True	Boot	Running	OK	Normal	
True						

lbrtfdc	lbrtfdc	Not Available	Kernel Driver	False	nwlnkflt	IPX Traffic Filter Driver							
System	Stopped	OK	Ignore	False	False	c:\winnt\system32\drivers\nwlnkflt.sys	Kernel Driver	False	False				
lp6nds35	lp6nds35	Not Available	Kernel Driver	False	Manual	Stopped	OK	Normal	False	False			
Disabled	Stopped	OK	Normal	False	nwlnkfw	IPX Traffic Forwarder Driver							
mnmdd	mnmdd	c:\winnt\system32\drivers\mnmdd.sys	Kernel Driver	False	c:\winnt\system32\drivers\nwlnkfw.sys	Kernel Driver	False	False					
Driver	True	System	Running	OK	Ignore	False	False						
True					parallel	Parallel class driver	c:\winnt\system32\drivers\parallel.sys	Kernel Driver	True	Manual	Running	OK	Normal
modem	Modem	c:\winnt\system32\drivers\modem.sys	Kernel Driver	False	False	True							
Driver	False	Manual	Stopped	OK	Ignore	False							
False					parport	Parallel port driver	c:\winnt\system32\drivers\parport.sys	Kernel Driver	True	System	Running	OK	Ignore
mouclass	Mouse Class Driver	c:\winnt\system32\drivers\mouclass.sys	Kernel Driver	False	True								
Kernel Driver	True	System	Running	OK	Normal	False							
False	True				partmgr	PartMgr	c:\winnt\system32\drivers\partmgr.sys	Kernel Driver	True	Boot	Running	OK	Normal
mountmgr	MountMgr	c:\winnt\system32\drivers\mountmgr.sys	Kernel Driver	False	True								
Driver	True	Boot	Running	OK	Normal	False							
True					parvdm	ParVdm	c:\winnt\system32\drivers\parvdm.sys	Kernel Driver	True	Auto	Running	OK	Ignore
mraid35x	mraid35x	Not Available	Kernel Driver	False	True								
Disabled	Stopped	OK	Normal	False	False								
mrxsmb	MRXSMB	c:\winnt\system32\drivers\mrxsmb.sys	File System Driver	True	System	Running	OK	Normal	False				
Driver	True	System	Running	OK	Normal	False							
True					pci	PCI Bus Driver	c:\winnt\system32\drivers\pci.sys	Kernel Driver	True	Boot	Running	OK	Critical
msfs	Msfs	c:\winnt\system32\drivers\msfs.sys	File System Driver	True	System	Running	OK	Normal	False				
Driver	True	System	Running	OK	Normal	False							
True					pcidump	PCIDump	Not Available	Kernel Driver	False				
mskssrv	Microsoft Streaming Service Proxy	c:\winnt\system32\drivers\mskssrv.sys	Kernel Driver	Manual	Stopped	OK	Ignore	False	False				
Kernel Driver	True	System	Running	OK	Normal	False							
Manual	Stopped	OK	Normal	False	False								
msplock	Microsoft Streaming Clock Proxy	c:\winnt\system32\drivers\msplock.sys	Kernel Driver	Manual	Stopped	OK	Normal	False	False				
Kernel Driver	True	System	Running	OK	Normal	False							
Manual	Stopped	OK	Normal	False	False								
mspqm	Microsoft Streaming Quality Manager Proxy	c:\winnt\system32\drivers\mspqm.sys	Kernel Driver	Manual	Stopped	OK	Normal	False	False				
Kernel Driver	True	System	Running	OK	Normal	False							
Manual	Stopped	OK	Normal	False	False								
mup	Mup	c:\winnt\system32\drivers\mup.sys	File System Driver	True	Boot	Running	OK	Normal	False				
Driver	True	Boot	Running	OK	Normal	False							
True					ncrc710	Nrc710	Not Available	Kernel Driver	Disabled	Stopped	OK	Normal	False
ncrc710	Nrc710	Not Available	Kernel Driver	Disabled	Stopped	OK	Normal	False	False				
Disabled	Stopped	OK	Normal	False	False								
ndis	NDIS System Driver	c:\winnt\system32\drivers\ndis.sys	Kernel Driver	False	True								
Kernel Driver	True	Boot	Running	OK	Normal	False							
False	True				ndistapi	Remote Access NDIS TAPI Driver	c:\winnt\system32\drivers\ndistapi.sys	Kernel Driver	Manual	Running	OK	Normal	False
Kernel Driver	True	System	Running	OK	Normal	False							
Manual	Running	OK	Normal	False	True								
ndiswan	Remote Access NDIS WAN Driver	c:\winnt\system32\drivers\ndiswan.sys	Kernel Driver	Manual	Running	OK	Normal	False	True				
Kernel Driver	True	System	Running	OK	Normal	False							
Manual	Running	OK	Normal	False	True								
ndproxy	NDIS Proxy	c:\winnt\system32\drivers\ndproxy.sys	Kernel Driver	False	True								
Kernel Driver	True	Manual	Running	OK	Normal	False							
False	True				netbios	NetBIOS Interface	c:\winnt\system32\drivers\netbios.sys	File System Driver	True	System	Running	OK	Normal
File System Driver	True	System	Running	OK	Normal	False							
False	True				netbt	NetBios over Tcpip	c:\winnt\system32\drivers\netbt.sys	Kernel Driver	False	True			
Kernel Driver	True	System	Running	OK	Normal	False							
False	True				netdetect	NetDetect	c:\winnt\system32\drivers\netdetect.sys	Kernel Driver	False	Manual	Stopped	OK	Normal
Kernel Driver	False	Manual	Stopped	OK	Normal	False							
False					npfs	Npfs	c:\winnt\system32\drivers\npfs.sys	File System Driver	True	System	Running	OK	Normal
File System Driver	True	System	Running	OK	Normal	False							
True					ntfs	Ntfs	c:\winnt\system32\drivers\ntfs.sys	File System Driver	True	Disabled	Running	OK	Normal
File System Driver	True	Disabled	Running	OK	Normal	False							
True					null	Null	c:\winnt\system32\drivers\null.sys	Kernel Driver	True	System	Running	OK	Normal
Kernel Driver	True	System	Running	OK	Normal	False							
Driver	True	System	Running	OK	Normal	False							
True													



TEMP %USERPROFILE%\Local Settings\Temp  
 FCLIENT10\Administrator  
 TMP %USERPROFILE%\Local Settings\Temp  
 FCLIENT10\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
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
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	192	11	204800	
1413120	7/14/2003 3:39:44 PM	5.00.2195.5382		44.77 KB	
(45,840 bytes)		12/7/1999 7:00:00 AM			
csrss.exe	Not Available	216	13	Not Available	
Not Available		7/14/2003 3:39:46 PM		Unknown	Unknown
Unknown					
winlogon.exe	c:\winnt\system32\winlogon.exe	212	13		
204800	1413120	7/14/2003 3:39:47 PM		5.00.2195.5386	
174.77 KB (178,960 bytes)		2/28/2003 2:03:49 PM			
services.exe	c:\winnt\system32\services.exe	268	9		
204800	1413120	7/14/2003 3:39:48 PM		5.00.2195.3940	
86.77 KB (88,848 bytes)		12/7/1999 7:00:00 AM			
lsass.exe	c:\winnt\system32\lsass.exe	280	9	204800	
1413120	7/14/2003 3:39:48 PM	5.00.2195.5430		32.77 KB	
(33,552 bytes)		12/7/1999 7:00:00 AM			
termsrv.exe	c:\winnt\system32\termsrv.exe	392	10		
204800	1413120	7/14/2003 3:39:48 PM		5.00.2195.5276	
138.77 KB (142,096 bytes)		2/28/2003 2:03:47 PM			
svchost.exe	c:\winnt\system32\svchost.exe	496	8		
204800	1413120	7/14/2003 3:39:50 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	524	8		
204800	1413120	7/14/2003 3:39:50 PM		5.00.2195.4299	
44.27 KB (45,328 bytes)		2/12/2003 7:39:06 AM			
msdtc.exe	c:\winnt\system32\msdtc.exe	556	8	204800	
1413120	7/14/2003 3:39:50 PM	1999.9.3421.3		6.77 KB	
(6,928 bytes)		2/12/2003 7:52:40 AM			
svchost.exe	c:\winnt\system32\svchost.exe	720	8		
204800	1413120	7/14/2003 3:39:51 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	744	9	204800	
1413120	7/14/2003 3:39:51 PM	5.00.2195.4907		81.27 KB	
(83,216 bytes)		7/22/2002 1:05:04 PM			

regsvc.exe	c:\winnt\system32\regsvc.exe	828	8	204800	
1413120	7/14/2003 3:39:51 PM	5.00.2195.3649		65.27 KB	
(66,832 bytes)		2/28/2003 2:03:45 PM			
rsys.exe	Not Available	928	8	Not Available	
Not Available		7/14/2003 3:39:52 PM		Unknown	Unknown
Unknown					
mstask.exe	c:\winnt\system32\mstask.exe	800	8	204800	
1413120	7/14/2003 3:40:16 PM	4.71.2195.1		115.77 KB	
(118,544 bytes)		2/28/2003 2:03:40 PM			
tcpsvcs.exe	c:\winnt\system32\tcpsvcs.exe	1052	8		
204800	1413120	7/14/2003 3:40:16 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	1108	8		
204800	1413120	7/14/2003 3:40:16 PM		1.50.1085.0070	
192.08 KB (196,685 bytes)		2/28/2003 2:03:52 PM			
svchost.exe	c:\winnt\system32\svchost.exe	1124	8		
204800	1413120	7/14/2003 3:40:16 PM		5.00.2134.1	
7.77 KB (7,952 bytes)		12/7/1999 7:00:00 AM			
dfssvc.exe	c:\winnt\system32\dfssvc.exe	1112	8	204800	
1413120	7/14/2003 3:40:21 PM	5.00.2195.3649		88.27 KB	
(90,384 bytes)		2/28/2003 2:03:31 PM			
explorer.exe	c:\winnt\explorer.exe	1320	8	204800	
1413120	7/14/2003 3:41:27 PM	5.00.3502.5321		237.27 KB	
(242,960 bytes)		2/28/2003 2:03:49 PM			
svchost.exe	c:\winnt\system32\svchost.exe	1520	8		
204800	1413120	7/14/2003 3:41:32 PM		5.00.2134.1	
7.77 KB (7,952 bytes)		12/7/1999 7:00:00 AM			
dllhost.exe	Not Available	1328	8	Not Available	
Not Available		7/14/2003 3:45:20 PM		Unknown	Unknown
Unknown					
inetinfo.exe	c:\winnt\system32\inetrv\inetinfo.exe	1572	8		
204800	1413120	7/14/2003 5:02:45 PM		5.00.0984	14.27 KB
(14,608 bytes)		2/28/2003 2:04:01 PM			
mmc.exe	c:\winnt\system32\mmc.exe	4072	8	204800	
1413120	7/14/2003 5:32:17 PM	5.00.2195.4933		589.27 KB	
(603,408 bytes)		2/28/2003 2:03:37 PM			
rsvp.exe	c:\winnt\system32\rsvp.exe	832	8	204800	
1413120	7/14/2003 5:33:02 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					
wbemprox.dll	1.50.1085.0045	40.08 KB (41,040 bytes)			
2/28/2003 2:03:52 PM		Microsoft Corporation			
c:\winnt\system32\wbem\wbemprox.dll					
mlang.dll	6.00.2800.1106	561.50 KB (574,976 bytes)			8/29/2002
8:14:40 AM		Microsoft Corporation			
c:\winnt\system32\mlang.dll					
rassapi.dll	5.00.2195.5438	14.27 KB (14,608 bytes)			2/28/2003
2:03:44 PM		Microsoft Corporation			
c:\winnt\system32\rassapi.dll					
adsnt.dll	5.00.2195.5206	196.27 KB (200,976 bytes)			2/28/2003
2:03:28 PM		Microsoft Corporation			
c:\winnt\system32\adsnt.dll					
dbghelp.dll	5.00.2195.5242	159.27 KB (163,088 bytes)			
7/22/2002 1:05:04 PM		Microsoft Corporation			
c:\winnt\system32\dbghelp.dll					
localsec.dll	5.00.2195.4663	230.27 KB (235,792 bytes)			
2/28/2003 2:03:37 PM		Microsoft Corporation			
c:\winnt\system32\localsec.dll					

devmgr.dll	5.00.2195.5305	216.77 KB (221,968 bytes)	2/28/2003 2:03:31 PM	Microsoft Corporation	c:\winnt\system32\devmgr.dll
filemgmt.dll	5.00.2195.3649	287.27 KB (294,160 bytes)	2/28/2003 2:03:34 PM	Microsoft Corporation	c:\winnt\system32\filemgmt.dll
pdh.dll	5.00.2195.4952	149.27 KB (152,848 bytes)	2/28/2003 2:03:44 PM	Microsoft Corporation	c:\winnt\system32\pdh.dll
smlogcfg.dll	5.00.2195.3975	278.77 KB (285,456 bytes)	2/28/2003 2:03:46 PM	Microsoft Corporation	c:\winnt\system32\smlogcfg.dll
cabinet.dll	5.00.2147.1	54.77 KB (56,080 bytes)	7:00:00 AM	Microsoft Corporation	c:\winnt\system32\cabinet.dll
msinfo32.dll	5.00.2195.4601	312.27 KB (319,760 bytes)	2/28/2003 2:03:53 PM	Microsoft Corporation	c:\program files\common files\microsoft shared\msinfo\msinfo32.dll
riched20.dll	5.30.23.1209	421.77 KB (431,888 bytes)	2/28/2003 2:03:45 PM	Microsoft Corporation	c:\winnt\system32\riched20.dll
riched32.dll	5.00.2134.1	3.77 KB (3,856 bytes)	7:00:00 AM	Microsoft Corporation	c:\winnt\system32\riched32.dll
els.dll	5.00.2195.4749	153.77 KB (157,456 bytes)	2/28/2003 2:03:33 PM	Microsoft Corporation	c:\winnt\system32\els.dll
ntsmgr.dll	1,0,0,1	427.77 KB (438,032 bytes)	7:00:00 AM	Microsoft Corporation and HighGround Systems, Inc.	c:\winnt\system32\ntsmgr.dll
mmfutil.dll	1.50.1085.0000	32.06 KB (32,829 bytes)	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)	7:00:00 AM	Executive Software International, Inc.	c:\winnt\system32\dfrgres.dll
dfrgsnap.dll	5.00.2195.3649	41.77 KB (42,768 bytes)	2/28/2003 2:03:31 PM	Executive Software International, Inc.	c:\winnt\system32\dfrgsnap.dll
dmdskres.dll	2195.3649.297.3	119.50 KB (122,368 bytes)	2/28/2003 2:03:32 PM	Microsoft Corp., VERITAS Software	c:\winnt\system32\dmdskres.dll
dmutil.dll	2195.3649.297.3	42.27 KB (43,280 bytes)	2/28/2003 2:03:32 PM	VERITAS Software Corp.	c:\winnt\system32\dmutil.dll
ntmsapi.dll	5.00.1948.1	52.27 KB (53,520 bytes)	2/28/2003 2:03:42 PM	Microsoft Corporation	c:\winnt\system32\ntmsapi.dll
dmdskmgr.dll	2195.3649.297.3	159.77 KB (163,600 bytes)	2/28/2003 2:03:32 PM	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	5.00.2195.5352	816.27 KB (835,856 bytes)	2/28/2003 2:03:37 PM	Microsoft Corporation	c:\winnt\system32\mmcmdmgr.dll
mmc.exe	5.00.2195.4933	589.27 KB (603,408 bytes)	2/28/2003 2:03:37 PM	Microsoft Corporation	c:\winnt\system32\mmc.exe
iislog.dll	5.00.0984	76.77 KB (78,608 bytes)	9/17/2002 11:40:50 AM	Microsoft Corporation	c:\winnt\system32\inet\iislog.dll
httpext.dll	5.00.0984	240.27 KB (246,032 bytes)	9/17/2002 11:40:50 AM	Microsoft Corporation	c:\winnt\system32\inet\httpext.dll
rpcproxy.dll	5.00.2195.5419	16.27 KB (16,656 bytes)	2/28/2003 2:03:56 PM	Microsoft Corporation	c:\winnt\system32\rpcproxy\rpcproxy.dll
fpexedll.dll	4.0.2.5526	20.06 KB (20,541 bytes)	2/28/2003 2:03:55 PM	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)	2/28/2003 2:04:01 PM	Microsoft Corporation	c:\winnt\system32\inet\md5filt.dll
gzip.dll	5.00.0984	30.27 KB (30,992 bytes)	2/28/2003 2:04:01 PM	Microsoft Corporation	c:\winnt\system32\inet\gzip.dll
compfilt.dll	5.00.0984	22.77 KB (23,312 bytes)	2/28/2003 2:04:01 PM	Microsoft Corporation	c:\winnt\system32\inet\compfilt.dll
sspifilt.dll	5.00.0984	42.77 KB (43,792 bytes)	2/28/2003 2:04:02 PM	Microsoft Corporation	c:\winnt\system32\inet\sspifilt.dll
iscomlog.dll	5.00.0984	24.27 KB (24,848 bytes)	2/28/2003 2:04:01 PM	Microsoft Corporation	c:\winnt\system32\inet\iscomlog.dll
lonsint.dll	5.00.0984	11.77 KB (12,048 bytes)	2/28/2003 2:04:01 PM	Microsoft Corporation	c:\winnt\system32\inet\lonsint.dll
inetsloc.dll	5.00.0984	20.27 KB (20,752 bytes)	2/28/2003 2:03:35 PM	Microsoft Corporation	c:\winnt\system32\inet\inetsloc.dll
iisfecnv.dll	5.00.0984	7.27 KB (7,440 bytes)	2/12/2003 7:53:06 AM	Microsoft Corporation	c:\winnt\system32\inet\iisfecnv.dll
isatq.dll	5.00.0984	61.27 KB (62,736 bytes)	9/17/2002 11:40:50 AM	Microsoft Corporation	c:\winnt\system32\inet\isatq.dll
infocomm.dll	5.00.0984	240.77 KB (246,544 bytes)	9/17/2002 11:40:50 AM	Microsoft Corporation	c:\winnt\system32\inet\infocomm.dll
w3svc.dll	5.00.0984	341.27 KB (349,456 bytes)	9/17/2002 11:40:52 AM	Microsoft Corporation	c:\winnt\system32\inet\w3svc.dll
security.dll	5.00.2154.1	5.77 KB (5,904 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\security.dll
svcxext.dll	5.00.0984	39.77 KB (40,720 bytes)	2/28/2003 2:04:02 PM	Microsoft Corporation	c:\winnt\system32\inet\svcxext.dll
admexs.dll	5.00.0984	27.77 KB (28,432 bytes)	2/28/2003 2:04:00 PM	Microsoft Corporation	c:\winnt\system32\inet\admexs.dll
wamreg.dll	5.00.0984	45.77 KB (46,864 bytes)	2/28/2003 2:04:02 PM	Microsoft Corporation	c:\winnt\system32\inet\wamreg.dll
metadata.dll	5.00.0984	68.77 KB (70,416 bytes)	2/28/2003 2:04:01 PM	Microsoft Corporation	c:\winnt\system32\inet\metadata.dll
iismap.dll	5.00.0984	55.77 KB (57,104 bytes)	2/28/2003 2:03:35 PM	Microsoft Corporation	c:\winnt\system32\inet\iismap.dll
nsepm.dll	5.00.0984	43.27 KB (44,304 bytes)	2/28/2003 2:04:02 PM	Microsoft Corporation	c:\winnt\system32\inet\nsepm.dll
admwprox.dll	5.00.0984	31.77 KB (32,528 bytes)	2/12/2003 7:53:06 AM	Microsoft Corporation	c:\winnt\system32\inet\admwprox.dll
coadmin.dll	5.00.0984	39.77 KB (40,720 bytes)	2/28/2003 2:04:01 PM	Microsoft Corporation	c:\winnt\system32\inet\coadmin.dll
iisadmin.dll	5.00.0984	15.27 KB (15,632 bytes)	2/28/2003 2:04:01 PM	Microsoft Corporation	c:\winnt\system32\inet\iisadmin.dll
rpref.dll	5.00.0984	4.27 KB (4,368 bytes)	2/28/2003 2:04:02 PM	Microsoft Corporation	c:\winnt\system32\inet\rpref.dll
iisrtl.dll	5.00.0984	119.77 KB (122,640 bytes)	9/17/2002 11:40:50 AM	Microsoft Corporation	c:\winnt\system32\inet\iisrtl.dll
inetinfo.exe	5.00.0984	14.27 KB (14,608 bytes)	2/28/2003 2:04:01 PM	Microsoft Corporation	c:\winnt\system32\inet\inetinfo.exe
tapisrv.dll	5.00.2195.5227	169.27 KB (173,328 bytes)	2/28/2003 2:03:47 PM	Microsoft Corporation	c:\winnt\system32\tapisrv.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	
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	
msacm32.drv	5.00.2134.1	20.77 KB (21,264 bytes)	12/7/1999	7:00:00 AM	Microsoft Corporation	c:\winnt\system32\msacm32.drv	
msi.dll	2.0.2600.1	1.90 MB (1,991,168 bytes)	2/28/2003	2:03:39 PM	Microsoft Corporation	c:\winnt\system32\msi.dll	
powrprof.dll	5.00.3502.5305	13.27 KB (13,584 bytes)	2/28/2003	2:03:44 PM	Microsoft Corporation	c:\winnt\system32\powrprof.dll	
batmeter.dll	5.00.3502.5305	20.27 KB (20,752 bytes)	2/28/2003	2:03:28 PM	Microsoft Corporation	c:\winnt\system32\batmeter.dll	
stobject.dll	5.00.2195.4455	79.27 KB (81,168 bytes)	2/28/2003	2:03:47 PM	Microsoft Corporation	c:\winnt\system32\stobject.dll	
webcheck.dll	6.00.2800.1106	252.00 KB (258,048 bytes)	8/29/2002	8:14:40 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	5.00.3315.4065	55.27 KB (56,592 bytes)	2/28/2003	2:03:41 PM	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	6.00.2800.1106	1002.00 KB (1,026,048 bytes)	8/29/2002	8:14:40 AM	Microsoft Corporation	c:\winnt\system32\browseui.dll	
shdocvw.dll	6.00.2800.1106	1.28 MB (1,338,368 bytes)	8/29/2002	8:14:40 AM	Microsoft Corporation	c:\winnt\system32\shdocvw.dll	
explorer.exe	5.00.3502.5321	237.27 KB (242,960 bytes)	2/28/2003	2:03:49 PM	Microsoft Corporation	c:\winnt\explorer.exe	
dfssvc.exe	5.00.2195.3649	88.27 KB (90,384 bytes)	2/28/2003	2:03:31 PM	Microsoft Corporation	c:\winnt\system32\dfssvc.exe	
winhttp.dll	5.1.2600.1039	(xpsp1.020511-1800)	303.00 KB (310,272 bytes)	2/28/2003	2:04:00 PM	Microsoft Corporation	c:\winnt\system32\winhttp.dll
wininet.dll	6.00.2800.1106	572.00 KB (585,728 bytes)	8/29/2002	8:14:40 AM	Microsoft Corporation	c:\winnt\system32\wininet.dll	
util.dll	5.00.2153.1	25.77 KB (26,384 bytes)	12/7/1999	7:00:00 AM	Microsoft Corporation	c:\winnt\system32\util.dll	
wtsapi32.dll	5.00.2134.1	14.27 KB (14,608 bytes)	12/7/1999	7:00:00 AM	Microsoft Corporation	c:\winnt\system32\wtsapi32.dll	
advpack.dll	6.00.2800.1106	89.00 KB (91,136 bytes)	8/29/2002	8:14:40 AM	Microsoft Corporation	c:\winnt\system32\advpack.dll	
wuaueng.dll	5.4.3628.1	built by: lab04_n	182.50 KB (186,880 bytes)	2/28/2003	2:04:00 PM	Microsoft Corporation	c:\winnt\system32\wuaueng.dll
wuauerv.dll	5.4.3628.1	built by: lab04_n	8.50 KB (8,704 bytes)	2/28/2003	2:04:00 PM	Microsoft Corporation	c:\winnt\system32\wuauerv.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	5.00.2195.4874	70.77 KB (72,464 bytes)	2/28/2003	2:03:42 PM	Microsoft Corporation	c:\winnt\system32\netui0.dll	
ntlanman.dll	5.00.2195.5428	35.27 KB (36,112 bytes)	12/7/1999	7:00:00 AM	Microsoft Corporation	c:\winnt\system32\ntlanman.dll	
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	
ntmarta.dll	5.00.2195.4836	99.77 KB (102,160 bytes)	2/28/2003	2:03:42 PM	Microsoft Corporation	c:\winnt\system32\ntmarta.dll	
provthrd.dll	1.50.1085.0000	68.07 KB (69,708 bytes)	2/12/2003	12:55:49 PM	Microsoft Corporation	c:\winnt\system32\wbem\provthrd.dll	
ntevt.dll	1.50.1085.0072	192.06 KB (196,671 bytes)	2/28/2003	2:03:51 PM	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	1.50.1085.0076	164.07 KB (168,009 bytes)	2/28/2003	2:03:51 PM	Microsoft Corporation	c:\winnt\system32\wbem\framedyn.dll	
cimwin32.dll	1.50.1085.0073	1.04 MB (1,085,520 bytes)	2/28/2003	2:03:51 PM	Microsoft Corporation	c:\winnt\system32\wbem\cimwin32.dll	
wbemsvcs.dll	1.50.1085.0007	40.07 KB (41,036 bytes)	2/28/2003	2:03:52 PM	Microsoft Corporation	c:\winnt\system32\wbem\wbemsvcs.dll	
wbemess.dll	1.50.1085.0074	364.07 KB (372,804 bytes)	2/28/2003	2:03:52 PM	Microsoft Corporation	c:\winnt\system32\wbem\wbemess.dll	
fastprox.dll	1.50.1085.0056	144.08 KB (147,536 bytes)	2/28/2003	2:03:51 PM	Microsoft Corporation	c:\winnt\system32\wbem\fastprox.dll	
wbemcore.dll	1.50.1085.0085	628.07 KB (643,146 bytes)	2/28/2003	2:03:52 PM	Microsoft Corporation	c:\winnt\system32\wbem\wbemcore.dll	
wbemcomn.dll	1.50.1085.0077	692.07 KB (708,675 bytes)	2/28/2003	2:03:52 PM	Microsoft Corporation	c:\winnt\system32\wbem\wbemcomn.dll	
winmgmt.exe	1.50.1085.0070	192.08 KB (196,685 bytes)	2/28/2003	2:03:52 PM	Microsoft Corporation	c:\winnt\system32\wbem\winmgmt.exe	
simptcp.dll	5.00.2134.1	19.27 KB (19,728 bytes)	2/12/2003	7:52:36 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	
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	4.71.2195.1	115.77 KB (118,544 bytes)	2/28/2003	2:03:40 PM	Microsoft Corporation	c:\winnt\system32\mstask.exe	
regsvcs.exe	5.00.2195.3649	65.27 KB (66,832 bytes)	2/28/2003	2:03:45 PM	Microsoft Corporation	c:\winnt\system32\regsvcs.exe	
llsrpc.dll	5.00.2195.4907	47.77 KB (48,912 bytes)	12/7/1999	7:00:00 AM	Microsoft Corporation	c:\winnt\system32\llsrpc.dll	

llsrvr.exe	5.00.2195.4907	81.27 KB (83,216 bytes)	7/22/2002	msjter40.dll	4.00.2927.2	52.27 KB (53,520 bytes)	
1:05:04 PM	Microsoft Corporation			2/28/2003 2:03:40 PM	Microsoft Corporation		
c:\winnt\system32\llsrvr.exe				c:\winnt\system32\msjter40.dll			
rasdlg.dll	5.00.2195.5438	515.77 KB (528,144 bytes)	12/7/1999	mswstr10.dll	4.00.3829.2	600.27 KB (614,672 bytes)	
7:00:00 AM	Microsoft Corporation			2/28/2003 2:03:41 PM	Microsoft Corporation		
c:\winnt\system32\rasdlg.dll				c:\winnt\system32\mswstr10.dll			
netcfgx.dll	5.00.2195.4874	534.77 KB (547,600 bytes)	2/28/2003	msjter40.dll	4.00.6218.0	1.43 MB (1,503,504 bytes)	2/28/2003
2:03:41 PM	Microsoft Corporation			2:03:39 PM	Microsoft Corporation		
c:\winnt\system32\netcfgx.dll				c:\winnt\system32\msjter40.dll			
rasmans.dll	5.00.2195.5436	149.27 KB (152,848 bytes)		msjetoledb40.dll	4.00.5919.0	340.27 KB (348,432 bytes)	
2/28/2003 2:03:44 PM	Microsoft Corporation			2/28/2003 2:03:40 PM	Microsoft Corporation		
c:\winnt\system32\rasmans.dll				c:\winnt\system32\msjetoledb40.dll			
wmi.dll	5.00.2191.1	6.27 KB (6,416 bytes)	12/7/1999 7:00:00 AM	iasrad.dll	5.00.2195.4841	94.77 KB (97,040 bytes)	2/28/2003
Microsoft Corporation				2:03:34 PM	Microsoft Corporation		
c:\winnt\system32\wmi.dll				c:\winnt\system32\iasrad.dll			
netshell.dll	5.00.2195.5431	457.77 KB (468,752 bytes)	2/28/2003	iassam.dll	5.00.2195.5427	98.27 KB (100,624 bytes)	2/28/2003
2:03:42 PM	Microsoft Corporation			2:03:34 PM	Microsoft Corporation		
c:\winnt\system32\netshell.dll				c:\winnt\system32\iassam.dll			
netman.dll	5.00.2195.5282	89.27 KB (91,408 bytes)	2/28/2003	iasads.dll	5.00.2195.5080	73.77 KB (75,536 bytes)	2/28/2003
2:03:42 PM	Microsoft Corporation			2:03:34 PM	Microsoft Corporation		
c:\winnt\system32\netman.dll				c:\winnt\system32\iasads.dll			
comsvcs.dll	2000.2.3497.0	1.37 MB (1,439,504 bytes)		iaspolcy.dll	5.00.2134.1	25.27 KB (25,872 bytes)	
2/28/2003 2:03:30 PM	Microsoft Corporation			12/7/1999 7:00:00 AM	Microsoft Corporation		
c:\winnt\system32\comsvcs.dll				c:\winnt\system32\iaspolcy.dll			
ntmsdba.dll	5.00.2195.5279	169.27 KB (173,328 bytes)		iassvcs.dll	5.00.2195.4916	58.77 KB (60,176 bytes)	2/28/2003
2/28/2003 2:03:42 PM	Microsoft Corporation			2:03:34 PM	Microsoft Corporation		
c:\winnt\system32\ntmsdba.dll				c:\winnt\system32\iassvcs.dll			
sens.dll	5.00.2163.1	36.77 KB (37,648 bytes)	12/7/1999	iasdo.dll	5.00.2195.4115	263.27 KB (269,584 bytes)	2/28/2003
7:00:00 AM	Microsoft Corporation			2:03:34 PM	Microsoft Corporation		
c:\winnt\system32\sens.dll				c:\winnt\system32\iasdo.dll			
iashlpr.dll	5.00.2184.1	33.27 KB (34,064 bytes)	12/7/1999	ntmssvc.dll	5.00.2195.5254	391.77 KB (401,168 bytes)	
7:00:00 AM	Microsoft Corporation			2/28/2003 2:03:42 PM	Microsoft Corporation		
c:\winnt\system32\iashlpr.dll				c:\winnt\system32\ntmssvc.dll			
iasacct.dll	5.00.2195.4115	28.27 KB (28,944 bytes)	2/28/2003	ias.dll	5.00.2134.1	7.27 KB (7,440 bytes)	12/7/1999 7:00:00 AM
2:03:34 PM	Microsoft Corporation			Microsoft Corporation			
c:\winnt\system32\iasacct.dll				c:\winnt\system32\ias.dll			
iasuser.dll	5.00.2195.4609	19.77 KB (20,240 bytes)	12/7/1999	es.dll	2000.2.3497.0	225.27 KB (230,672 bytes)	2/28/2003
7:00:00 AM	Microsoft Corporation			2:03:33 PM	Microsoft Corporation		
c:\winnt\system32\iasuser.dll				c:\winnt\system32\es.dll			
iasnap.dll	5.00.2195.4115	58.77 KB (60,176 bytes)	2/28/2003	mtxoci.dll	2000.2.3497.0	103.77 KB (106,256 bytes)	2/28/2003
2:03:34 PM	Microsoft Corporation			2:03:41 PM	Microsoft Corporation		
c:\winnt\system32\iasnap.dll				c:\winnt\system32\mtxoci.dll			
iaspipe.dll	5.00.2134.1	41.77 KB (42,768 bytes)	12/7/1999	resutils.dll	5.00.2195.5339	39.77 KB (40,720 bytes)	2/28/2003
7:00:00 AM	Microsoft Corporation			2:03:45 PM	Microsoft Corporation		
c:\winnt\system32\iaspipe.dll				c:\winnt\system32\resutils.dll			
expsrv.dll	6.72.9414	371.77 KB (380,688 bytes)	2/28/2003 2:03:33 PM	clusapi.dll	5.00.2195.4678	54.27 KB (55,568 bytes)	2/28/2003
Microsoft Corporation				2:03:29 PM	Microsoft Corporation		
c:\winnt\system32\expsrv.dll				c:\winnt\system32\clusapi.dll			
vbajet32.dll	6.1.8268	30.27 KB (30,992 bytes)	2/28/2003	msvcp50.dll	5.00.7051	552.50 KB (565,760 bytes)	12/7/1999
2:03:48 PM	Microsoft Corporation			7:00:00 AM	Microsoft Corporation		
c:\winnt\system32\vbajet32.dll				c:\winnt\system32\msvcp50.dll			
msjtes40.dll	4.00.5914.0	236.27 KB (241,936 bytes)		xolehlp.dll	1999.9.3421.3	17.27 KB (17,680 bytes)	2/12/2003
2/28/2003 2:03:40 PM	Microsoft Corporation			7:52:40 AM	Microsoft Corporation		
c:\winnt\system32\msjtes40.dll				c:\winnt\system32\xolehlp.dll			
oledb32r.dll	2.70.9001.0	built by: Lab06_N(dagbuild) 64.00 KB (65,536 bytes)	5/29/2003 1:59:43 PM	msdtclog.dll	2000.2.3497.0	86.77 KB (88,848 bytes)	
Microsoft Corporation				2/28/2003 2:03:38 PM	Microsoft Corporation		
c:\program files\common files\system\ole db\oledb32r.dll				c:\winnt\system32\msdtclog.dll			
comdlg32.dll	5.00.3315.3727	221.27 KB (226,576 bytes)		mtxclu.dll	2000.2.3497.0	51.27 KB (52,496 bytes)	2/28/2003
12/7/1999 7:00:00 AM	Microsoft Corporation			2:03:41 PM	Microsoft Corporation		
c:\winnt\system32\comdlg32.dll				c:\winnt\system32\mtxclu.dll			
msdart.dll	2.71.9031.4	built by: Lab06_N(dagbuild) 124.00 KB (126,976 bytes)	5/29/2003 1:59:43 PM	msdtcprx.dll	2000.2.3497.0	683.77 KB (700,176 bytes)	
Microsoft Corporation				2/28/2003 2:03:38 PM	Microsoft Corporation		
c:\winnt\system32\msdart.dll				c:\winnt\system32\msdtcprx.dll			
oledb32.dll	2.71.9031.4	built by: Lab06_N(dagbuild) 408.00 KB (417,792 bytes)	5/29/2003 1:59:43 PM	txfaux.dll	2000.2.3497.0	383.27 KB (392,464 bytes)	2/28/2003
Microsoft Corporation				2:03:48 PM	Microsoft Corporation		
c:\program files\common files\system\ole db\oledb32.dll				c:\winnt\system32\txfaux.dll			
msjint40.dll	4.00.2927.2	148.27 KB (151,824 bytes)		msdtctm.dll	2000.2.3497.0	1.08 MB (1,128,208 bytes)	
2/28/2003 2:03:40 PM	Microsoft Corporation			2/28/2003 2:03:38 PM	Microsoft Corporation		
c:\winnt\system32\msjint40.dll				c:\winnt\system32\msdtctm.dll			



msdtc.exe	1999.9.3421.3	6.77 KB (6,928 bytes)	2/12/2003 7:52:40 AM	scecli.dll	5.00.2195.4874	109.27 KB (111,888 bytes)	2/28/2003 2:03:45 PM
Microsoft Corporation				Microsoft Corporation			
inetpp.dll	5.00.2195.4299	64.27 KB (65,808 bytes)	2/28/2003 2:03:35 PM	c:\winnt\system32\scecli.dll			
Microsoft Corporation				atld.dll	3.00.9435	73.06 KB (74,810 bytes)	2/28/2003 2:03:28 PM
c:\winnt\system32\inetpp.dll				Microsoft Corporation			
win32spl.dll	5.00.2195.5201	92.27 KB (94,480 bytes)	12/7/1999 7:00:00 AM	c:\winnt\system32\atld.dll			
Microsoft Corporation				certcli.dll	5.00.2195.3649	130.27 KB (133,392 bytes)	2/28/2003 2:03:29 PM
c:\winnt\system32\win32spl.dll				Microsoft Corporation			
usbmon.dll	5.00.2195.4299	11.27 KB (11,536 bytes)	2/28/2003 2:03:48 PM	c:\winnt\system32\certcli.dll			
Microsoft Corporation				esent.dll	6.0.3940.25	1.09 MB (1,137,936 bytes)	2/28/2003 2:03:33 PM
c:\winnt\system32\usbmon.dll				Microsoft Corporation			
tcpmon.dll	5.00.2195.4299	40.77 KB (41,744 bytes)	2/28/2003 2:03:47 PM	c:\winnt\system32\esent.dll			
Microsoft Corporation				ntdsatq.dll	5.00.2195.5246	31.27 KB (32,016 bytes)	2/28/2003 2:03:42 PM
c:\winnt\system32\tcpmon.dll				Microsoft Corporation			
pjlmon.dll	5.00.2165.1	12.77 KB (13,072 bytes)	11/30/1999 6:39:36 PM	c:\winnt\system32\ntdsatq.dll			
Microsoft Corporation				ntdsa.dll	5.00.2195.5438	1002.27 KB (1,026,320 bytes)	2/28/2003 2:03:42 PM
c:\winnt\system32\pjlmon.dll				Microsoft Corporation			
cnbjmon.dll	5.00.2134.1	43.77 KB (44,816 bytes)	11/30/1999 6:38:48 PM	c:\winnt\system32\ntdsa.dll			
Microsoft Corporation				kdcsvc.dll	5.00.2195.5246	141.77 KB (145,168 bytes)	2/28/2003 2:03:36 PM
c:\winnt\system32\cnbjmon.dll				Microsoft Corporation			
localspl.dll	5.00.2195.5423	250.27 KB (256,272 bytes)	12/7/1999 7:00:00 AM	c:\winnt\system32\kdcsvc.dll			
Microsoft Corporation				sfmapi.dll	5.00.2134.1	38.77 KB (39,696 bytes)	12/7/1999 7:00:00 AM
c:\winnt\system32\localspl.dll				Microsoft Corporation			
spoolss.dll	5.00.2195.5400	61.77 KB (63,248 bytes)	2/12/2003 7:39:06 AM	c:\winnt\system32\sfmapi.dll			
Microsoft Corporation				rassfm.dll	5.00.2195.4874	21.27 KB (21,776 bytes)	2/28/2003 2:03:45 PM
c:\winnt\system32\spoolss.dll				Microsoft Corporation			
spoolsv.exe	5.00.2195.4299	44.27 KB (45,328 bytes)	2/12/2003 7:39:06 AM	c:\winnt\system32\rassfm.dll			
Microsoft Corporation				rsabase.dll	5.00.2195.3839	128.27 KB (131,344 bytes)	7/22/2002 1:05:04 PM
c:\winnt\system32\spoolsv.exe				Microsoft Corporation			
rpss.dll	5.00.2195.5429	231.27 KB (236,816 bytes)	2/28/2003 2:03:45 PM	c:\winnt\system32\rsabase.dll			
Microsoft Corporation				schannel.dll	5.00.2195.5284	139.27 KB (142,608 bytes)	5/4/2001 1:05:02 PM
c:\winnt\system32\rpss.dll				Microsoft Corporation			
svchost.exe	5.00.2134.1	7.77 KB (7,952 bytes)	12/7/1999 7:00:00 AM	c:\winnt\system32\schannel.dll			
Microsoft Corporation				netlogon.dll	5.00.2195.5400	362.77 KB (371,472 bytes)	2/28/2003 2:03:42 PM
c:\winnt\system32\svchost.exe				Microsoft Corporation			
rdpwsx.dll	5.00.2195.5243	97.90 KB (100,248 bytes)	2/28/2003 2:03:45 PM	c:\winnt\system32\netlogon.dll			
Microsoft Corporation				kerberos.dll	5.00.2195.5246	202.77 KB (207,632 bytes)	2/28/2003 2:03:36 PM
c:\winnt\system32\rdpwsx.dll				Microsoft Corporation			
ntsapi.dll	5.00.2195.4907	6.77 KB (6,928 bytes)	12/7/1999 7:00:00 AM	c:\winnt\system32\kerberos.dll			
Microsoft Corporation				msprivs.dll	5.00.2154.1	41.50 KB (42,496 bytes)	12/7/1999 7:00:00 AM
c:\winnt\system32\ntsapi.dll				Microsoft Corporation			
mstlsapi.dll	5.00.2195.3895	25.77 KB (26,384 bytes)	2/28/2003 2:03:41 PM	c:\winnt\system32\msprivs.dll			
Microsoft Corporation				samsrv.dll	5.00.2195.5201	374.27 KB (383,248 bytes)	12/7/1999 7:00:00 AM
c:\winnt\system32\mstlsapi.dll				Microsoft Corporation			
icaapi.dll	5.00.2195.3895	122.77 KB (125,712 bytes)	2/28/2003 2:03:34 PM	c:\winnt\system32\samsrv.dll			
Microsoft Corporation				lsasrv.dll	5.00.2195.5430	500.27 KB (512,272 bytes)	12/7/1999 7:00:00 AM
c:\winnt\system32\icaapi.dll				Microsoft Corporation			
regapi.dll	5.00.2195.5201	35.27 KB (36,112 bytes)	2/28/2003 2:03:45 PM	c:\winnt\system32\lsasrv.dll			
Microsoft Corporation				lsass.exe	5.00.2195.5430	32.77 KB (33,552 bytes)	12/7/1999 7:00:00 AM
c:\winnt\system32\regapi.dll				Microsoft Corporation			
termsrv.exe	5.00.2195.5276	138.77 KB (142,096 bytes)	2/28/2003 2:03:47 PM	c:\winnt\system32\lsass.exe			
Microsoft Corporation				wmicore.dll	5.00.2195.3649	72.27 KB (74,000 bytes)	2/28/2003 2:03:49 PM
c:\winnt\system32\termsrv.exe				Microsoft Corporation			
iissuba.dll	5.00.0984	9.77 KB (10,000 bytes)	12/7/1999 7:00:00 AM	c:\winnt\system32\wmicore.dll			
Microsoft Corporation				rasadhlp.dll	5.00.2168.1	7.27 KB (7,440 bytes)	12/7/1999 7:00:00 AM
c:\winnt\system32\iissuba.dll				Microsoft Corporation			
dssenh.dll	5.00.2195.3665	142.77 KB (146,192 bytes)	2/28/2003 2:03:57 PM	c:\winnt\system32\rasadhlp.dll			
Microsoft Corporation				winnr.dll	5.00.2160.1	18.77 KB (19,216 bytes)	12/7/1999 7:00:00 AM
c:\winnt\system32\dssenh.dll				Microsoft Corporation			
oakley.dll	5.00.2195.5326	382.27 KB (391,440 bytes)	2/28/2003 2:03:43 PM	c:\winnt\system32\winnr.dll			
Microsoft Corporation				rnr20.dll	5.00.2195.4874	35.77 KB (36,624 bytes)	2/28/2003 2:03:45 PM
c:\winnt\system32\oakley.dll				Microsoft Corporation			
mfc42u.dll	6.00.8665.0	972.05 KB (995,384 bytes)	12/7/1999 7:00:00 AM	c:\winnt\system32\rnr20.dll			
Microsoft Corporation				wshtepip.dll	5.00.2195.4874	17.27 KB (17,680 bytes)	2/28/2003 2:03:49 PM
c:\winnt\system32\mfc42u.dll				Microsoft Corporation			
polagent.dll	5.00.2195.5428	94.77 KB (97,040 bytes)	2/28/2003 2:03:44 PM	c:\winnt\system32\wshtepip.dll			
Microsoft Corporation				msafd.dll	5.00.2195.4874	103.27 KB (105,744 bytes)	2/28/2003 2:03:38 PM
c:\winnt\system32\polagent.dll				Microsoft Corporation			
				c:\winnt\system32\msafd.dll			

mwssock.dll	5.00.2195.4874	62.77 KB (64,272 bytes)		dhcpcsvc.dll	5.00.2195.4874	87.77 KB (89,872 bytes)	
2/28/2003 2:03:41 PM	Microsoft Corporation			12/7/1999 7:00:00 AM	Microsoft Corporation		
c:\winnt\system32\mwssock.dll				c:\winnt\system32\dhcpcsvc.dll			
msgsvc.dll	5.00.2195.4874	34.77 KB (35,600 bytes)	12/7/1999	eventlog.dll	5.00.2195.5336	44.27 KB (45,328 bytes)	
7:00:00 AM	Microsoft Corporation			2/28/2003 2:03:33 PM	Microsoft Corporation		
c:\winnt\system32\msgsvc.dll				c:\winnt\system32\eventlog.dll			
alrsvc.dll	5.00.2134.1	17.77 KB (18,192 bytes)	12/7/1999	ntdsapi.dll	5.00.2195.4827	56.27 KB (57,616 bytes)	2/28/2003
7:00:00 AM	Microsoft Corporation			2:03:42 PM	Microsoft Corporation		
c:\winnt\system32\alrsvc.dll				c:\winnt\system32\ntdsapi.dll			
trkwks.dll	5.00.2195.4874	88.77 KB (90,896 bytes)	2/28/2003	scesrv.dll	5.00.2195.5316	242.77 KB (248,592 bytes)	2/28/2003
2:03:48 PM	Microsoft Corporation			2:03:45 PM	Microsoft Corporation		
c:\winnt\system32\trkwks.dll				c:\winnt\system32\scesrv.dll			
seclogon.dll	5.00.2195.5201	17.27 KB (17,680 bytes)		umpnpgmgr.dll	5.00.2182.1	86.27 KB (88,336 bytes)	
2/28/2003 2:03:45 PM	Microsoft Corporation			12/7/1999 7:00:00 AM	Microsoft Corporation		
c:\winnt\system32\seclogon.dll				c:\winnt\system32\umpnpgmgr.dll			
psbase.dll	5.00.2195.4822	111.77 KB (114,448 bytes)	2/28/2003	services.exe	5.00.2195.3940	86.77 KB (88,848 bytes)	
2:03:44 PM	Microsoft Corporation			12/7/1999 7:00:00 AM	Microsoft Corporation		
c:\winnt\system32\psbase.dll				c:\winnt\system32\services.exe			
cryptsvc.dll	5.00.2195.4368	73.27 KB (75,024 bytes)		clbcatq.dll	2000.2.3497.0	497.77 KB (509,712 bytes)	2/28/2003
2/28/2003 2:03:31 PM	Microsoft Corporation			2:03:29 PM	Microsoft Corporation		
c:\winnt\system32\cryptsvc.dll				c:\winnt\system32\clbcatq.dll			
cryptdll.dll	5.00.2135.1	41.27 KB (42,256 bytes)	12/7/1999	oleaut32.dll	2.40.4518	612.27 KB (626,960 bytes)	12/7/1999
7:00:00 AM	Microsoft Corporation			7:00:00 AM	Microsoft Corporation		
c:\winnt\system32\cryptdll.dll				c:\winnt\system32\oleaut32.dll			
wkssvc.dll	5.00.2195.4874	95.27 KB (97,552 bytes)	12/7/1999	cscui.dll	5.00.2195.4104	233.77 KB (239,376 bytes)	2/28/2003
7:00:00 AM	Microsoft Corporation			2:03:31 PM	Microsoft Corporation		
c:\winnt\system32\wkssvc.dll				c:\winnt\system32\cscui.dll			
srsvcs.dll	5.00.2195.5400	81.77 KB (83,728 bytes)	12/7/1999	msv1_0.dll	5.00.2195.4745	112.27 KB (114,960 bytes)	12/7/1999
7:00:00 AM	Microsoft Corporation			7:00:00 AM	Microsoft Corporation		
c:\winnt\system32\srsvcs.dll				c:\winnt\system32\msv1_0.dll			
cfgmgr32.dll	5.00.2134.1	16.77 KB (17,168 bytes)		winspool.drv	5.00.2195.5225	111.27 KB (113,936 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	2195.3649.297.3	12.27 KB (12,560 bytes)		winscard.dll	5.00.2134.1	77.27 KB (79,120 bytes)	
2/28/2003 2:03:32 PM	VERITAS Software Corp.			12/7/1999 7:00:00 AM	Microsoft Corporation		
c:\winnt\system32\dmserver.dll				c:\winnt\system32\winscard.dll			
lmhsvc.dll	5.00.2195.4874	9.77 KB (10,000 bytes)	12/7/1999	wlnotify.dll	5.00.2195.5377	54.27 KB (55,568 bytes)	
7:00:00 AM	Microsoft Corporation			2/28/2003 2:03:49 PM	Microsoft Corporation		
c:\winnt\system32\lmhsvc.dll				c:\winnt\system32\wlnotify.dll			
dnssrslvr.dll	5.00.2195.5354	89.77 KB (91,920 bytes)		cscdll.dll	5.00.2195.5434	98.77 KB (101,136 bytes)	2/28/2003
2/28/2003 2:03:32 PM	Microsoft Corporation			2:03:31 PM	Microsoft Corporation		
c:\winnt\system32\dnssrslvr.dll				c:\winnt\system32\cscdll.dll			
tapi32.dll	5.00.2182.1	123.27 KB (126,224 bytes)	12/7/1999	wdmaud.drv	5.00.2195.3649	21.27 KB (21,776 bytes)	
7:00:00 AM	Microsoft Corporation			2/28/2003 2:03:49 PM	Microsoft Corporation		
c:\winnt\system32\tapi32.dll				c:\winnt\system32\wdmaud.drv			
rasman.dll	5.00.2195.5292	54.77 KB (56,080 bytes)	12/7/1999	lz32.dll	5.00.2134.1	9.77 KB (10,000 bytes)	12/7/1999
7:00:00 AM	Microsoft Corporation			7:00:00 AM	Microsoft Corporation		
c:\winnt\system32\rasman.dll				c:\winnt\system32\lz32.dll			
rasapi32.dll	5.00.2195.5438	191.77 KB (196,368 bytes)		version.dll	5.00.2134.1	15.77 KB (16,144 bytes)	12/7/1999
12/7/1999 7:00:00 AM	Microsoft Corporation			7:00:00 AM	Microsoft Corporation		
c:\winnt\system32\rasapi32.dll				c:\winnt\system32\version.dll			
rtutils.dll	5.00.2168.1	43.77 KB (44,816 bytes)	12/7/1999	rsaenh.dll	5.00.2195.3839	130.77 KB (133,904 bytes)	2/28/2003
7:00:00 AM	Microsoft Corporation			2:03:57 PM	Microsoft Corporation		
c:\winnt\system32\rtutils.dll				c:\winnt\system32\rsaenh.dll			
adslpcc.dll	5.00.2195.5400	127.77 KB (130,832 bytes)	2/28/2003	mecat32.dll	5.131.2134.1	7.77 KB (7,952 bytes)	12/7/1999
2:03:27 PM	Microsoft Corporation			7:00:00 AM	Microsoft Corporation		
c:\winnt\system32\adslpcc.dll				c:\winnt\system32\mec32.dll			
activeds.dll	5.00.2195.5312	175.27 KB (179,472 bytes)		ole32.dll	5.00.2195.5400	968.27 KB (991,504 bytes)	2/28/2003
2/28/2003 2:03:25 PM	Microsoft Corporation			2:03:43 PM	Microsoft Corporation		
c:\winnt\system32\activeds.dll				c:\winnt\system32\ole32.dll			
mprapi.dll	5.00.2181.1	79.27 KB (81,168 bytes)	12/7/1999	imagehlp.dll	5.00.2195.5242	125.77 KB (128,784 bytes)	
7:00:00 AM	Microsoft Corporation			5/4/2001 1:05:02 PM	Microsoft Corporation		
c:\winnt\system32\mprapi.dll				c:\winnt\system32\imagehlp.dll			
iphlpapi.dll	5.00.2195.2	68.27 KB (69,904 bytes)		msasn1.dll	5.00.2195.4067	51.27 KB (52,496 bytes)	12/7/1999
2/28/2003 2:03:35 PM	Microsoft Corporation			7:00:00 AM	Microsoft Corporation		
c:\winnt\system32\iphlpapi.dll				c:\winnt\system32\msasn1.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							

crypt32.dll	5.131.2195.4558	464.27 KB (475,408 bytes)	2/28/2003	2:03:30 PM	Microsoft Corporation	userenv.dll	5.00.2195.5425	363.77 KB (372,496 bytes)	12/7/1999	7:00:00 AM	Microsoft Corporation
c:\winnt\system32\crypt32.dll						c:\winnt\system32\userenv.dll					
wintrust.dll	5.131.2195.3775	162.27 KB (166,160 bytes)	2/28/2003	2:03:49 PM	Microsoft Corporation	user32.dll	5.00.2195.4314	395.77 KB (405,264 bytes)	12/7/1999	7:00:00 AM	Microsoft Corporation
c:\winnt\system32\wintrust.dll						c:\winnt\system32\user32.dll					
mpr.dll	5.00.2195.3649	53.77 KB (55,056 bytes)	2/28/2003	2:03:37 PM	Microsoft Corporation	gdi32.dll	5.00.2195.5252	228.77 KB (234,256 bytes)	12/7/1999	7:00:00 AM	Microsoft Corporation
c:\winnt\system32\mpr.dll						c:\winnt\system32\gdi32.dll					
shlwapi.dll	6.00.2800.1106	386.00 KB (395,264 bytes)	8/29/2002	8:14:40 AM	Microsoft Corporation	rpert4.dll	5.00.2195.5419	440.27 KB (450,832 bytes)	2/28/2003	2:03:45 PM	Microsoft Corporation
c:\winnt\system32\shlwapi.dll						c:\winnt\system32\rpert4.dll					
shell32.dll	5.00.3502.5436	2.26 MB (2,374,416 bytes)	2/28/2003	2:03:46 PM	Microsoft Corporation	advapi32.dll	5.00.2195.5385	358.77 KB (367,376 bytes)	12/7/1999	7:00:00 AM	Microsoft Corporation
c:\winnt\system32\shell32.dll						c:\winnt\system32\advapi32.dll					
msgina.dll	5.00.2195.4733	324.77 KB (332,560 bytes)	12/7/1999	7:00:00 AM	Microsoft Corporation	kernel32.dll	5.00.2195.5400	716.77 KB (733,968 bytes)	12/7/1999	7:00:00 AM	Microsoft Corporation
c:\winnt\system32\msgina.dll						c:\winnt\system32\kernel32.dll					
comctl32.dll	5.81	517.27 KB (529,680 bytes)	8/29/2002	8:14:40 AM	Microsoft Corporation	msvcrt.dll	6.10.9359.0	284.05 KB (290,869 bytes)	7/22/2002	1:05:04 PM	Microsoft Corporation
c:\winnt\system32\comctl32.dll						c:\winnt\system32\msvcrt.dll					
setupapi.dll	5.00.2195.5400	553.77 KB (567,056 bytes)	12/7/1999	7:00:00 AM	Microsoft Corporation	winlogon.exe	5.00.2195.5386	174.77 KB (178,960 bytes)	2/28/2003	2:03:49 PM	Microsoft Corporation
c:\winnt\system32\setupapi.dll						c:\winnt\system32\winlogon.exe					
winmm.dll	5.00.2161.1	184.77 KB (189,200 bytes)	12/7/1999	7:00:00 AM	Microsoft Corporation	sfcfiles.dll	5.00.2195.5426	951.27 KB (974,096 bytes)	2/28/2003	2:03:46 PM	Microsoft Corporation
c:\winnt\system32\winmm.dll						c:\winnt\system32\sfcfiles.dll					
winsta.dll	5.00.2195.4655	36.77 KB (37,648 bytes)	2/28/2003	2:03:49 PM	Microsoft Corporation	ntdll.dll	5.00.2195.5400	479.27 KB (490,768 bytes)	5/4/2001	1:05:02 PM	Microsoft Corporation
c:\winnt\system32\winsta.dll						c:\winnt\system32\ntdll.dll					
wsock32.dll	5.00.2195.4874	21.27 KB (21,776 bytes)	2/28/2003	2:03:49 PM	Microsoft Corporation	smss.exe	5.00.2195.5382	44.77 KB (45,840 bytes)	12/7/1999	7:00:00 AM	Microsoft Corporation
c:\winnt\system32\wsock32.dll						c:\winnt\system32\smss.exe					
dnsapi.dll	5.00.2195.5354	131.27 KB (134,416 bytes)	2/28/2003	2:03:32 PM	Microsoft Corporation	[Services]					
c:\winnt\system32\dnsapi.dll						Display Name	Name	State	Start Mode	Service Type	
wldap32.dll	5.00.2195.5400	158.77 KB (162,576 bytes)	2/28/2003	2:03:49 PM	Microsoft Corporation	Path	Error Control		Start Name	Tag ID	
c:\winnt\system32\wldap32.dll						Alerter	Alerter	Running	Auto	Share	Process
ws2help.dll	5.00.2134.1	17.77 KB (18,192 bytes)	12/7/1999	7:00:00 AM	Microsoft Corporation	c:\winnt\system32\services.exe		Normal	LocalSystem		0
c:\winnt\system32\ws2help.dll						Application Management		AppMgmt	Stopped	Manual	Share
ws_32.dll	5.00.2195.4874	66.77 KB (68,368 bytes)	2/28/2003	2:03:49 PM	Microsoft Corporation	Process	c:\winnt\system32\services.exe	Normal	LocalSystem		0
c:\winnt\system32\ws_32.dll						Background Intelligent Transfer Service		BITS	Stopped	Manual	
samlib.dll	5.00.2195.4827	49.77 KB (50,960 bytes)	12/7/1999	7:00:00 AM	Microsoft Corporation	Share Process	c:\winnt\system32\svchost.exe -k bitsgroup	Normal	LocalSystem		0
c:\winnt\system32\samlib.dll						Computer Browser		Browser	Stopped	Manual	Share Process
netrap.dll	5.00.2134.1	11.27 KB (11,536 bytes)	12/7/1999	7:00:00 AM	Microsoft Corporation	c:\winnt\system32\services.exe		Normal	LocalSystem		0
c:\winnt\system32\netrap.dll						Indexing Service		cisvc	Stopped	Manual	Share Process
netapi32.dll	5.00.2195.5427	305.27 KB (312,592 bytes)	2/28/2003	2:03:41 PM	Microsoft Corporation	c:\winnt\system32\cisvc.exe		Normal	LocalSystem		0
c:\winnt\system32\netapi32.dll						ClipBook		ClipSrv	Stopped	Manual	Own Process
profmap.dll	5.00.2181.1	29.27 KB (29,968 bytes)	12/7/1999	7:00:00 AM	Microsoft Corporation	c:\winnt\system32\clipsrv.exe		Normal	LocalSystem		0
c:\winnt\system32\profmap.dll						Distributed File System		Dfs	Running	Auto	Own
secur32.dll	5.00.2195.4587	47.27 KB (48,400 bytes)	2/28/2003	2:03:45 PM	Microsoft Corporation	Process	c:\winnt\system32\dfssvc.exe	Normal	LocalSystem		0
c:\winnt\system32\secur32.dll						DHCP Client		Dhcp	Running	Auto	Share Process
sfc.dll	5.00.2195.3649	92.11 KB (94,320 bytes)	2/28/2003	2:03:45 PM	Microsoft Corporation	c:\winnt\system32\services.exe		Normal	LocalSystem		0
c:\winnt\system32\sfc.dll						Logical Disk Manager Administrative Service		dmadmin	Stopped	Manual	Share Process
nddeapi.dll	5.00.2195.4509	15.77 KB (16,144 bytes)	2/28/2003	2:03:41 PM	Microsoft Corporation	Manual	Share Process	c:\winnt\system32\dmadmin.exe /com			
c:\winnt\system32\nddeapi.dll						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		Remote Procedure Call (RPC) Locator	RpcLocator	Stopped	
c:\winnt\system32\faxsvc.exe	Normal	LocalSystem	0			Manual	Own Process	c:\winnt\system32\locator.exe	Normal
Internet Authentication Service	IAS	Running	Auto	Share		LocalSystem	0		
Process	c:\winnt\system32\svchost.exe -k netsvcs	Normal				Remote Procedure Call (RPC)	RpcSs	Running	Auto
LocalSystem	0					Process	c:\winnt\system32\svchost.exe -k rpsvc	Normal	
IIS Admin Service	IISADMIN	Running	Auto	Share		LocalSystem	0		
Process	c:\winnt\system32\inetrv\inetinfo.exe	Normal				QoS RSVP	RSVP	Running	Manual
LocalSystem	0					c:\winnt\system32\rsvp.exe -s	Normal	LocalSystem	0
Intersite Messaging	IsmServ	Stopped	Disabled	Own Process		Security Accounts Manager	SamSs	Running	Auto
c:\winnt\system32\ismsserv.exe	Normal	LocalSystem	0			Process	c:\winnt\system32\lsass.exe	Normal	LocalSystem
Kerberos Key Distribution Center	kdc	Stopped	Disabled			Smart Card Helper	SCardDrv	Stopped	Manual
Share Process	c:\winnt\system32\lsass.exe	Normal				c:\winnt\system32\scardsvr.exe	Ignore	LocalSystem	0
LocalSystem	0					Smart Card	SCardSvr	Stopped	Manual
Server	lanmanserver	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
Workstation	lanmanworkstation	Running	Auto	Share		c:\winnt\system32\mstask.exe	Normal	LocalSystem	0
Process	c:\winnt\system32\services.exe	Normal	LocalSystem	0		RunAs Service	seclogon	Running	Auto
License Logging Service	LicenseService	Running	Auto			c:\winnt\system32\services.exe	Ignore	LocalSystem	0
Own Process	c:\winnt\system32\llssrv.exe	Normal				System Event Notification	SENS	Running	Auto
LocalSystem	0					Process	c:\winnt\system32\svchost.exe -k netsvcs	Normal	
TCP/IP NetBIOS Helper Service	LmHosts	Running	Auto	Share		LocalSystem	0		
Process	c:\winnt\system32\services.exe	Normal	LocalSystem	0		Internet Connection Sharing	SharedAccess	Stopped	Manual
Messenger Messenger	Running	Auto	Share Process			Share Process	c:\winnt\system32\svchost.exe -k netsvcs	Normal	
c:\winnt\system32\services.exe	Normal	LocalSystem	0			LocalSystem	0		
NetMeeting Remote Desktop Sharing	mnmsrvc	Stopped	Manual			Simple TCP/IP Services	SimpTcp	Running	Auto
Own Process	c:\winnt\system32\mnmsrvc.exe	Normal				Process	c:\winnt\system32\tcpvcs.exe	Normal	LocalSystem
LocalSystem	0					Print Spooler	Spooler	Running	Auto
Distributed Transaction Coordinator	MSDTC	Running	Auto			c:\winnt\system32\spoolsv.exe	Normal	LocalSystem	0
Own Process	c:\winnt\system32\msdtc.exe	Normal				Performance Logs and Alerts	SysmonLog	Stopped	Auto
LocalSystem	0					Own Process	c:\winnt\system32\smlogsvc.exe	Normal	
Windows Installer	MSIServer	Stopped	Manual	Share Process		LocalSystem	0		
c:\winnt\system32\msiexec.exe /v	Normal	LocalSystem	0			Telephony TapiSrv	Running	Manual	Share Process
Network DDE	NetDDE	Stopped	Manual	Share Process		c:\winnt\system32\svchost.exe -k tapisrv	Normal	LocalSystem	0
c:\winnt\system32\netdde.exe	Normal	LocalSystem	0			Terminal Services	TermService	Running	Auto
Network DDE DSDM	NetDDEdsdm	Stopped	Manual	Share		Process	c:\winnt\system32\termsrv.exe	Normal	LocalSystem
Process	c:\winnt\system32\netdde.exe	Normal	LocalSystem	0		Telnet	TlntSvr	Stopped	Manual
Net Logon Netlogon	Stopped	Manual	Share Process			c:\winnt\system32\tlntsvr.exe	Normal	LocalSystem	0
c:\winnt\system32\lsass.exe	Normal	LocalSystem	0			Distributed Link Tracking Server	TrkSvr	Stopped	Manual
Network Connections Netman	Running	Manual	Share Process			Process	c:\winnt\system32\services.exe	Normal	LocalSystem
c:\winnt\system32\svchost.exe -k netsvcs	Normal	LocalSystem	0			Distributed Link Tracking Client	TrkWks	Running	Auto
File Replication	NtFrs	Stopped	Manual	Own Process		Process	c:\winnt\system32\services.exe	Normal	LocalSystem
c:\winnt\system32\ntfrs.exe	Ignore	LocalSystem	0			Uninterruptible Power Supply	UPS	Stopped	Manual
NT LM Security Support Provider	NtLmSsp	Stopped	Manual			Process	c:\winnt\system32\ups.exe	Normal	LocalSystem
Share Process	c:\winnt\system32\lsass.exe	Normal				Utility Manager	UtilMan	Stopped	Manual
LocalSystem	0					c:\winnt\system32\utilman.exe	Normal	LocalSystem	0
Removable Storage	NtmsSvc	Running	Auto	Share Process		Windows Time	W32Time	Stopped	Manual
c:\winnt\system32\svchost.exe -k netsvcs	Normal	LocalSystem	0			c:\winnt\system32\services.exe	Normal	LocalSystem	0
Plug and Play	PlugPlay	Running	Auto	Share Process		World Wide Web Publishing Service	W3SVC	Running	Auto
c:\winnt\system32\services.exe	Normal	LocalSystem	0			Share Process	c:\winnt\system32\inetrv\inetinfo.exe	Normal	
IPSEC Policy Agent	PolicyAgent	Running	Auto	Share		LocalSystem	0		
Process	c:\winnt\system32\lsass.exe	Normal	LocalSystem	0		Windows Management Instrumentation	WinMgmt	Running	Auto
Protected Storage	ProtectedStorage	Running	Auto	Share		Own Process	c:\winnt\system32\wbem\winmgmt.exe	Ignore	
Process	c:\winnt\system32\services.exe	Normal	LocalSystem	0		LocalSystem	0		
Remote Access Auto Connection Manager	RasAuto	Stopped	Manual			Windows Management Instrumentation Driver Extensions	Wmi		
Share Process	c:\winnt\system32\svchost.exe -k netsvcs	Normal				Running	Manual	Share Process	c:\winnt\system32\services.exe
LocalSystem	0					Normal	LocalSystem	0	
Remote Access Connection Manager	RasMan	Stopped	Manual			Automatic Updates	wuauerv	Running	Auto
Share Process	c:\winnt\system32\svchost.exe -k netsvcs	Normal				c:\winnt\system32\svchost.exe -k wugroup	Normal	LocalSystem	0
LocalSystem	0								
Routing and Remote Access	RemoteAccess	Stopped	Disabled			[Program Groups]			
Share Process	c:\winnt\system32\svchost.exe -k netsvcs	Normal				Group Name	Name	User Name	
LocalSystem	0					Accessories	Default User:Accessories	Default User	
Remote Registry Service	RemoteRegistry	Running	Auto			Accessories\Accessibility	Default User:Accessories\Accessibility	Default User	
Own Process	c:\winnt\system32\regsvc.exe	Normal				Accessories\Entertainment	Default User:Accessories\Entertainment	Default User	
LocalSystem	0								
Remote Command Service	RMSYS	Running	Auto	Own					
Process	c:\program files\benchcraft\rsys.exe	Normal							
.\Administrator	0								

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  
 Startup All Users:Startup All Users  
 Accessories FCLIENT10\Administrator:Accessories  
 FCLIENT10\Administrator  
 Accessories\Accessibility  
 FCLIENT10\Administrator:Accessories\Accessibility  
 FCLIENT10\Administrator  
 Accessories\Entertainment  
 FCLIENT10\Administrator:Accessories\Entertainment  
 FCLIENT10\Administrator  
 Accessories\System Tools  
 FCLIENT10\Administrator:Accessories\System Tools  
 FCLIENT10\Administrator  
 Administrative Tools FCLIENT10\Administrator:Administrative Tools  
 FCLIENT10\Administrator  
 Benchcraft FCLIENT10\Administrator:Benchcraft  
 FCLIENT10\Administrator  
 QLogic Corporation FCLIENT10\Administrator:QLogic Corporation  
 FCLIENT10\Administrator  
 QLogic Corporation\SANblade Control VIX  
 FCLIENT10\Administrator:QLogic Corporation\SANblade Control VIX  
 FCLIENT10\Administrator  
 Startup FCLIENT10\Administrator:Startup  
 FCLIENT10\Administrator

[Startup Programs]

Program	Command	User Name	Location
Shortcut to synctime.cmd	c:\batfiles\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	C:\WINNT\System32\mspaint.exe

[Internet Explorer]

[ Following are sub-categories of this main category ]

[Summary]

Item	Value
------	-------

Version 6.0.2800.1106  
 Build 62800.1106  
 Product ID 55736-276-6004927-04967  
 Application Path C:\Program Files\Internet Explorer  
 Language English (United States)  
 Active Printer Not Available

Cipher Strength 128-bit  
 Content Advisor Disabled  
 Update Versions ;SP1;  
 Java VM Version 5.0.3805.0  
 IEAK Install No

[File Versions]

File	Version	Size	Date	Path	Company
advapi32.dll	5.0.2195.5385	359 KB	7/22/2002 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
advpack.dll	6.0.2800.1106	89 KB	8/29/2002 8:14:40 AM	C:\WINNT\system32	Microsoft Corporation
browsecl.dll	6.0.2800.1106	62 KB	8/29/2002 8:14:40 AM	C:\WINNT\system32	Microsoft Corporation
browseui.dll	6.0.2800.1106	1002 KB	8/29/2002 8:14:40 AM	C:\WINNT\system32	Microsoft Corporation
ckcnv.exe	5.0.2189.1	9 KB	12/7/1999 8:00:00 AM	C:\WINNT\system32	Microsoft Corporation
comctl32.dll	5.81.4916.400	517 KB	8/29/2002 8:14:40 AM	C:\WINNT\system32	Microsoft Corporation
crypt32.dll	5.131.2195.4558	464 KB	7/22/2002 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
enhshg.dll	File not present	Not Available	Not Available	Not Available	Not Available
iemigrat.dll	File not present	Not Available	Not Available	Not Available	Not Available
iesetup.dll	6.0.2800.1106	57 KB	8/29/2002 8:14:40 AM	C:\WINNT\system32	Microsoft Corporation
iexplore.exe	6.0.2800.1106	89 KB	8/29/2002 8:14:40 AM	C:\Program Files\Internet Explorer	Microsoft Corporation
imagehlp.dll	5.0.2195.5242	126 KB	7/22/2002 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
inseng.dll	6.0.2800.1106	68 KB	8/29/2002 8:14:40 AM	C:\WINNT\system32	Microsoft Corporation
jobexec.dll	5.0.0.1	47 KB	12/7/1999 8:00:00 AM	C:\WINNT\system32	Microsoft Corporation
jscrip.dll	5.6.0.6626	576 KB	6/26/2001 5:36:02 PM	C:\WINNT\system32	Microsoft Corporation
jsproxy.dll	6.0.2800.1106	12 KB	8/29/2002 8:14:40 AM	C:\WINNT\system32	Microsoft Corporation
mshtml.dll	6.0.2800.1106	2722 KB	8/29/2002 8:14:40 AM	C:\WINNT\system32	Microsoft Corporation
msjava.dll	5.0.3805.0	924 KB	7/22/2002 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
msoss.dll	File not present	Not Available	Not Available	Not Available	Not Available
msxml.dll	8.0.6730.0	494 KB	7/22/2002 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
occache.dll	6.0.2800.1106	86 KB	8/29/2002 8:14:40 AM	C:\WINNT\system32	Microsoft Corporation
ole32.dll	5.0.2195.5400	968 KB	7/22/2002 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
oleaut32.dll	2.40.4518.0	612 KB	7/22/2002 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
olepro32.dll	5.0.4518.0	160 KB	7/22/2002 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
rsabase.dll	5.0.2195.3839	128 KB	7/22/2002 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
rsaenh.dll	5.0.2195.3839	131 KB	7/22/2002 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation

rasapi32.dll	5.0.2195.5438	192 KB	7/22/2002 1:05:04 PM
C:\WINNT\system32	Microsoft Corporation		
rsasig.dll	File not present	Not Available	Not Available
Not Available			
schannel.dll	5.1.2195.0	139 KB	7/22/2002 1:05:04 PM
C:\WINNT\system32	Microsoft Corporation		
shdoc401.dll	File not present	Not Available	Not
Available			
shdocvw.dll	6.0.2800.1106	1307 KB	8/29/2002 8:14:40 AM
C:\WINNT\system32	Microsoft Corporation		
shell32.dll	5.0.3502.5436	2319 KB	7/22/2002 1:05:04 PM
C:\WINNT\system32	Microsoft Corporation		
shlwapi.dll	6.0.2800.1106	386 KB	8/29/2002 8:14:40 AM
C:\WINNT\system32	Microsoft Corporation		
url.dll	6.0.2800.1106	104 KB	8/29/2002 8:14:40 AM
C:\WINNT\system32	Microsoft Corporation		
urlmon.dll	6.0.2800.1106	472 KB	8/29/2002 8:14:40 AM
C:\WINNT\system32	Microsoft Corporation		
vbscript.dll	5.6.0.7426	452 KB	2/26/2002 4:58:06 PM
C:\WINNT\system32	Microsoft Corporation		
webcheck.dll	6.0.2800.1106	252 KB	8/29/2002 8:14:40 AM
C:\WINNT\system32	Microsoft Corporation		
win.com	5.0.2134.1	24 KB	12/7/1999 8:00:00 AM
C:\WINNT\system32	Microsoft Corporation		
wininet.dll	6.0.2800.1106	572 KB	8/29/2002 8:14:40 AM
C:\WINNT\system32	Microsoft Corporation		
winsock.dll	3.10.0.103	3 KB	12/7/1999 8:00:00 AM
C:\WINNT\system32	Microsoft Corporation		
wintrust.dll	5.131.2195.3775	162 KB	7/22/2002 1:05:04 PM
C:\WINNT\system32	Microsoft Corporation		
wsock.vxd	File not present	Not Available	Not Available
Not Available			
wsock32.dll	5.0.2195.4874	21 KB	7/22/2002 1:05:04 PM
C:\WINNT\system32	Microsoft Corporation		
wsock32n.dll	File not present	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	Disabled
AutoConfigURL	
Proxy	Disabled
ProxyServer	
ProxyOverride	

[Cache]

[ Following are sub-categories of this main category ]

[Summary]

Item	Value
Page Refresh Type	Automatic
Temporary Internet Files Folder	C:\Documents and Settings\Administrator\Local Settings\Temporary Internet Files
Total Disk Space	17351 MB
Available Disk Space	12110 MB
Maximum Cache Size	541 MB
Available Cache Size	542 MB

[List of Objects]

Program File	Status	Version	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			2/12/2003 to 1/19/2103
sha1RSA				

[Other People Certificates]

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

[Publishers]

Name
No publisher information available

[Security]

Zone	Security Level
Local intranet	Custom
Trusted sites	Custom
Internet	Custom
Restricted sites	Custom

## Registry Parameters

### Client 1

SYTPTUNE.TXT

-----

COM+ Settings  
 TPCC.AllTxns:  
 Activation:  
 Enable Object Pooling - Selected  
 Minimum Pool Size - 75  
 Maximum Pool Size - 75  
 Creation Timeout (ms) - 500000  
 Enable object construction - Selected  
 Enable Just In Time Activation - Selected  
 Component supports events and statistics - Unselected

Registry Settings:  
 Windows Registry Editor Version 5.00

[HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo]

[HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo\Parameters]

"ListenBackLog"=dword:000000aa  
 "DispatchEntries"=hex(7):4c,00,44,00,41,00,50,00,53,00,56,00,43,00,00,00,00,00











"MinorVersion"=dword:00000000  
"InstallPath"="C:\\WINNT\\System32\\inetrv"  
"CertMapList"="C:\\WINNT\\System32\\inetrv\\iisrmap.dll"  
"AccessDeniedMessage"="Error: Access is Denied."  
"Filter DLLs"=""  
"LogFileDirectory"="C:\\WINNT\\System32\\LogFiles"  
"AcceptExOutstanding"=dword:00000050

[HKEY\_LOCAL\_MACHINE\\SYSTEM\\CurrentControlSet\\Services\\W3SVC\\Parameters\\ADCLaunch]

[HKEY\_LOCAL\_MACHINE\\SYSTEM\\CurrentControlSet\\Services\\W3SVC\\Parameters\\ADCLaunch\\AdvancedDataFactory]

[HKEY\_LOCAL\_MACHINE\\SYSTEM\\CurrentControlSet\\Services\\W3SVC\\Parameters\\ADCLaunch\\RDS\\Server.DataFactory]

[HKEY\_LOCAL\_MACHINE\\SYSTEM\\CurrentControlSet\\Services\\W3SVC\\Parameters\\Script Map]

[HKEY\_LOCAL\_MACHINE\\SYSTEM\\CurrentControlSet\\Services\\W3SVC\\Parameters\\Virtual Roots]

"/"="c:\\inetpub\\wwwroot,,205"  
"/Scripts"="c:\\inetpub\\scripts,,1"  
"/IISHelp"="c:\\winnt\\help\\iishelp,,1"  
"/IISAdmin"="C:\\WINNT\\System32\\inetrv\\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\\W3SVC\\Performance]

"Library"="w3ctrs.dll"  
"Open"="OpenW3PerformanceData"  
"Close"="CloseW3PerformanceData"  
"Collect"="CollectW3PerformanceData"  
"WbemAdapFileTime"=hex:00,5a,65,9a,60,5e,c2,01  
"WbemAdapFileSize"=dword:00001d10  
"WbemAdapStatus"=dword:00000000  
"Last Counter"=dword:00000b76  
"Last Help"=dword:00000b77  
"First Counter"=dword:00000ad4  
"First Help"=dword:00000ad5  
"Library Validation Code"=hex:00,5a,65,9a,60,5e,c2,01,10,1d,00,00,00,00,00,00

[HKEY\_LOCAL\_MACHINE\\SYSTEM\\CurrentControlSet\\Services\\W3SVC\\Security]

"Security"=hex:01,00,14,80,a0,00,00,00,ac,00,00,00,14,00,00,00,30,00,00,00,2,\\00,1c,00,01,00,00,00,02,80,14,00,ff,01,0f,00,01,01,00,00,00,00,00,01,00,00,\\00,00,02,00,70,00,04,00,00,00,00,18,00,fd,01,02,00,01,01,00,00,00,00,00,\\05,12,00,00,00,74,00,6f,00,00,00,1c,00,ff,01,0f,00,01,02,00,00,00,00,00,05,\\20,00,00,00,20,02,00,00,72,00,73,00,00,00,18,00,8d,01,02,00,01,01,00,00,00,\\00,00,05,0b,00,00,00,20,02,00,00,00,00,1c,00,fd,01,02,00,01,02,00,00,00,00,\\00,05,20,00,00,00,23,02,00,00,72,00,73,00,01,01,00,00,00,00,00,05,12,00,00,\\00,01,01,00,00,00,00,00,05,12,00,00,00

[HKEY\_LOCAL\_MACHINE\\SYSTEM\\CurrentControlSet\\Services\\W3SVC\\Enum]

"0"="Root\\LEGACY\_W3SVC\\0000"  
"Count"=dword:00000001  
"NextInstance"=dword:00000001

## Client 5

SYTPTUNE.TXT

-----  
COM+ Settings  
TPCC.AllTxns:  
Activation:  
Enable Object Pooling - Selected  
Minimum Pool Size - 75  
Maximum Pool Size - 75  
Creation Timeout (ms) - 500000  
Enable object construction - Selected  
Enable Just In Time Activation - Selected  
Component supports events and statistics - Unselected

Registry Settings:  
Windows Registry Editor Version 5.00

[HKEY\_LOCAL\_MACHINE\\SYSTEM\\CurrentControlSet\\Services\\InetInfo]

[HKEY\_LOCAL\_MACHINE\\SYSTEM\\CurrentControlSet\\Services\\InetInfo\\Parameters]

"ListenBackLog"=dword:000000aa  
"DispatchEntries"=hex(7):4c,00,44,00,41,00,50,00,53,00,56,00,43,00,00,00,00,00  
"PoolThreadLimit"=dword:00000400  
"ThreadTimeout"=dword:00015180

[HKEY\_LOCAL\_MACHINE\\SYSTEM\\CurrentControlSet\\Services\\InetInfo\\Performance]

"Library"="infectrs.dll"  
"Open"="OpenINFOPerformanceData"  
"Close"="CloseINFOPerformanceData"  
"Collect"="CollectINFOPerformanceData"  
"WbemAdapFileTime"=hex:00,a0,13,ec,a1,31,c2,01  
"WbemAdapFileSize"=dword:00002510  
"WbemAdapStatus"=dword:00000000  
"Last Counter"=dword:00000bfe  
"Last Help"=dword:00000bff  
"First Counter"=dword:00000bbe  
"First Help"=dword:00000bbf  
"Library Validation Code"=hex:7c,c8,2a,18,5c,df,c2,01,10,25,00,00,00,00,00,00

Windows Registry Editor Version 5.00

[HKEY\_LOCAL\_MACHINE\\SOFTWARE\\Microsoft\\TPCC]

"Path"="C:\\Inetpub\\wwwroot\\"  
"NumberOfDeliveryThreads"=dword:0000000b  
"MaxConnections"=dword:00007530  
"MaxPendingDeliveries"=dword:00000bb8  
"DB\_Protocol"="ODBC"  
"TxnMonitor"="COM"  
"DbServer"="ibmserv4"  
"DbName"="tpcc"  
"DbUser"="sa"  
"DbPassword"=""  
"COM\_SinglePool"="YES"

Windows Registry Editor Version 5.00

[HKEY\_LOCAL\_MACHINE\\SYSTEM\\CurrentControlSet\\Services\\W3SVC]

"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,00,65,00,74,00,73,\
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,00,49,00,4e,00,0
0,00,\
00,00
"DependOnGroup"=hex(7):00,00
"ObjectName"="LocalSystem"
"Description"="Provides Web connectivity and administration through the
Internet Information Services snap-in."
"FailureActions"=hex:ff,ff,ff,ff,00,00,00,00,00,00,00,03,00,00,00,b8,30,10,\
00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\
ASP]
"NOTE"="This is for backward compatibility only."

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\
ASP\Parameters]

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\P
arameters]
"MajorVersion"=dword:00000005
"MinorVersion"=dword:00000000
"InstallPath"="C:\WINNT\System32\inetrv"
"CertMapList"="C:\WINNT\System32\inetrv\iiscmap.dll"
"AccessDeniedMessage"="Error: Access is Denied."
"Filter DLLs"=""
"LogFileDirectory"="C:\WINNT\System32\LogFiles"
"AcceptExOutstanding"=dword:00000050

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\P
arameters\ADCLaunch]

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\P
arameters\ADCLaunch\AdvancedDataFactory]

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\P
arameters\ADCLaunch\RDSServer.DataFactory]

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\P
arameters\Script Map]

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\P
arameters\Virtual Roots]
"/"="c:\inetpub\wwwroot,,205"
"/Scripts"="c:\inetpub\scripts,,1"
"/IISHelp"="c:\winnt\help\iishelp,,1"
"/IISAdmin"="C:\WINNT\System32\inetrv\iisadmin,,1"
"/IISSamples"="c:\inetpub\iissamples,,1"
"/MSADC"="c:\program files\common files\system\msadc,,1"
"/_yti_bin"="C:\Program Files\Common Files\Microsoft Shared\Web Server
Extensions\40\isapi,,1"
"/Rpc"="C:\WINNT\System32\RpcProxy,,1"
"/Printers"="C:\WINNT\web\printers,,201"

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\P
erformance]
"Library"="w3ctrs.dll"
"Open"="OpenW3PerformanceData"
"Close"="CloseW3PerformanceData"
"Collect"="CollectW3PerformanceData"
"WbemAdapFileTime"=hex:00,5a,65,9a,60,5e,c2,01
"WbemAdapFileSize"=dword:00001d10
"WbemAdapStatus"=dword:00000000
"Last Counter"=dword:00000b76
"Last Help"=dword:00000b77

```

```

"First Counter"=dword:00000ad4
"First Help"=dword:00000ad5
"Library Validation
Code"=hex:00,5a,65,9a,60,5e,c2,01,10,1d,00,00,00,00,00
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\
Security]
"Security"=hex:01,00,14,80,a0,00,00,00,ac,00,00,00,14,00,00,00,30,00,00,0
2,\
00,1c,00,01,00,00,00,02,80,14,00,ff,01,0f,00,01,01,00,00,00,00,01,00,00,\
00,00,02,00,70,00,04,00,00,00,00,18,00,fd,01,02,00,01,01,00,00,00,00,00,\
05,12,00,00,00,74,00,6f,00,00,00,1c,00,ff,01,0f,00,01,02,00,00,00,00,05,\
20,00,00,00,20,02,00,00,72,00,73,00,00,18,00,8d,01,02,00,01,01,00,00,00,\
00,00,05,0b,00,00,20,02,00,00,00,1c,00,fd,01,02,00,01,02,00,00,00,\
00,05,20,00,00,23,02,00,00,72,00,73,00,01,01,00,00,00,00,05,12,00,00,\
00,01,01,00,00,00,00,05,12,00,00,00
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\
enum]
"0"="Root\LEGACY_W3SVC\0000"
"Count"=dword:00000001
"NextInstance"=dword:00000001

Client 6
SYTPTUNE.TXT
-----
COM+ Settings
TPCC.AllTxns:
Activation:
Enable Object Pooling - Selected
Minimum Pool Size - 75
Maximum Pool Size - 75
Creation Timeout (ms) - 500000
Enable object construction - Selected
Enable Just In Time Activation - Selected
Component supports events and statistics - Unselected

Registry Settings:
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo]

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo\P
arameters]
"ListenBackLog"=dword:000000aa
"DispatchEntries"=hex(7):4c,00,44,00,41,00,50,00,53,00,56,00,43,00,00,00,
00
"PoolThreadLimit"=dword:00000400
"ThreadTimeout"=dword:00015180

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo\P
erformance]
"Library"="infectrs.dll"
"Open"="OpenINFOPerformanceData"
"Close"="CloseINFOPerformanceData"
"Collect"="CollectINFOPerformanceData"
"WbemAdapFileTime"=hex:00,a0,13,ec,a1,31,c2,01
"WbemAdapFileSize"=dword:00002510
"WbemAdapStatus"=dword:00000000
"Last Counter"=dword:00000bfe
"Last Help"=dword:00000bff
"First Counter"=dword:00000bbe
"First Help"=dword:00000bbf
"Library Validation
Code"=hex:7c,c8,2a,18,5c,df,c2,01,10,25,00,00,00,00,00,00

```







"WbemAdapStatus"=dword:00000000  
"Last Counter"=dword:00000b76  
"Last Help"=dword:00000b77  
"First Counter"=dword:00000ad4  
"First Help"=dword:00000ad5  
"Library Validation  
Code"=hex:00,5a,65,9a,60,5e,c2,01,10,1d,00,00,00,00,00,00

[HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Security]  
"Security"=hex:01,00,14,80,a0,00,00,00,ac,00,00,00,14,00,00,00,30,00,00,00,02,\  
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,01,02,00,01,01,00,00,00,00,\  
05,12,00,00,00,74,00,6f,00,00,00,1c,00,ff,01,0f,00,01,02,00,00,00,00,05,\  
20,00,00,00,20,02,00,00,72,00,73,00,00,00,18,00,8d,01,02,00,01,01,00,00,\  
00,00,05,0b,00,00,00,20,02,00,00,00,1c,00,fd,01,02,00,01,02,00,00,00,\  
00,05,20,00,00,00,23,02,00,00,72,00,73,00,01,01,00,00,00,00,05,12,00,00,\  
00,01,01,00,00,00,00,05,12,00,00,00

[HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Enum]  
"0"="Root\LEGACY\_W3SVC\0000"  
"Count"=dword:00000001  
"NextInstance"=dword:00000001

## RTE Input Parameters

Profile: 15232wh\_8cl\_8rte\_lessN02moreN13\_no3f  
File Path: C:\Program  
Files\BenchCraft\15232wh\_8cl\_8rte\_lessN02moreN13\_no3f.pro  
Version: 4

Number of Engines: 63

Name: frte10a  
Description: frte10a  
Directory: c:\rtelogs\frte10a.log  
Machine: frte10  
Parameter Set: PARAM2  
Index: 0  
Seed: 4678  
Configured Users: 2460  
Pipe Name: DRIVER14723953  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 0  
Additional Options:

Name: frte10b  
Description: frte10b  
Directory: c:\rtelogs\frte10b.log  
Machine: frte10  
Parameter Set: PARAM2  
Index: 32812500  
Seed: 4678  
Configured Users: 2460  
Pipe Name: DRIVER24872015  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 1

Additional Options:

Name: frte10c  
Description: frte10c  
Directory: c:\rtelogs\frte10c.log  
Machine: frte10  
Parameter Set: PARAM2  
Index: 65625000  
Seed: 4678  
Configured Users: 2460  
Pipe Name: DRIVER34966171  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 0  
Additional Options:

Name: frte10d  
Description: frte10d  
Directory: c:\rtelogs\frte10d.log  
Machine: frte10  
Parameter Set: PARAM2  
Index: 98437500  
Seed: 4678  
Configured Users: 2460  
Pipe Name: DRIVER45012500  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 1  
Additional Options:

Name: frte10e  
Description: frte10e  
Directory: c:\rtelogs\frte10e.log  
Machine: frte10  
Parameter Set: PARAM2  
Index: 131250000  
Seed: 4678  
Configured Users: 2450  
Pipe Name: DRIVER55066765  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 0  
Additional Options:

Name: frte10f  
Description: frte10f  
Directory: c:\rtelogs\frte10f.log  
Machine: frte10  
Parameter Set: PARAM2  
Index: 164062500  
Seed: 4678  
Configured Users: 2450  
Pipe Name: DRIVER65126187  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 1



Additional Options:

Name: frte10g  
Description: frte10g  
Directory: c:\rtelogs\frte10g.log  
Machine: frte10  
Parameter Set: PARAM2  
Index: 196875000  
Seed: 4678  
Configured Users: 2450  
Pipe Name: DRIVER75175640  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 0  
Additional Options:

Name: frte10h  
Description: frte10h  
Directory: c:\rtelogs\frte10h.log  
Machine: frte10  
Parameter Set: PARAM2  
Index: 229687500  
Seed: 4678  
Configured Users: 2450  
Pipe Name: DRIVER85211109  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 1  
Additional Options:

Name: frte20a  
Description: frte20a  
Directory: c:\rtelogs\frte20a.log  
Machine: frte20  
Parameter Set: PARAM2  
Index: 262500000  
Seed: 4678  
Configured Users: 2430  
Pipe Name: DRIVER155552765  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 0  
Additional Options:

Name: frte20b  
Description: frte20b  
Directory: c:\rtelogs\frte20b.log  
Machine: frte20  
Parameter Set: PARAM2  
Index: 295312500  
Seed: 4678  
Configured Users: 2430  
Pipe Name: DRIVER165598015  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 1

Additional Options:

Name: frte20c  
Description: frte20c  
Directory: c:\rtelogs\frte20c.log  
Machine: frte20  
Parameter Set: PARAM2  
Index: 328125000  
Seed: 4678  
Configured Users: 2430  
Pipe Name: DRIVER175634359  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 0  
Additional Options:

Name: frte20d  
Description: frte20d  
Directory: c:\rtelogs\frte20d.log  
Machine: frte20  
Parameter Set: PARAM2  
Index: 360937500  
Seed: 4678  
Configured Users: 2430  
Pipe Name: DRIVER185679015  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 1  
Additional Options:

Name: frte20e  
Description: frte20e  
Directory: c:\rtelogs\frte20e.log  
Machine: frte20  
Parameter Set: PARAM2  
Index: 393750000  
Seed: 4678  
Configured Users: 2430  
Pipe Name: DRIVER195713546  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 0  
Additional Options:

Name: frte20f  
Description: frte20f  
Directory: c:\rtelogs\frte20f.log  
Machine: frte20  
Parameter Set: PARAM2  
Index: 426562500  
Seed: 4678  
Configured Users: 2430  
Pipe Name: DRIVER205755140  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 1

Additional Options:

Name: frte20g  
Description: frte20g  
Directory: c:\rtelogs\frte20g.log  
Machine: frte20  
Parameter Set: PARAM2  
Index: 459375000  
Seed: 4678  
Configured Users: 2430  
Pipe Name: DRIVER215788046  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 0  
Additional Options:

Name: frte20h  
Description: frte20h  
Directory: c:\rtelogs\frte20h.log  
Machine: frte20  
Parameter Set: PARAM2  
Index: 492187500  
Seed: 4678  
Configured Users: 2430  
Pipe Name: DRIVER225826796  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 1  
Additional Options:

Name: frte30a  
Description: frte30a  
Directory: c:\rtelogs\frte30a.log  
Machine: frte30  
Parameter Set: PARAM2  
Index: 525000000  
Seed: 4678  
Configured Users: 2430  
Pipe Name: DRIVER2966335687  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 0  
Additional Options:

Name: frte30b  
Description: frte30b  
Directory: c:\rtelogs\frte30b.log  
Machine: frte30  
Parameter Set: PARAM2  
Index: 557812500  
Seed: 4678  
Configured Users: 2430  
Pipe Name: DRIVER3066392812  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 1

Additional Options:

Name: frte30c  
Description: frte30c  
Directory: c:\rtelogs\frte30c.log  
Machine: frte30  
Parameter Set: PARAM2  
Index: 590625000  
Seed: 4678  
Configured Users: 2430  
Pipe Name: DRIVER3166432171  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 0  
Additional Options:

Name: frte30d  
Description: frte30d  
Directory: c:\rtelogs\frte30d.log  
Machine: frte30  
Parameter Set: PARAM2  
Index: 623437500  
Seed: 4678  
Configured Users: 2430  
Pipe Name: DRIVER3266504046  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 1  
Additional Options:

Name: frte30e  
Description: frte30e  
Directory: c:\rtelogs\frte30e.log  
Machine: frte30  
Parameter Set: PARAM2  
Index: 656250000  
Seed: 4678  
Configured Users: 2440  
Pipe Name: DRIVER3366550687  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 0  
Additional Options:

Name: frte30g  
Description: frte30g  
Directory: c:\rtelogs\frte30g.log  
Machine: frte30  
Parameter Set: PARAM2  
Index: 721875000  
Seed: 4678  
Configured Users: 2440  
Pipe Name: DRIVER3566700406  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 0

Additional Options:

Name: frte30h  
Description: frte30h  
Directory: c:\rtelogs\frte30h.log  
Machine: frte30  
Parameter Set: PARAM2  
Index: 754687500  
Seed: 4678  
Configured Users: 2440  
Pipe Name: DRIVER3666766140  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 1  
Additional Options:

Name: frte40a  
Description: frte40a  
Directory: c:\rtelogs\frte40a.log  
Machine: frte40  
Parameter Set: PARAM2  
Index: 787500000  
Seed: 4678  
Configured Users: 2460  
Pipe Name: DRIVER4367196046  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 0  
Additional Options:

Name: frte40b  
Description: frte40b  
Directory: c:\rtelogs\frte40b.log  
Machine: frte40  
Parameter Set: PARAM2  
Index: 820312500  
Seed: 4678  
Configured Users: 2460  
Pipe Name: DRIVER4467244484  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 1  
Additional Options:

Name: frte40c  
Description: frte40c  
Directory: c:\rtelogs\frte40c.log  
Machine: frte40  
Parameter Set: PARAM2  
Index: 853125000  
Seed: 4678  
Configured Users: 2460  
Pipe Name: DRIVER4567282265  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 0

Additional Options:

Name: frte40d  
Description: frte40d  
Directory: c:\rtelogs\frte40d.log  
Machine: frte40  
Parameter Set: PARAM2  
Index: 885937500  
Seed: 4678  
Configured Users: 2460  
Pipe Name: DRIVER4667357593  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 1  
Additional Options:

Name: frte40e  
Description: frte40e  
Directory: c:\rtelogs\frte40e.log  
Machine: frte40  
Parameter Set: PARAM2  
Index: 918750000  
Seed: 4678  
Configured Users: 2450  
Pipe Name: DRIVER4767405796  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 0  
Additional Options:

Name: frte40f  
Description: frte40f  
Directory: c:\rtelogs\frte40f.log  
Machine: frte40  
Parameter Set: PARAM2  
Index: 951562500  
Seed: 4678  
Configured Users: 2450  
Pipe Name: DRIVER4867450406  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 1  
Additional Options:

Name: frte40g  
Description: frte40g  
Directory: c:\rtelogs\frte40g.log  
Machine: frte40  
Parameter Set: PARAM2  
Index: 984375000  
Seed: 4678  
Configured Users: 2450  
Pipe Name: DRIVER4967505484  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 0

Additional Options:

Name: frte40h  
Description: frte40h  
Directory: c:\rtelogs\frte40h.log  
Machine: frte40  
Parameter Set: PARAM2  
Index: 1017187500  
Seed: 4678  
Configured Users: 2450  
Pipe Name: DRIVER5067590062  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 1  
Additional Options:

Name: frte50a  
Description: frte50a  
Directory: c:\rtelogs\frte50a.log  
Machine: frte50  
Parameter Set: PARAM2  
Index: 1050000000  
Seed: 4678  
Configured Users: 2460  
Pipe Name: DRIVER5768066062  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 0  
Additional Options:

Name: frte50b  
Description: frte50b  
Directory: c:\rtelogs\frte50b.log  
Machine: frte50  
Parameter Set: PARAM2  
Index: 1082812500  
Seed: 4678  
Configured Users: 2460  
Pipe Name: DRIVER5868120031  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 1  
Additional Options:

Name: frte50c  
Description: frte50c  
Directory: c:\rtelogs\frte50c.log  
Machine: frte50  
Parameter Set: PARAM2  
Index: 1115625000  
Seed: 4678  
Configured Users: 2460  
Pipe Name: DRIVER5968167296  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 0

Additional Options:

Name: frte50d  
Description: frte50d  
Directory: c:\rtelogs\frte50d.log  
Machine: frte50  
Parameter Set: PARAM2  
Index: 1148437500  
Seed: 4678  
Configured Users: 2460  
Pipe Name: DRIVER6068289875  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 1  
Additional Options:

Name: frte50e  
Description: frte50e  
Directory: c:\rtelogs\frte50e.log  
Machine: frte50  
Parameter Set: PARAM2  
Index: 1181250000  
Seed: 4678  
Configured Users: 2450  
Pipe Name: DRIVER6168341984  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 0  
Additional Options:

Name: frte50f  
Description: frte50f  
Directory: c:\rtelogs\frte50f.log  
Machine: frte50  
Parameter Set: PARAM2  
Index: 1214062500  
Seed: 4678  
Configured Users: 2450  
Pipe Name: DRIVER6268394203  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 1  
Additional Options:

Name: frte50g  
Description: frte50g  
Directory: c:\rtelogs\frte50g.log  
Machine: frte50  
Parameter Set: PARAM2  
Index: 1246875000  
Seed: 4678  
Configured Users: 2450  
Pipe Name: DRIVER6369221937  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 0

Additional Options:

Name: frte50h  
Description: frte50h  
Directory: c:\rtelogs\frte50h.log  
Machine: frte50  
Parameter Set: PARAM2  
Index: 1279687500  
Seed: 4678  
Configured Users: 2450  
Pipe Name: DRIVER6469272609  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 1  
Additional Options:

Name: frte60a  
Description: frte60a  
Directory: c:\rtelogs\frte60a.log  
Machine: frte60  
Parameter Set: PARAM2  
Index: 1312500000  
Seed: 4678  
Configured Users: 2330  
Pipe Name: DRIVER7159034000  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 0  
Additional Options:

Name: frte60b  
Description: frte60b  
Directory: c:\rtelogs\frte60b.log  
Machine: frte60  
Parameter Set: PARAM2  
Index: 1345312500  
Seed: 4678  
Configured Users: 2330  
Pipe Name: DRIVER7259107640  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 1  
Additional Options:

Name: frte60c  
Description: frte60c  
Directory: c:\rtelogs\frte60c.log  
Machine: frte60  
Parameter Set: PARAM2  
Index: 1378125000  
Seed: 4678  
Configured Users: 2330  
Pipe Name: DRIVER7359143578  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 0

Additional Options:

Name: frte60d  
Description: frte60d  
Directory: c:\rtelogs\frte60d.log  
Machine: frte60  
Parameter Set: PARAM2  
Index: 1410937500  
Seed: 4678  
Configured Users: 2330  
Pipe Name: DRIVER7459167015  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 1  
Additional Options:

Name: frte60e  
Description: frte60e  
Directory: c:\rtelogs\frte60e.log  
Machine: frte60  
Parameter Set: PARAM2  
Index: 1443750000  
Seed: 4678  
Configured Users: 2330  
Pipe Name: DRIVER7559188718  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 0  
Additional Options:

Name: frte60f  
Description: frte60f  
Directory: c:\rtelogs\frte60f.log  
Machine: frte60  
Parameter Set: PARAM2  
Index: 1476562500  
Seed: 4678  
Configured Users: 2330  
Pipe Name: DRIVER7659224156  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 1  
Additional Options:

Name: frte60g  
Description: frte60g  
Directory: c:\rtelogs\frte60g.log  
Machine: frte60  
Parameter Set: PARAM2  
Index: 1509375000  
Seed: 4678  
Configured Users: 2330  
Pipe Name: DRIVER7759246812  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 0

Additional Options:

Name: frte60h  
Description: frte60h  
Directory: c:\frtelogs\frte60h.log  
Machine: frte60  
Parameter Set: PARAM2  
Index: 1542187500  
Seed: 4678  
Configured Users: 2330  
Pipe Name: DRIVER7859276281  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 1  
Additional Options:

Name: frte70a  
Description: frte70a  
Directory: c:\frtelogs\frte70a.log  
Machine: frte70  
Parameter Set: PARAM2  
Index: 1575000000  
Seed: 4678  
Configured Users: 2330  
Pipe Name: DRIVER853952421  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 0  
Additional Options:

Name: frte70b  
Description: frte70b  
Directory: c:\frtelogs\frte70b.log  
Machine: frte70  
Parameter Set: PARAM2  
Index: 1607812500  
Seed: 4678  
Configured Users: 2330  
Pipe Name: DRIVER8617658171  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 1  
Additional Options:

Name: frte70c  
Description: frte70c  
Directory: c:\frtelogs\frte70c.log  
Machine: frte70  
Parameter Set: PARAM2  
Index: 1640625000  
Seed: 4678  
Configured Users: 2330  
Pipe Name: DRIVER8717687234  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 0

Additional Options:

Name: frte70d  
Description: frte70d  
Directory: c:\frtelogs\frte70d.log  
Machine: frte70  
Parameter Set: PARAM2  
Index: 1673437500  
Seed: 4678  
Configured Users: 2330  
Pipe Name: DRIVER8817717718  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 1  
Additional Options:

Name: frte70e  
Description: frte70e  
Directory: c:\frtelogs\frte70e.log  
Machine: frte70  
Parameter Set: PARAM2  
Index: 1706250000  
Seed: 4678  
Configured Users: 2340  
Pipe Name: DRIVER8917754812  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 0  
Additional Options:

Name: frte70f  
Description: frte70f  
Directory: c:\frtelogs\frte70f.log  
Machine: frte70  
Parameter Set: PARAM2  
Index: 1739062500  
Seed: 4678  
Configured Users: 2340  
Pipe Name: DRIVER9017811078  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 1  
Additional Options:

Name: frte70g  
Description: frte70g  
Directory: c:\frtelogs\frte70g.log  
Machine: frte70  
Parameter Set: PARAM2  
Index: 1771875000  
Seed: 4678  
Configured Users: 2340  
Pipe Name: DRIVER9117828906  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 0

Additional Options:

Name: frte70h  
Description: frte70h  
Directory: c:\frtelogs\рте70h.log  
Machine: frte70  
Parameter Set: PARAM2  
Index: 1804687500  
Seed: 4678  
Configured Users: 2340  
Pipe Name: DRIVER9218168437  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 1  
Additional Options:

Name: frte80a  
Description: frte80a  
Directory: c:\frtelogs\рте80a.log  
Machine: frte80  
Parameter Set: PARAM2  
Index: 1837500000  
Seed: 4678  
Configured Users: 2450  
Pipe Name: DRIVER9919498796  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 0  
Additional Options:

Name: frte80b  
Description: frte80b  
Directory: c:\frtelogs\рте80b.log  
Machine: frte80  
Parameter Set: PARAM2  
Index: 1870312500  
Seed: 4678  
Configured Users: 2450  
Pipe Name: DRIVER10019683375  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 1  
Additional Options:

Name: frte80c  
Description: frte80c  
Directory: c:\frtelogs\рте80c.log  
Machine: frte80  
Parameter Set: PARAM2  
Index: 1903125000  
Seed: 4678  
Configured Users: 2450  
Pipe Name: DRIVER10119700734  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 0

Additional Options:

Name: frte80d  
Description: frte80d  
Directory: c:\frtelogs\рте80d.log  
Machine: frte80  
Parameter Set: PARAM2  
Index: 1935937500  
Seed: 4678  
Configured Users: 2450  
Pipe Name: DRIVER10219721250  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 1  
Additional Options:

Name: frte80e  
Description: frte80e  
Directory: c:\frtelogs\рте80e.log  
Machine: frte80  
Parameter Set: PARAM2  
Index: 1968750000  
Seed: 4678  
Configured Users: 2450  
Pipe Name: DRIVER10319736046  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 0  
Additional Options:

Name: frte80f  
Description: frte80f  
Directory: c:\frtelogs\рте80f.log  
Machine: frte80  
Parameter Set: PARAM2  
Index: 2001562500  
Seed: 4678  
Configured Users: 2450  
Pipe Name: DRIVER10419752234  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 1  
Additional Options:

Name: frte80g  
Description: frte80g  
Directory: c:\frtelogs\рте80g.log  
Machine: frte80  
Parameter Set: PARAM2  
Index: 2034375000  
Seed: 4678  
Configured Users: 2450  
Pipe Name: DRIVER10519766734  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 0

Additional Options:

Name: frte80h  
Description: frte80h  
Directory: c:\frtelogs\frte80h.log  
Machine: frte80  
Parameter Set: PARAM2  
Index: 2067187500  
Seed: 4678  
Configured Users: 2450  
Pipe Name: DRIVER10619779687  
Connect Rate: 1500  
Start Rate: 0  
Max. Concurrency: 2460  
Concurrency Rate: 1500  
CLIENT\_NURAND: 233  
CPU: 1  
Additional Options:

Number of User groups: 63

Driver Engine: frte10a  
IIS Server: fclient10a  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 1 - 246  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2460  
District id: 1  
Scale Down: No

Driver Engine: frte10b  
IIS Server: fclient10b  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 247 - 492  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2460  
District id: 1  
Scale Down: No

Driver Engine: frte10c  
IIS Server: fclient10c  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 493 - 738  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2460  
District id: 1  
Scale Down: No

Driver Engine: frte10d  
IIS Server: fclient10d  
SQL Server: ibmserv4  
Database: tpcc  
User: sa

Protocol: HTML  
w\_id Range: 739 - 984  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2460  
District id: 1  
Scale Down: No

Driver Engine: frte10e  
IIS Server: fclient10e  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 985 - 1229  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2450  
District id: 1  
Scale Down: No

Driver Engine: frte10f  
IIS Server: fclient10f  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 1230 - 1474  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2450  
District id: 1  
Scale Down: No

Driver Engine: frte10g  
IIS Server: fclient10g  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 1475 - 1719  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2450  
District id: 1  
Scale Down: No

Driver Engine: frte10h  
IIS Server: fclient10h  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 1720 - 1964  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2450  
District id: 1  
Scale Down: No

Driver Engine: frte20a  
IIS Server: fclient20a  
SQL Server: ibmserv4



Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 1965 - 2207  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2430  
District id: 1  
Scale Down: No

Driver Engine: frte20b  
IIS Server: fclient20b  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 2208 - 2450  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2430  
District id: 1  
Scale Down: No

Driver Engine: frte20c  
IIS Server: fclient20c  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 2451 - 2693  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2430  
District id: 1  
Scale Down: No

Driver Engine: frte20d  
IIS Server: fclient20d  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 2694 - 2936  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2430  
District id: 1  
Scale Down: No

Driver Engine: frte20e  
IIS Server: fclient20e  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 2937 - 3179  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2430  
District id: 1  
Scale Down: No

Driver Engine: frte20f

IIS Server: fclient20f  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 3180 - 3422  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2430  
District id: 1  
Scale Down: No

Driver Engine: frte20g  
IIS Server: fclient20g  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 3423 - 3665  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2430  
District id: 1  
Scale Down: No

Driver Engine: frte20h  
IIS Server: fclient20h  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 3666 - 3908  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2430  
District id: 1  
Scale Down: No

Driver Engine: frte30a  
IIS Server: fclient30a  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 3909 - 4151  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2430  
District id: 1  
Scale Down: No

Driver Engine: frte30b  
IIS Server: fclient30b  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 4152 - 4394  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2430  
District id: 1  
Scale Down: No

Driver Engine: frte30c  
IIS Server: fclient30c  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 4395 - 4637  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2430  
District id: 1  
Scale Down: No

Driver Engine: frte30d  
IIS Server: fclient30d  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 4638 - 4880  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2430  
District id: 1  
Scale Down: No

Driver Engine: frte30e  
IIS Server: fclient30e  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 4881 - 5124  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2440  
District id: 1  
Scale Down: No

Driver Engine: frte30g  
IIS Server: fclient30g  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 5125 - 5368  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2440  
District id: 1  
Scale Down: No

Driver Engine: frte30h  
IIS Server: fclient30h  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 5369 - 5612  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2440

District id: 1  
Scale Down: No

Driver Engine: frte40a  
IIS Server: fclient40a  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 5613 - 5858  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2460  
District id: 1  
Scale Down: No

Driver Engine: frte40b  
IIS Server: fclient40b  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 5859 - 6104  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2460  
District id: 1  
Scale Down: No

Driver Engine: frte40c  
IIS Server: fclient40c  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 6105 - 6350  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2460  
District id: 1  
Scale Down: No

Driver Engine: frte40d  
IIS Server: fclient40d  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 6351 - 6596  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2460  
District id: 1  
Scale Down: No

Driver Engine: frte40e  
IIS Server: fclient40e  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 6597 - 6841  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232

Scale: Normal  
User Count: 2450  
District id: 1  
Scale Down: No  
  
Driver Engine: frt40f  
IIS Server: fclient40f  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 6842 - 7086  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2450  
District id: 1  
Scale Down: No

Driver Engine: frt40g  
IIS Server: fclient40g  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 7087 - 7331  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2450  
District id: 1  
Scale Down: No

Driver Engine: frt40h  
IIS Server: fclient40h  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 7332 - 7576  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2450  
District id: 1  
Scale Down: No

Driver Engine: frt50a  
IIS Server: fclient50a  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 7577 - 7822  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2460  
District id: 1  
Scale Down: No

Driver Engine: frt50b  
IIS Server: fclient50b  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 7823 - 8068

w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2460  
District id: 1  
Scale Down: No

Driver Engine: frt50c  
IIS Server: fclient50c  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 8069 - 8314  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2460  
District id: 1  
Scale Down: No

Driver Engine: frt50d  
IIS Server: fclient50d  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 8315 - 8560  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2460  
District id: 1  
Scale Down: No

Driver Engine: frt50e  
IIS Server: fclient50e  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 8561 - 8805  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2450  
District id: 1  
Scale Down: No

Driver Engine: frt50f  
IIS Server: fclient50f  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 8806 - 9050  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2450  
District id: 1  
Scale Down: No

Driver Engine: frt50g  
IIS Server: fclient50g  
SQL Server: ibmserv4  
Database: tpcc  
User: sa

Protocol: HTML  
w\_id Range: 9051 - 9295  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2450  
District id: 1  
Scale Down: No

Driver Engine: frte50h  
IIS Server: fclient50h  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 9296 - 9540  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2450  
District id: 1  
Scale Down: No

Driver Engine: frte60a  
IIS Server: fclient60a  
SQL Server: ibmserv4  
Database: tpcc  
User: us  
Protocol: HTML  
w\_id Range: 9541 - 9773  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2330  
District id: 1  
Scale Down: No

Driver Engine: frte60b  
IIS Server: fclient60b  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 9774 - 10006  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2330  
District id: 1  
Scale Down: No

Driver Engine: frte60c  
IIS Server: fclient60c  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 10007 - 10239  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2330  
District id: 1  
Scale Down: No

Driver Engine: frte60d  
IIS Server: fclient60d  
SQL Server: ibmserv4

Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 10240 - 10472  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2330  
District id: 1  
Scale Down: No

Driver Engine: frte60e  
IIS Server: fclient60e  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 10473 - 10705  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2330  
District id: 1  
Scale Down: No

Driver Engine: frte60f  
IIS Server: fclient60f  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 10706 - 10938  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2330  
District id: 1  
Scale Down: No

Driver Engine: frte60g  
IIS Server: fclient60g  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 10939 - 11171  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2330  
District id: 1  
Scale Down: No

Driver Engine: frte60h  
IIS Server: fclient60h  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 11172 - 11404  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2330  
District id: 1  
Scale Down: No

Driver Engine: frte70a

IIS Server: fclient70a  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 11405 - 11637  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2330  
District id: 1  
Scale Down: No

Driver Engine: frte70b  
IIS Server: fclient70b  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 11638 - 11870  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2330  
District id: 1  
Scale Down: No

Driver Engine: frte70c  
IIS Server: fclient70c  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 11871 - 12103  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2330  
District id: 1  
Scale Down: No

Driver Engine: frte70d  
IIS Server: fclient70d  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 12104 - 12336  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2330  
District id: 1  
Scale Down: No

Driver Engine: frte70e  
IIS Server: fclient70e  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 12337 - 12570  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2340  
District id: 1  
Scale Down: No

Driver Engine: frte70f  
IIS Server: fclient70f  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 12571 - 12804  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2340  
District id: 1  
Scale Down: No

Driver Engine: frte70g  
IIS Server: fclient70g  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 12805 - 13038  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2340  
District id: 1  
Scale Down: No

Driver Engine: frte70h  
IIS Server: fclient70h  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 13039 - 13272  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2340  
District id: 1  
Scale Down: No

Driver Engine: frte80a  
IIS Server: fclient80a  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 13273 - 13517  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2450  
District id: 1  
Scale Down: No

Driver Engine: frte80b  
IIS Server: fclient80b  
SQL Server: ibmserv4  
Database: tpcc  
User: sa  
Protocol: HTML  
w\_id Range: 13518 - 13762  
w\_id Min Warehouse: 1  
w\_id Max Warehouse: 15232  
Scale: Normal  
User Count: 2450

District id: 1  
 Scale Down: No

Driver Engine: frte80c  
 IIS Server: fclient80c  
 SQL Server: ibmserv4  
 Database: tpcc  
 User: sa  
 Protocol: HTML  
 w\_id Range: 13763 - 14007  
 w\_id Min Warehouse: 1  
 w\_id Max Warehouse: 15232  
 Scale: Normal  
 User Count: 2450  
 District id: 1  
 Scale Down: No

Driver Engine: frte80d  
 IIS Server: fclient80d  
 SQL Server: ibmserv4  
 Database: tpcc  
 User: sa  
 Protocol: HTML  
 w\_id Range: 14008 - 14252  
 w\_id Min Warehouse: 1  
 w\_id Max Warehouse: 15232  
 Scale: Normal  
 User Count: 2450  
 District id: 1  
 Scale Down: No

Driver Engine: frte80e  
 IIS Server: fclient80e  
 SQL Server: ibmserv4  
 Database: tpcc  
 User: sa  
 Protocol: HTML  
 w\_id Range: 14253 - 14497  
 w\_id Min Warehouse: 1  
 w\_id Max Warehouse: 15232  
 Scale: Normal  
 User Count: 2450  
 District id: 1  
 Scale Down: No

Driver Engine: frte80f  
 IIS Server: fclient80f  
 SQL Server: ibmserv4  
 Database: tpcc  
 User: sa  
 Protocol: HTML  
 w\_id Range: 14498 - 14742  
 w\_id Min Warehouse: 1  
 w\_id Max Warehouse: 15232  
 Scale: Normal  
 User Count: 2450  
 District id: 1  
 Scale Down: No

Driver Engine: frte80g  
 IIS Server: fclient80g  
 SQL Server: ibmserv4  
 Database: tpcc  
 User: sa  
 Protocol: HTML  
 w\_id Range: 14743 - 14987  
 w\_id Min Warehouse: 1  
 w\_id Max Warehouse: 15232

Scale: Normal  
 User Count: 2450  
 District id: 1  
 Scale Down: No

Driver Engine: frte80h  
 IIS Server: fclient80h  
 SQL Server: ibmserv4  
 Database: tpcc  
 User: sa  
 Protocol: HTML  
 w\_id Range: 14988 - 15232  
 w\_id Min Warehouse: 1  
 w\_id Max Warehouse: 15232  
 Scale: Normal  
 User Count: 2450  
 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	Delay
New Order	10.00	12.05		18.01		0.10
Payment	10.00	12.05		3.01		0.10
Delivery	1.00	5.05		2.01		0.10
Stock Level	1.00	5.05		2.01		0.10
Order Status	1.00	10.05		2.01		0.10
PARAM2						
3 Tier						
	Txn	Think	Key	RT	RT	Menu
	Weight	Time	Time	Delay	Fence	Delay
New Order	10.09	12.05		18.01		0.10
Payment	9.66	12.05		3.01		0.10
Delivery	0.90	5.05		2.01		0.10
Stock Level	0.90	5.05		2.01		0.10
Order Status	0.90	10.05		2.01		0.10

5.00 0.10  
 5.00 0.10  
 5.00 0.10  
 20.00 0.10  
 5.00 0.10  
 5.00 0.10  
 5.00 0.10  
 20.00 0.10  
 5.00 0.10  
 5.00 0.10  
 5.00 0.10  
 20.00 0.10  
 5.00 0.10

# Appendix D: 60-Day Space

TPC-C 60-Day Space Requirements						
Warehouses	15,232				tpmC	190,510.02
Table	Rows	Data KB	Index KB	Extra 5% KB	8HR Space	Total Space KB
Warehouse	15,232	1,648	160	90.40		1,898.40
District	152,320	17,168	200	868.40		18,236.40
Item	100,000	9,528	184	485.60		10,197.60
New-Order	137,088,000	2,442,552	6,136		1,218,560.00	3,667,248.00
History	456,960,000	27,281,200	105,912		5,480,586.88	32,867,698.88
Orders	456,960,000	14,921,144	7,278,752		4,442,544.31	26,642,440.31
Customer	456,960,000	332,334,552	20,739,880	17,653,721.60		370,728,153.60
Order-Line	4,852,606,786	304,639,640	720,840		61,107,379.25	366,467,859.25
Stock	1,523,200,000	487,424,000	1,031,288	24,422,797.60		512,878,749.04
Totals		1,169,071,432	29,884,016	42,077,963.60	72,249,070.44	1,313,282,482.04
Segment	LogDev Cnt.	Segment Size	Needed	Overhead		Not Needed
misc	20	3,136,000	429,675,579	4,296,756		(430,836,334.63)
big	20	5,696,000	883,606,903	8,836,069		(886,746,972.23)
master, msdb,model	1	13,312	13,312			-
tpcc_root	1	8,192	8,192			-
tempdb	1	8,704	8,704			-
Totals		8,862,208.00	1,313,312,690.04	13,132,824.82		(1,317,583,306.86)
Dynamic Space	346,841,984.00	Sum of Data for Order, Order-Line and History				
Static Space	907,324,252.42	Data + Index + 5% Space + Overhead - Dynamic Space				
Free Space	72,279,278.44	Total Segment Size - Dynamic Space - Static Space - Not needed				
Daily Growth	69,408,473.08	(Dynamic Space/W * 62.5) * tpmC				
Daily Spread	(31,833,431.18)	Free Space - 1.5 * Daily Growth (Zero If Negative)				
60-Day Space (KB)	5,071,832,637.2	Static Space + 60 (Daily Growth + Daily Spread)				
60-Day Space (GB)	4,836.88	60-Day Space in GB (Excludes OS,Paging and RDBMS Logs)				
Available (GB)	9,954.84	Total storage configured and available for database, minus logs, in RAID-0 configuration.				
Log File Storage Requirement						
Log Size (MB)	417,000.00	Total Size of Log File				
% Log Used	33.1079	% of Log File Used During Entire Run				
Total N-O Txn	31,002,140.00	Total Count of New-Order Transactions during Entire Run				
Log / N-O Txn	4.56	KB of Log per New-Order Transaction				
8 Hour Log (GB)	397.68	8 Hours of Log in GB (Excluding Space for Redundancy)				
Log Configured (GB)	474.04					
Disk Capacity	MB	GB				
18.2GB	17,736	16.93				
36.4GB	35,472	33.86				
Space Usage	GB Needed		Disks Priced	Disk Size	GB Priced	GB Usable
60-Day (RAID-0)	4,836.88		532	18.2GB	9,006.76	9,480.80
			28	36.4GB	948.08	948.08
					Total DB	9,954.84
8hr Log (RAID-1)	397.68		28	36.4GB	948.08	474.04
					Total Log	474.04
OS, SQL Server	4.00		1	18.2GB	16.93	16.93
Total Space	5,238.56		589		10,919.85	10,445.81

---

## Appendix E: Third-Party Quotations



Microsoft Corporation  
One Microsoft Way  
Redmond, WA 98052-6399

Tel 425 882 8080  
Fax 425 936 7329  
<http://www.microsoft.com/>

**Microsoft**

July 17, 2003

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

Ms. King:

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

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

Part Number	Description	Unit Price	Quantity	Price
810-00846	<b>SQL Server 2000 Enterprise Edition</b> <i>Per processor licensing Discount Schedule: Open Program Level C Unit Price reflects a 17% discount from the retail unit price of \$19,999.</i>	\$16,541	16	\$264,656
C11-00821	<b>Windows 2000 Server</b> <i>Server license only - No CALs Discount Schedule: Open Program - No Level Unit Price reflects a 8% discount from the retail unit price of \$799.</i>	\$738	8	\$5,904
254-00170	<b>Visual C++ Standard</b> <i>No discounts applied</i>	\$109	1	\$109
PRO-PRORS-16U-01	<b>Database Server Support Package</b> <i>1 Year Term</i>	\$1,950	3	\$5,850

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).

Reference ID: PCchki0317076892

Please include this Reference ID in any correspondence regarding this price quote.



# QUOTATION

Date 07/17/2003  
Quote No. 98037017

Customer  
**Attn: Chris King**  
IBM/RTP  
3039 Cornwallis Road  
RTP, NC 27709  
919-543-0349  
919-543-4045 (fax)

Quoted By: Kristina Homan  
G2FSE: CRAIG BORSACK  
Manufacturer QLOGIC  
Quote valid for 30 days, unless noted  
F.O.B.: Shipping Point

cc:

Please accept the following quotation on behalf of QLOGIC and G2.

L/I	Mfg PN	Cust PN	Qty	Unit Price	Leadtime/ARO
1	QLA2350-BK		1	\$2,095.000	10-12 weeks

Unit price quote is valid for quantity of 1-500.

**Thank you for the opportunity to quote your requirements.  
Please let me know if you have any questions or need additional**

**ATLANTA**  
1-800-648-8978  
770/729-1896Fax

**HUNTSVILLE**  
1-800-648-8978  
256/533-5525Fax

**RALEIGH**  
1-800-648-8978  
919/481-1958Fax

**TAMPA**  
1-800-648-8978  
256/533-5525Fax