

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

Submitted for Review
March 23, 2001

Amended April 6, 2001



First Edition - March 2001

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

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

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

© Copyright International Business Machines Corporation 2001. All rights reserved.

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

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

Trademarks

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

The following terms used in this publication are trademarks of other companies as follows: TPC Benchmark, TPC-C and tpmC are trademark of Transaction Processing Performance Council; Intel, Pentium and Xeon are trademarks or registered trademarks of Intel Corporation; Microsoft, Windows 2000, and BenchCraft are trademarks or registered trademarks of Microsoft Corporation. Other company, product, or service names, which may be denoted by two asterisks (**), may be trademarks or service marks of others.

Notes

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

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

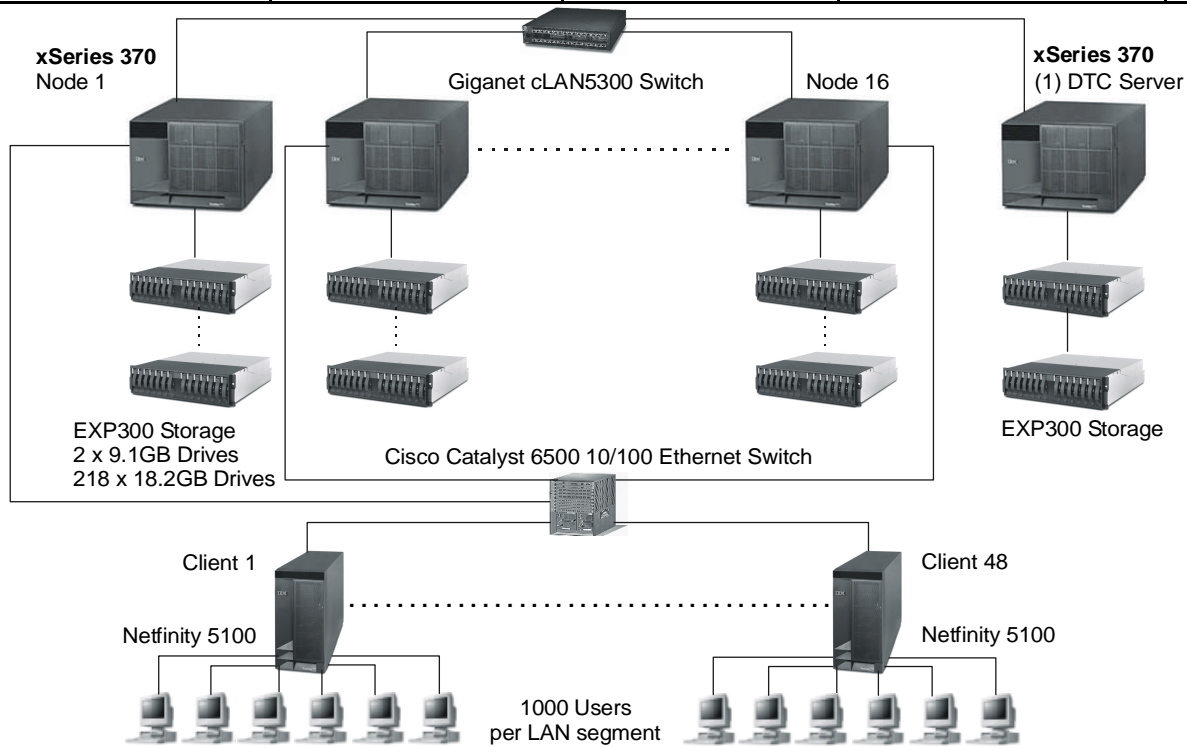


**IBM @server xSeries 370
with SQL Server 2000**

TPC-C Rev. 3.5

Report Date: Mar. 23, 2001
(Amended April 6, 2001)

Total System Cost	TPC-C Throughput	Price/Performance	Availability Date	
\$10,204,816	363,129.75 tpmC	\$28.10/tpmC	May 31, 2001	
Processors	Database Manager	Operating System	Other Software	# Users
Database Servers = 128 DTC Server = 8 Intel^(R) Pentium^(R) III XeonTM 900MHz 2MB L2 Cache	Microsoft^(R) SQL Server 2000	Microsoft Windows^(R) 2000 Datacenter Server	Microsoft COM+ Microsoft Visual C++ 6.0	288,000



System Component	Qty	Each of 16 Database Servers:	Qty	Each of 48 Clients:
Processors	8	900MHz Pentium III Xeon	2	733MHz Pentium III
Cache		w/2MB L2 Cache		w/256KB L2 Cache
Memory	32	512MB ECC SDRAM	4	128MB
Disk Controllers	8	ServeRAID-4H Ultra160 SCSI Adapter	1	Ultra160 SCSI Onboard
Disk Drives	2	9.1GB (10000 rpm)	1	9.1GB Hard Disk
Total Storage	218	18.2GB (10000 rpm)		
Other		58TB (28.995TB online)		
Tape Drive	1	3502-R14 DLT Tape Library	8	One DTC Server:
			4	900MHz/2MB Pentium III Xeon
			1	128MB ECC SDRAM
			1	ServeRAID-4H Ultra160 SCSI Adapter
Interconnect	1	Giganet cLAN-1000 Host Adapter	6	9.1GB (10000 rpm) drive
	1	Giganet cLAN-5300 Switch	1	Giganet cLAN-1000 Host Adapter



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

TPC-C Revision 3.5

Report Date: March 23, 2001
(Amended April 6, 2001)

Description	Order Number	Third-Party Brand	Third-Party Pricing	Unit Price	Qty	Ext. Price	5-Yr. Maint.*
Server Hardware							
xSeries 370 900MHz/2MB Pentium III Xeon*	8681-3RX	IBM	1	\$19,114	17	\$324,938	\$181,645
900MHz/2MB L2 Cache Processor Upgrade	19K4637	IBM	1	6,599	119	785,281	0
8500R Memory Expansion Card	28L4454	IBM	1	625	16	10,000	0
8500 >4X Accelerator Filter	10K2335	IBM	1	1,249	17	21,233	0
512MB ECC SDRAM RDIMM Memory Kit	20L0249	IBM	1	1,575	512	806,400	0
ServeRAID-4H Ultra160 SCSI Adapter	37L6889	IBM	1	2,439	129	314,631	0
Ultra2 SCSI 4m Cable	03K9311	IBM	1	105	369	38,745	0
10/100 Ethernet Server Adapter with IPSec	06P3601	IBM	1	99	16	1,584	0
E54 15" (13.8" Viewable) Color Monitor*	6331N2N	IBM	1	173	17	2,941	4,760
3502-R14 DLT Tape Autoloader*	3502R14	IBM	1	13,779	1	13,779	3,844
IBM Smart-UPS Model 5000RMB	37L6861	IBM	1	3,299	17	56,083	0
Netfinity Rack*	9306900	IBM	1	1,725	40	69,000	52,320
Side Panel Kit	94G6669	IBM	1	195	6	1,170	0
Hardware installation and prep fees		IBM	1			82,700	0
Giganet cLAN-1000 Host Adapter (incl. 10%)	cLAN-1000	Giganet	3	795	19	15,105	75,000
Giganet cLAN5300 30-Port Switch (incl. 10%)	cLAN5300	Giganet	3	15,995	2	31,990	incl. above
Giganet cLAN-A1011 10M Cable (incl. 10%)	cLAN-A1011	Giganet	3	135	19	2,565	incl. above
Storage Hardware							
Netfinity EXP300 Rack Storage Enclosure*	35311RU	IBM	1	3,179	353	1,122,187	277,458
9.1GB 10K Ultra160 SCSI Drive	37L7204	IBM	1	299	38	11,362	0
18.2GB 10K Ultra160 SCSI Drive	37L7205	IBM	1	467	3,488	1,628,896	0
Subtotal						\$5,340,590	\$595,027
Server Software							
Datacenter OS Preload Kit/Ship Group/Warr.	22P4745	IBM	1			\$1,620,000	0
Microsoft SQL Server 2000	11K7641	Microsoft	2	15,802	128	2,022,656	0
Microsoft Windows 2000 Advanced Server	C10-00475	Microsoft	2	2,399	1	2,399	0
5-Year Maintenance for Software		Microsoft	2	2,095	80		\$167,600
Subtotal						\$3,642,656	\$167,600
Client Hardware							
Netfinity 5100 / 733MHz/256KB Pentium III*	8658-2RY	IBM	1	2,052	48	\$98,496	\$283,440
733MHz/256KB Pentium III Upgrade	00N7943	IBM	1	399	48	19,152	0
128MB DIMMs	33L3123	IBM	1	185	144	26,640	0
9.1GB 10K Ultra160 SCSI Drive	37L7204	IBM	1	299	48	14,352	0
Intel Pro/100+ Dual-Port Ethernet Adapter**	8472	Intel	1	219	144	31,536	0
E54 15" (13.8" Viewable) Color Monitor*	6331N2N	IBM	1	173	48	8,304	13,440
Subtotal						\$198,480	\$296,880
Client Software							
Microsoft Windows 2000 Server with COM+	C11-00821	Microsoft	2	738	48	35,424	incl. above
Microsoft Visual C++ Professional 6.0	048-00317	Microsoft	2	549	1	549	incl. above
Subtotal						\$35,973	\$0
User Connectivity							
8-Port 10/100Mbps Hub** (10% spares***)	DEH2924	Generic	5	23	39,600	\$910,800	\$0
Cisco Catalyst 6500 48-Port 10/100 RJ-45	WSX6348RJ45	Cisco	4	45,652	1	45,652	52,500
Subtotal						\$956,452	\$52,500
Total						\$10,174,151	\$1,112,007
Discount						(1,081,342)	\$0
Grand Total						\$9,092,809	\$1,112,007

Large volume discount; prices will vary if purchased individually.

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

Pricing: 1 - IBM; 2 - Microsoft; 3 - Giganet; 4 - Cisco; 5 - Software House International

Audited by Brad Askins of InfoSizing, Inc.

Five-Year Cost of Ownership: \$10,204,816

tpmC Rating: 363,129.75

\$/tpmC: \$28.10

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

Numerical Quantities Summary			
MQTh, Computed Maximum Qualified Throughput: % throughput difference, reported and reproducibility runs:			363,129.75 tpmC 0.03%
Response Times (in seconds)	90%	Average	Maximum
New-Order	0.31	0.22	392.15
Payment	0.28	0.19	315.38
Delivery (Interactive)	0.11	0.10	70.07
Stock-Level	0.56	0.36	52.68
Order Status	0.20	0.17	10.58
Delivery (Deferred)	0.24	0.17	6.13
Menu	0.11	0.11	665.21
Transaction Mix (in percent of total transactions)		Total Occurrences	Percent
New-Order		7,262,595	44.84
Payment		6,973,062	43.06
Delivery		652,421	4.03
Stock Level		653,068	4.03
Order Status		652,396	4.04
Emulation Delay (in seconds)	Response Time		Menu
New-Order	0.1		0.1
Payment	0.1		0.1
Order-Status	0.1		0.1
Delivery	0.1		0.1
Stock-Level	0.1		0.1
Keying/Think Times (in seconds)	Minimum	Average	Maximum
New-Order	18.00 / 0.00	18.01 / 12.05	18.06 / 120.50
Payment	3.00 / 0.00	3.02 / 12.04	3.06 / 120.50
Delivery	2.00 / 0.00	2.02 / 5.05	2.06 / 50.50
Stock Level	2.00 / 0.00	2.02 / 5.05	2.03 / 50.50
Order Status	2.00 / 0.00	2.02 / 10.04	2.06 / 100.50
Test Duration			
Ramp-up time			29 minutes 28 seconds
Measurement interval			20 minutes
Number of transactions (all types) completed in measurement interval			16,774,818
Ramp-down time			21 minutes 45 seconds
Number of checkpoints in measurement interval			1
Checkpoint interval			20 minutes

Table of Contents

Preface	11
General Items	15
Application Code and Definition Statements	15
Benchmark Sponsor	15
Parameter Settings	15
Configuration Diagrams	15
<i>Measured Configuration</i>	16
<i>Priced Configuration</i>	17
<i>Disk Drive Substitution</i>	17
<i>Network Configuration</i>	17
Clause 1: Logical Database Design Related Items	18
Table Definitions	18
Physical Organization of the Database	18
Insert and Delete Operations	18
Horizontal or Vertical Partitioning	18
Replication	18
Table Attributes	18
Clause 2: Transaction and Terminal Profiles Related Items	19
Random Number Generation	19
Screen Layout	19
Terminal Verification	19
Intelligent Terminals	19
Transaction Profiles	19
Deferred Delivery Mechanism	20
Clause 3: Transaction and System Properties Related Items	21
Atomicity Requirements	21
<i>Completed Transactions</i>	21
<i>Aborted Transactions</i>	21
Consistency Requirements	21
Isolation Requirements	22
<i>Failure of Memory and Instantaneous Interruption</i>	22
<i>Loss of Log Disk, Data Disk, and Interconnect</i>	22
<i>Loss of DTC Server</i>	23
Clause 4: Scaling and Database Population Related Items	24
Cardinality of Tables	24
Distribution of Tables and Logs	24
Database Model Implemented	26
Partitions/Replications Mapping	27
180-Day Space Requirement	27
Clause 5: Performance Metrics and Response Time Related Items	28
Measured tpmC	28
Response Times	28
Keying/Think Times	28
Response Time Frequency Distribution Curves	29
Steady State Methodology	33
Work Performed during Steady State	33
<i>Transaction Flow</i>	33
Checkpoints	33
Reproducibility Methodology	33
Measurement Interval	33
Transaction Mix	34
Percentage of Total Mix	34
Clause 6: SUT, Driver and Communication Definition Related Items	36

Description of RTE	36
Emulated Components	36
Benchmarked and Targeted System Configuration Diagrams	36
Network Configuration	36
Network Bandwidth	37
Operator Intervention	37
Clause 7: Pricing Related Items	38
Hardware and Software Components	38
Availability Date	38
Measured tpmC	38
Country-Specific Pricing	38
Usage Pricing	38
System Pricing	39
Clause 9: Audit Related Items	40
Auditor	40
Availability of the Full Disclosure Report	40
<i>Attestation letter</i>	41
Appendix A: Source Code	43
Web Client Source code	43
<i>dlldata.c</i>	43
<i>error.h</i>	43
<i>install.c</i>	45
<i>install.h</i>	54
<i>install.rc</i>	54
<i>install_com.cpp</i>	57
<i>install_resource.h</i>	61
<i>isapi_dll_resource.h</i>	61
<i>license.txt</i>	61
<i>methods.h</i>	63
<i>Readregistry.cpp</i>	65
<i>ReadRegistry.h</i>	66
<i>ReadWHouse.cpp</i>	66
<i>ReadWHouse.h</i>	69
<i>rtetime.h</i>	69
<i>SetAuditTime.cpp</i>	70
<i>spinlock.h</i>	72
<i>tpcc.cpp</i>	73
<i>tpc.def</i>	98
<i>tpcc.h</i>	98
<i>tpcc.rc</i>	100
<i>tpcc_com.cpp</i>	101
<i>tpcc_com.h</i>	103
<i>tpcc_com_all.cpp</i>	105
<i>tpcc_com_all.def</i>	110
<i>tpcc_com_all.h</i>	110
<i>tpcc_com_all.idl</i>	111
<i>tpcc_com_all.rc</i>	112
<i>tpcc_com_all.rgs</i>	113
<i>tpcc_com_all_i.c</i>	113
<i>tpcc_com_no.rgs</i>	114
<i>tpcc_com_os.rgs</i>	114
<i>tpcc_com_pay.rgs</i>	114
<i>tpcc_com_ps.def</i>	115
<i>tpcc_com_ps.h</i>	115
<i>tpcc_com_ps.idl</i>	117

<i>tpcc_com_ps_i.c</i>	117
<i>tpcc_com_ps_p.c</i>	118
<i>tpcc_com_rem.cpp</i>	129
<i>tpcc_com_rem.h</i>	130
<i>tpcc_com_remote.cpp</i>	132
<i>tpcc_com_remote.def</i>	137
<i>tpcc_com_remote.h</i>	137
<i>tpcc_com_remote.idl</i>	137
<i>tpcc_com_remote.rc</i>	138
<i>tpcc_com_remote.rgs</i>	139
<i>tpcc_com_remote_i.c</i>	139
<i>tpcc_com_remote_Methods.h</i>	139
<i>tpcc_com_remote_ps.def</i>	141
<i>tpcc_com_remote_ps.h</i>	141
<i>tpcc_com_remote_ps.idl</i>	143
<i>tpcc_com_remote_ps_i.c</i>	143
<i>tpcc_com_remote_ps_p.c</i>	144
<i>tpcc_com_remote_resource.h</i>	155
<i>tpcc_com_sl.rgs</i>	155
<i>tpcc_odbc.cpp</i>	155
<i>tpcc_odbc.h</i>	164
<i>trans.h</i>	166
<i>txn_base.h</i>	168
<i>txnlog.h</i>	168
Stored Procedures	171
<i>delivery.sql</i>	171
<i>neword.sql</i>	172
<i>ordstat.sql</i>	175
<i>payment.sql</i>	176
<i>stocklev.sql</i>	178
Appendix B: Database Design	180
Database Build	180
<i>16x1800</i>	180
<i>add_constraints_t1.sql</i>	180
<i>add_constraints_t2.sql</i>	180
<i>add_constraints_t3.sql</i>	181
<i>add_constraints_t4.sql</i>	181
<i>add_constraints_t5.sql</i>	182
<i>add_constraints_t6.sql</i>	182
<i>add_constraints_t7.sql</i>	183
<i>add_constraints_t8.sql</i>	184
<i>add_constraints_t9.sql</i>	184
<i>add_constraints_t10.sql</i>	185
<i>add_constraints_t11.sql</i>	185
<i>add_constraints_t12.sql</i>	186
<i>add_constraints_t13.sql</i>	186
<i>add_constraints_t14.sql</i>	187
<i>add_constraints_t15.sql</i>	187
<i>add_constraints_t16.sql</i>	188
<i>add_remote_servers.sql</i>	189
<i>add_views1.sql</i>	190
<i>add_views2.sql</i>	192
<i>add_views3.sql</i>	194
<i>add_views4.sql</i>	197
<i>add_views5.sql</i>	199

<i>add_views6.sql</i>	201
<i>add_views7.sql</i>	204
<i>add_views8.sql</i>	206
<i>add_views9.sql</i>	208
<i>add_views10.sql</i>	211
<i>add_views11.sql</i>	213
<i>add_views12.sql</i>	215
<i>add_views13.sql</i>	217
<i>add_views14.sql</i>	220
<i>add_views15.sql</i>	222
<i>add_views16.sql</i>	224
<i>item_iot1.sql</i>	226
<i>item_iot2.sql</i>	227
<i>item_iot3.sql</i>	228
<i>item_iot4.sql</i>	229
<i>item_iot5.sql</i>	230
<i>item_iot6.sql</i>	231
<i>item_iot7.sql</i>	231
<i>item_iot8.sql</i>	232
<i>item_iot9.sql</i>	233
<i>item_iot10.sql</i>	234
<i>item_iot11.sql</i>	235
<i>item_iot12.sql</i>	236
<i>item_iot13.sql</i>	237
<i>item_iot14.sql</i>	238
<i>item_iot15.sql</i>	238
<i>item_iot16.sql</i>	239
<i>backup.sql</i>	240
<i>backupdev.sql</i>	240
<i>createdb.sql</i>	240
<i>tables.sql</i>	242
<i>dbopt1.sql</i>	243
<i>dbopt2.sql</i>	244
<i>idxcuscl.sql</i>	244
<i>idxcusnc.sql</i>	245
<i>idxdiscl.sql</i>	245
<i>idxitmcl.sql</i>	245
<i>idxnodcl.sql</i>	245
<i>idxodlcl.sql</i>	246
<i>idxordcl.sql</i>	246
<i>idxordnc.sql</i>	246
<i>idxstkcl.sql</i>	246
<i>idxwarcl.sql</i>	247
<i>version.sql</i>	247
Load Source Code	247
<i>getargs.c</i>	247
<i>random.c</i>	249
<i>strings.c</i>	251
<i>time.c</i>	254
<i>tpcc.h</i>	254
<i>tpcldr.c</i>	255
<i>tpcldr.mak</i>	279
Appendix C: Tunable Parameters	283
Database Startup Parameters	283
Database Information	283

Software Update	284
Microsoft Windows 2000 Datacenter Configuration	284
<i>boot.ini</i>	284
<i>Enabling AWE</i>	284
<i>Enabling VIA</i>	284
<i>Specifying Remote DTC</i>	284
<i>System Information Report for Software Configuration</i>	284
<i>System Information Report for Hardware Configuration</i>	285
DTC Server Configuration	347
<i>Software Configuration</i>	347
<i>Hardware Configuration</i>	347
Client Configuration	355
<i>Transaction Monitor: COM+ Settings on Clients</i>	355
<i>Windows Registry Editor Version 5.00</i>	356
<i>System Information Report for Software Configuration</i>	357
<i>System Information Report for Hardware Configuration</i>	357
RTE Input Parameters	363
Appendix D: 180-Day Space	434
Appendix E: Third-Party Quotations	435

Preface

The TPC Benchmark™ C was developed by the Transaction Processing Performance Council (TPC). The TPC was founded to define transaction processing benchmarks and to disseminate objective, verifiable performance data to the industry. This full disclosure report is based on the TPC Benchmark C Standard Specification Version 3.5, released October 25, 1999.

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

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

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

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

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

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

General Items

Benchmark Sponsor

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

This benchmark was sponsored by International Business Machines Corporation.

Application Code and Definition Statements

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

Appendix A contains all source code implemented in this benchmark.

Parameter Settings

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

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

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

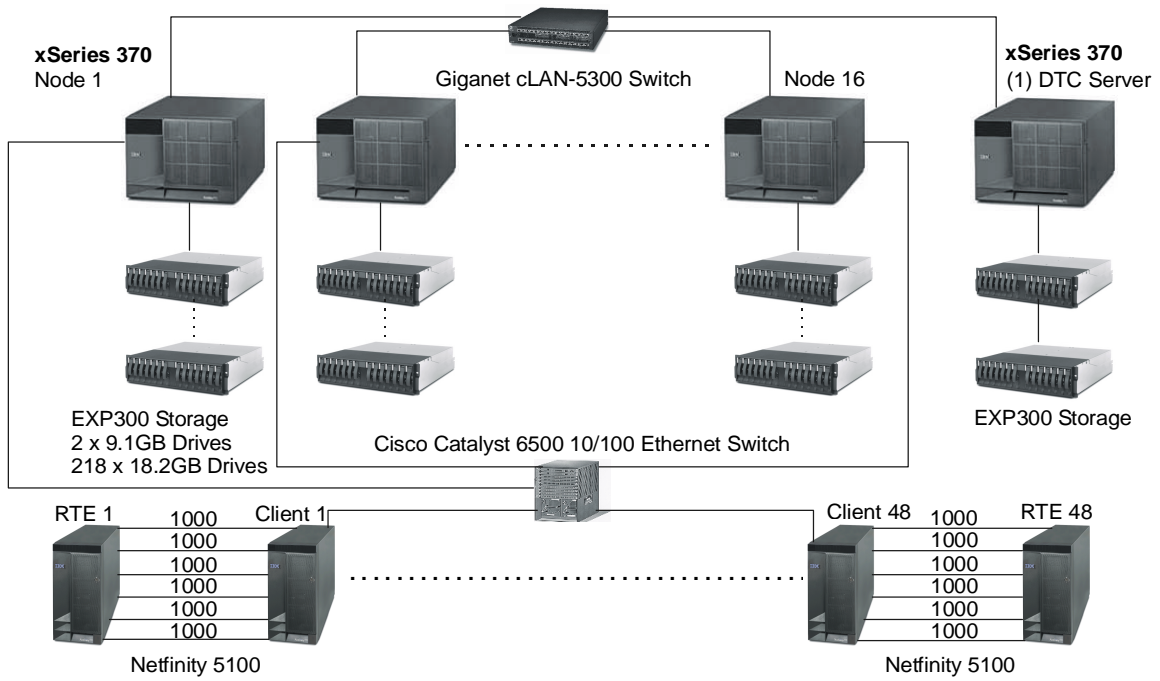
Appendix C contains the tunable parameters for the database, the operating system, and the transaction monitor.

Configuration Diagrams

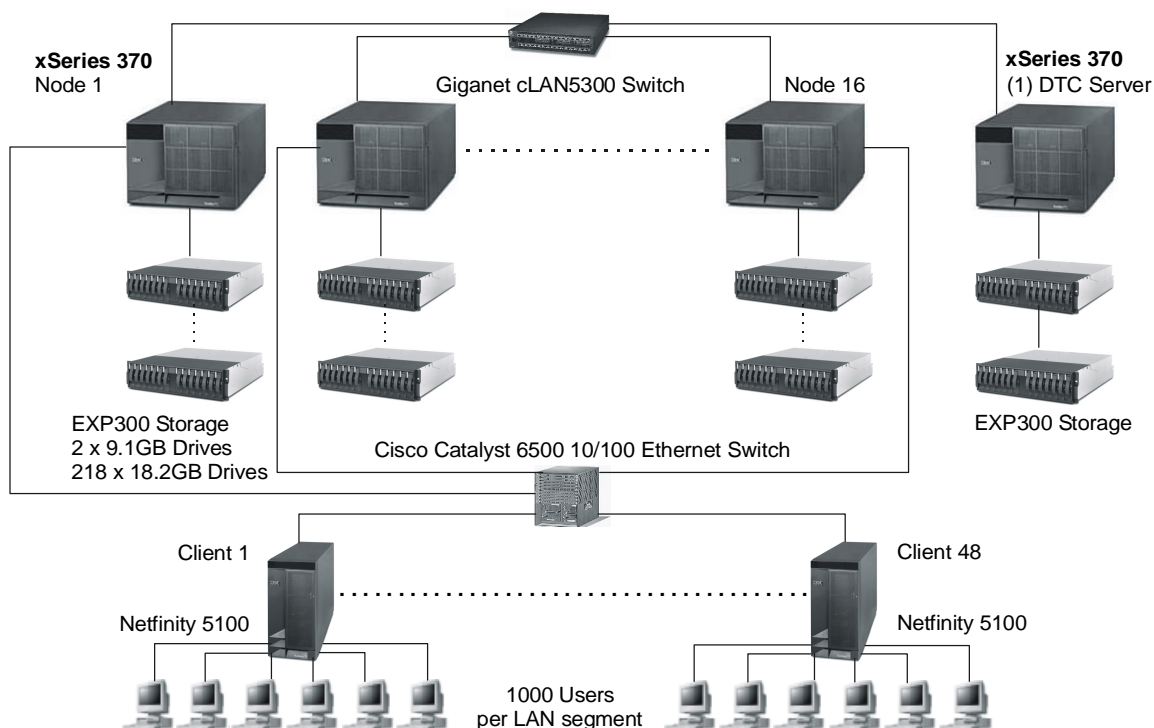
Diagrams of both measured and priced configurations must be provided, accompanied by a description of the differences.

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

Measured Configuration



Priced Configuration



The priced and measured configurations were identical with two exceptions. For a full description of the hardware and software components used in the priced configuration, see the Executive Summary at the front of this report.

Disk Drive Substitution

Disk drive substitution was used for the 18.2GB 10K Wide Ultra SCSI drives. The measured configuration used 3,450 18.2GB 10K-3 Wide Ultra SCSI drives (P/N 36L9749) and 38 18.2GB 10K Ultra160 SCSI drives (P/N 37L7205). The 38 drives were configured on one of the 32 nodes as follows:

- Eight (8) drives were configured as a RAID-10 disk array for the database log.
- Thirty (30) drives were configured as two 15-disk RAID-1E arrays for database tables.

The priced configuration included 3,488 18.2GB 10K Ultra160 SCSI drives (P/N 37L7205). I/O performance data reviewed with the auditor showed that the performance of the priced drives was equivalent to the performance of the drives used in the measured configuration for both the database log and database tables.

Network Configuration

In the measured configuration, the six LAN segment connections between most of the RTE-client pairs used Ethernet cross-over cables. The link speed for each Ethernet adapter port was set at 10Mbps. On one RTE-client pair, the six LAN segment connections used Type 5 Ethernet cables and 10Mbps Ethernet hubs. The priced configuration included 10Mbps hubs. Benchcraft was used to generate separate transaction reports for an RTE-client pair that used cross-over cables and an RTE-client pair that used Ethernet hubs. The response time data indicated no difference in performance between the use of hubs vs. cross-over cables. The data was submitted to the auditor. An additional IBM 10/100 Ethernet Server Adapter was installed on each node. This adapter provided connectivity to the IBM campus LAN and served no function during the benchmark. The additional adapter was not included in the priced configuration.

Clause 1: Logical Database Design Related Items

Table Definitions

Listings must be provided for all table definition statements and all other statements used to set up the database. (8.1.2.1)

Appendix B contains the code used to define and load the database tables.

Physical Organization of the Database

The physical organization of tables and indexes within the database must be disclosed. (8.1.2.2)

Physical space was allocated to SQL Server 2000 on the server disks as detailed in Figure 4-2.

Insert and Delete Operations

It must be ascertained that insert and/or delete operations to any of the tables can occur concurrently with the TPC-C transaction mix. Furthermore, any restriction in the SUT database implementation that precludes inserts beyond the limits defined in Clause 1.4.11 must be disclosed. This includes the maximum number of rows that can be inserted and the maximum key value for these new rows. (8.1.2.3)

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

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

Horizontal or Vertical Partitioning

While there are few restrictions placed upon horizontal or vertical partitioning of tables and rows in the TPC-C benchmark (see Clause 1.6), any such partitioning must be disclosed. (8.1.2.4)

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

Replication

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

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

Table Attributes

Additional and/or duplicated attributes in any table must be disclosed, along with a statement on the impact on performance (see Clause 1.4.7). (8.1.2.6)

No additional attributes were used in this benchmark.

Clause 2: Transaction and Terminal Profiles Related Items

Random Number Generation

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

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

Screen Layout

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

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

Terminal Verification

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

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

Intelligent Terminals

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

The terminals emulated in the priced configuration are IBM PC desktop computer systems. All processing of the input/output screens was handled by the Netfinity 5100 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 xSeries 370 server.

Transaction Profiles

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

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

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

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

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

The percentage of Delivery transactions that were skipped as a result of an insufficient number of rows in the NEW-ORDER table must be disclosed. (8.1.3.10)

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

Table 2-1. Transaction Statistics

New Order	Value (%)
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
Payment	
Home warehouse payment transactions	84.99
Remote warehouse payment transactions	15.01
Non-Primary Key Access	
Payment transactions using C_LAST	60.02
Order-Status transactions using C_LAST	60.02
Delivery	
Delivery transactions skipped	0
Transaction Mix	
New-Order	44.84
Payment	43.06
Delivery	4.03
Stock Level	4.03
Order Status	4.04

Deferred Delivery Mechanism

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

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

The source code is listed in Appendix A.

Clause 3: Transaction and System Properties Related Items

The results of the ACID test must be disclosed, along with a description of how the ACID requirements were met. This includes disclosing which case was followed for the execution of Isolation Test 7. (8.1.4.1)

Atomicity Requirements

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

All ACID tests were conducted according to specification.

Completed Transactions

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

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

Aborted Transactions

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

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

Consistency Requirements

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

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

Isolation Requirements

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

Isolation tests one through seven were run using shell scripts to issue queries to the database. The shell scripts were run twice, once on a node that had the warehouse being used locally and then again on a node that did not have the warehouse being used locally. Each script included timestamps to demonstrate the concurrency of operations. The results of the queries were captured and placed in files. The auditor reviewed the results and verified that the isolation requirements had been met.

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

Case A was followed for Isolation test seven.

Durability Requirements

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

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

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

Failure of Memory and Instantaneous Interruption

The two tests were combined into a single test. The test was performed as follows:

1. The four Consistency conditions were tested.
2. The current count of the total number of orders, SUM1, was determined by obtaining the sum of the D_NEXT_O_ID column for all rows in the District table.
3. A measurement was started under full load with all users submitting transactions. When all users were submitting transactions, a checkpoint was executed.
4. The checkpoint lasted 9 minutes. During that time, the measurement throughput reached a minimum of 90 percent of the reported throughput and maintained that level for at least 5 minutes.
5. After the checkpoint had completed, a single server node was powered off. Benchcraft was paused, then stopped.
6. The single server node was then powered on.
7. When the single server node was fully booted, SQL Server was started to initiate automatic recovery from the log.
8. After the database recovery completed, all the nodes were checkpointed. When the checkpoint was complete, the four Consistency tests were successfully executed. The current count of the total number of orders, SUM2, was obtained.
9. The following relationship was verified:
SUM2 >= (SUM1 + New Order completed - New Orders rolled back)

Loss of Log Disk, Data Disk, and Interconnect

These three tests were combined for loss of single durable medium failures and loss of interconnect. The following steps were performed:

1. The four Consistency conditions were tested.
2. The current count of the total number of orders, SUM1, was determined by obtaining the sum of the D_NEXT_O_ID column for all rows in the District table.

3. A measurement was started under full load with all users submitting transactions. When all users were submitting transactions, a checkpoint was executed.
4. The checkpoint lasted 9 minutes. During that time the measurement throughput reached a minimum of 90 percent of the reported throughput and maintained that level until the run was paused.
5. After the checkpoint had completed, one disk from the database log configuration array on one node was removed.
6. Since the disk was part of a RAID-10 array, SQL Server continued to process transactions without interruption.
7. A disk containing a portion of each of the tables in the database was removed from one node.
8. Since the disk was RAID-1E protected, SQL Server continued to process transactions without interruption.
9. Power was removed from the interconnect switch and Benchcraft was paused and then stopped.
10. The database logged MSDTC errors because it was unable to complete remote transactions or communicate with the DTC Servers.
11. The power cords were reconnected to the Gigaset switches. There was a burst of I/O on the database drives as the fabric came back up.
12. The database was checkpointed. The database was stopped on all nodes.
13. The nodes were booted and the database was started again. All nodes automatically initiated database recovery. Recovery took a few seconds with 0 to 3 transactions rolled back.
14. After the run was completed, the log and data disks removed in steps 5 and 7 were replaced and the ServeRAID controller rebuilt each drive.
15. The four Consistency conditions were again tested successfully.
16. Step 1 was repeated to obtain the current count of the total number of orders, SUM2.
17. The following relationship was verified:

$$\text{SUM2} \geq (\text{SUM1} + \text{New Order completed} - \text{New Orders rolled back})$$

Loss of DTC Server

1. The four Consistency conditions were tested.
2. The current count of the total number of orders, SUM1, was determined by obtaining the sum of the D_NEXT_O_ID column for all rows in the District table.
3. A measurement was started under full load with all users submitting transactions. When all users were submitting transactions a checkpoint was executed.
4. The checkpoint lasted 9 minutes. During that time the measurement throughput reached a minimum of 90 percent of the reported throughput and maintained that level until the run was paused.
5. After the checkpoint had completed, the DTC server was powered off. Benchcraft was paused, and then stopped.
6. The DTC server was powered on. When the DTC Server, booted the DTC service was started automatically. When the DTC service started, there was a burst of database I/O.
7. The database was checkpointed.
8. After the checkpoint completed, the four Consistency tests were successfully executed. The current count of the total number of orders, SUM2, was obtained.
9. The following relationship was verified:

$$\text{SUM2} \geq (\text{SUM1} + \text{New Order completed} - \text{New Orders rolled back})$$

Clause 4: Scaling and Database Population Related Items

Cardinality of Tables

The cardinality (e.g., the number of rows) of each table, as it existed at the start of the benchmark run (see Clause 4.2), must be disclosed. If the database was over-scaled and inactive rows of the WAREHOUSE table were deleted (see Clause 4.2.2), the cardinality of the WAREHOUSE table as initially configured and the number of rows deleted must be disclosed. (8.1.5.1)

The database was built with 28,800 warehouses, and the audited run used 28,800 warehouses.

Table 4-1. Initial Cardinality of Tables

Table Name	Rows
Warehouse	28,800
District	288,000
Customer	864,000,000
History	864,000,000
Orders	864,000,000
New Order	259,200,000
Order Line	8,639,860,511
Stock	2,880,000,000
Item	100,000
Inactive Warehouses	0

Distribution of Tables and Logs

The distribution of tables and logs across all media must be explicitly depicted for the tested and priced systems. (8.1.5.2)

The following series of table depicts the database configuration of the tested system to meet the 8-hour steady state requirement. The configuration was the same on all 16 nodes. The big_files file group contains the t_customer and t_stock tables. The misc_file group contains the other TPC-C tables.

Table 4-2. Distribution of Tables and Logs

Controller	Disk	Drives	Partition	Size	Use
1	Disk 0	2 - 9.1GB RAID-1 8 - 18.2GB RAID-10	C: NTFS	8.47GB	OS DB Log Unallocated
	Disk 1		E: RAW	66.41GB 1.39GB	
2	Disk 2	15 - 18.2GB RAID-1E	1 RAW	8.55GB	C:\devjp\big_files_1 C:\devjp\misc_files_1 backup1 Free
			2 RAW	4.05GB	
			F: NTFS	90.05GB 27.42GB	
	Disk 3	15 - 18.2GB RAID-1E	1 RAW	8.55GB	C:\devjp\big_files_2 C:\devjp\misc_files_2 Free
			2 RAW	4.05GB 117.47GB	
3	Disk 4	15 - 18.2GB RAID-1E	1 RAW	8.55GB	C:\devjp\big_files_3 C:\devjp\misc_files_3 backup2 Free
			2 RAW	4.05GB	
			G: NTFS	90.05GB 27.42GB	
	Disk 5	15 - 18.2GB RAID-1E	1 RAW	8.55GB	C:\devjp\big_files_4 C:\devjp\misc_files_4 Free
			2 RAW	4.05GB 117.47GB	
4	Disk 6	15 - 18.2GB RAID-1E	1 RAW	8.55GB	C:\devjp\big_files_5 C:\devjp\misc_files_5 backup3 Free
			2 RAW	4.05GB	
			H: NTFS	90.05GB 27.42GB	
	Disk 7	15 - 18.2GB RAID-1E	1 RAW	8.55GB	C:\devjp\big_files_6 C:\devjp\misc_files_6 Free
			2 RAW	4.05GB 117.47GB	
5	Disk 8	15 - 18.2GB RAID-1E	1 RAW	8.55GB	C:\devjp\big_files_7 C:\devjp\misc_files_7 backup4 Free
			2 RAW	4.05GB	
			K: NTFS	90.05GB 27.42GB	
	Disk 9	15 - 18.2GB RAID-1E	1 RAW	8.55GB	C:\devjp\big_files_8 C:\devjp\misc_files_8 Free
			2 RAW	4.05GB 117.47GB	
6	Disk 10	15 - 18.2GB RAID-1E	1 RAW	8.55GB	C:\devjp\big_files_9 C:\devjp\misc_files_9 Root .MDF file Free
			2 RAW	4.05GB	
			J: NTFS	90.05GB 27.42GB	
	Disk 11	15 - 18.2GB RAID-1E	1 RAW	8.55GB	C:\devjp\big_files_10 C:\devjp\misc_files_10 Free
			2 RAW	4.05GB 117.47GB	

Controller	Disk	Drives	Partition	Size	Use
7	Disk 12	15 - 18.2GB RAID-1E	1 RAW 2 RAW	8.55GB 4.05GB 117.47GB	C:\devjp\big_files_11 C:\devjp\misc_files_11 Free
	Disk 13	15 - 18.2GB RAID-1E	1 RAW 2 RAW	8.55GB 4.05GB 117.47GB	C:\devjp\big_files_12 C:\devjp\misc_files_12 Free
8	Disk 14	15 - 18.2GB RAID-1E	1 RAW 2 RAW	8.55GB 4.05GB 117.47GB	C:\devjp\big_files_13 C:\devjp\misc_files_13 Free
	Disk 15	15 - 18.2GB RAID-1E	1 RAW 2 RAW	8.55GB 4.05GB 117.47GB	C:\devjp\big_files_14 C:\devjp\misc_files_14 Free

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

Database Model Implemented

A statement must be provided that describes:

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

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

Partitions/Replications Mapping

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

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

180-Day Space Requirement

Details of the 180-day space computations, along with proof that the database is configured to sustain 8 hours of growth for the dynamic tables (Order, Order-Line, and History) must be disclosed (see Clause 4.2.3). (8.1.5.5)

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

Clause 5: Performance Metrics and Response Time Related Items

Measured tpmC

Measured tpmC must be reported. (8.1.6.1)

Measured tpmC: 363,129.75 tpmC

Price per tpmC: \$28.10 per tpmC

Response Times

Ninetieth percentile, maximum and average response times must be reported for all transaction types as well as for the Menu response time. (8.1.6.2)

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

Table 5-1. Response Times in Seconds

Transaction Type	Average	Maximum	90th %
New-Order	0.22	392.15	0.31
Payment	0.19	315.38	0.28
Delivery (interactive)	0.10	70.07	0.11
Stock Level	0.36	52.68	0.56
Order Status	0.17	10.58	0.20
Delivery (deferred)	0.17	6.13	0.24
Menu	0.11	665.21	0.11

Keying/Think Times

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

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

Table 5-2. Keying/Think Times

Transaction Type	Minimum	Average	Maximum
New-Order	18.00 / 0.00	18.01 / 12.05	18.06 / 120.50
Payment	3.00 / 0.00	3.02 / 12.04	3.06 / 120.50
Delivery	2.00 / 0.00	2.02 / 5.05	2.06 / 50.50
Stock Level	2.00 / 0.00	2.02 / 5.05	2.03 / 50.50
Order Status	2.00 / 0.00	2.02 / 10.04	2.06 / 100.50

Response Time Frequency Distribution Curves

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

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

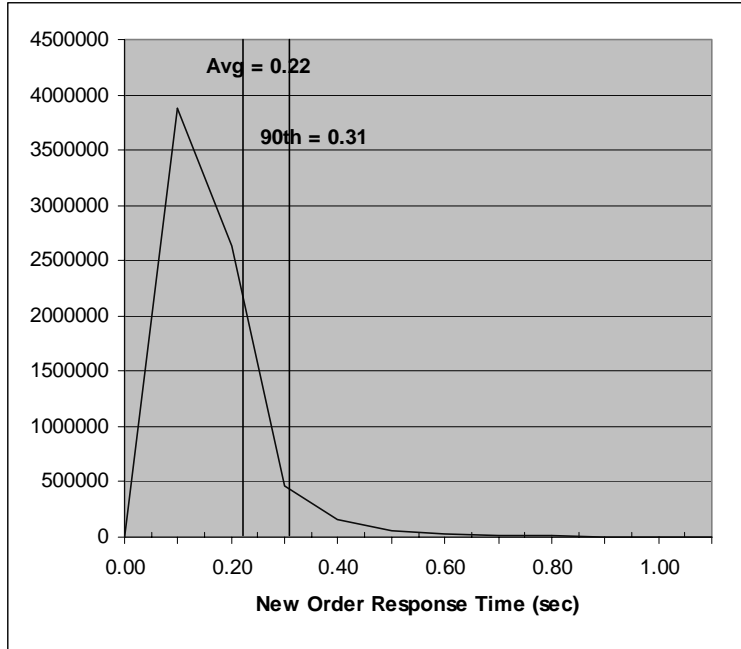


Figure 5-2. Payment Transaction - Response Time Frequency Distribution

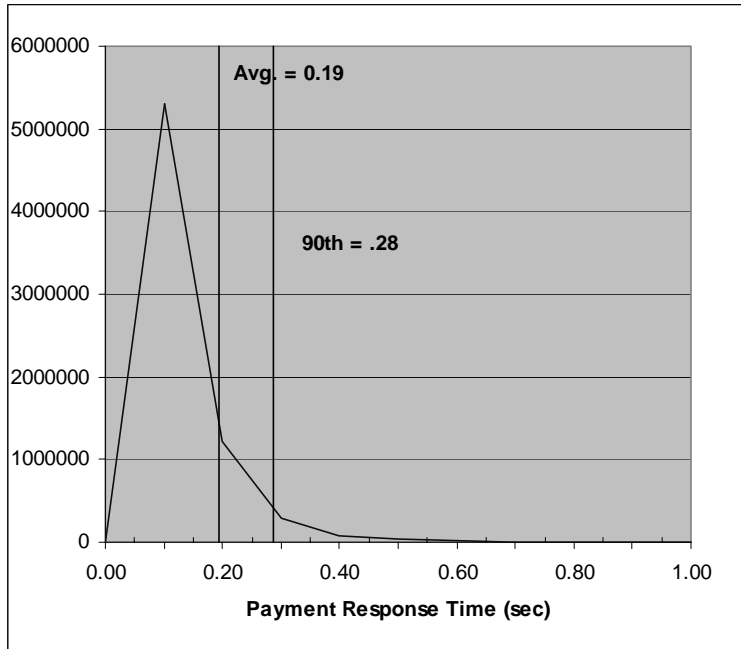


Figure 5-3. Order-Status Transaction - Response Time Frequency Distribution

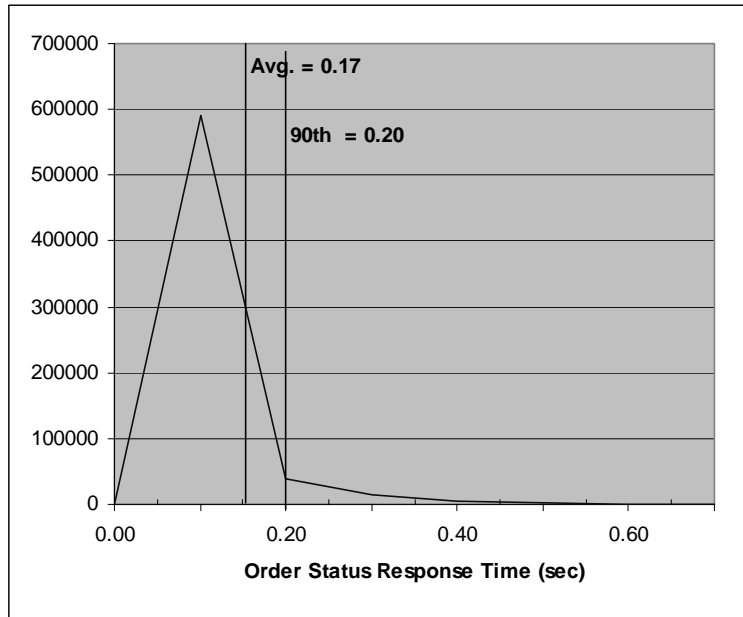


Figure 5-4. Delivery Transaction - Response Time Frequency Distribution

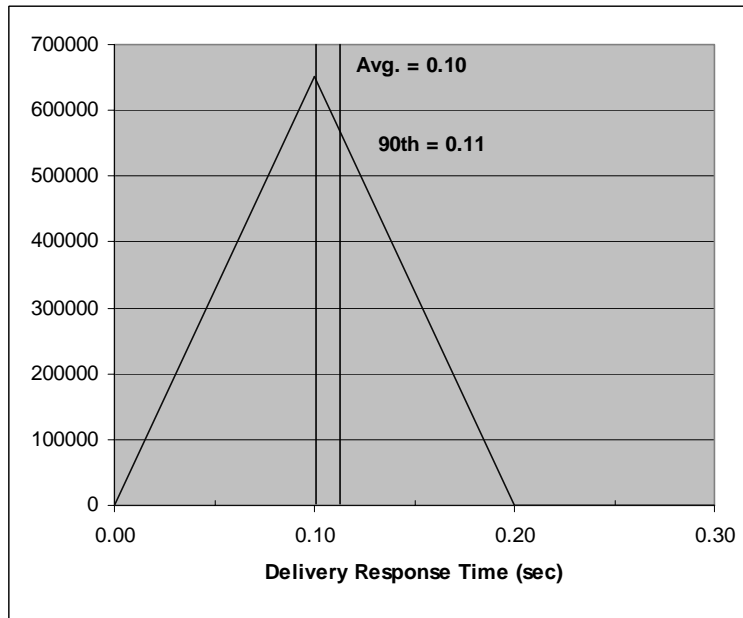
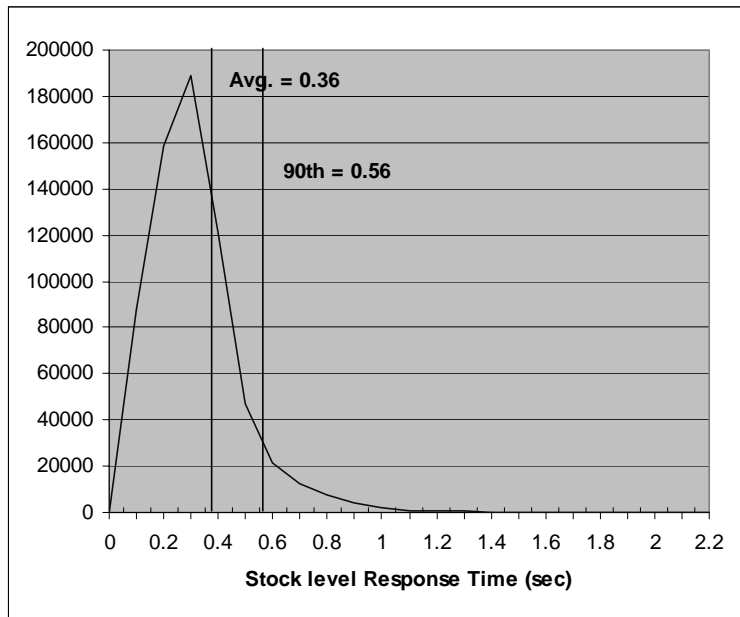


Figure 5-5. Stock-Level Transaction - Response Time Frequency Distribution



Performance Curve for Response Time vs. Throughput

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

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

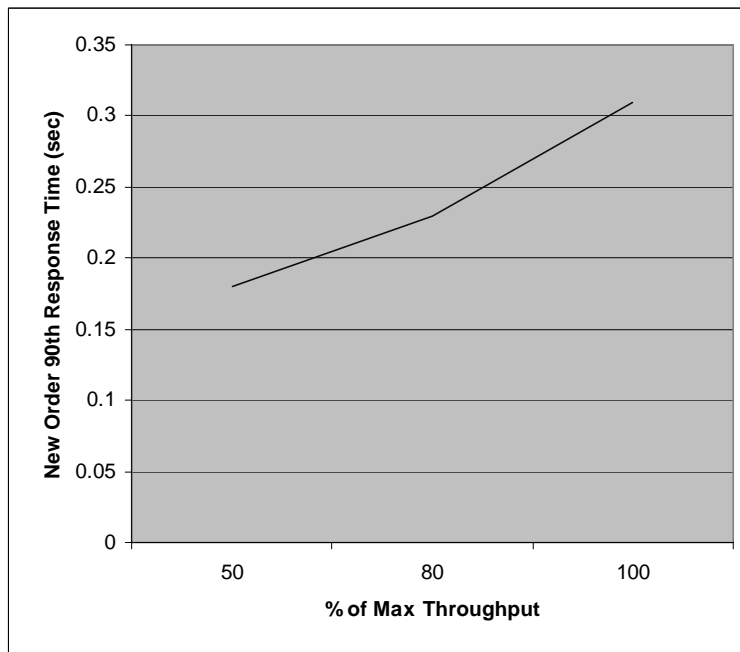


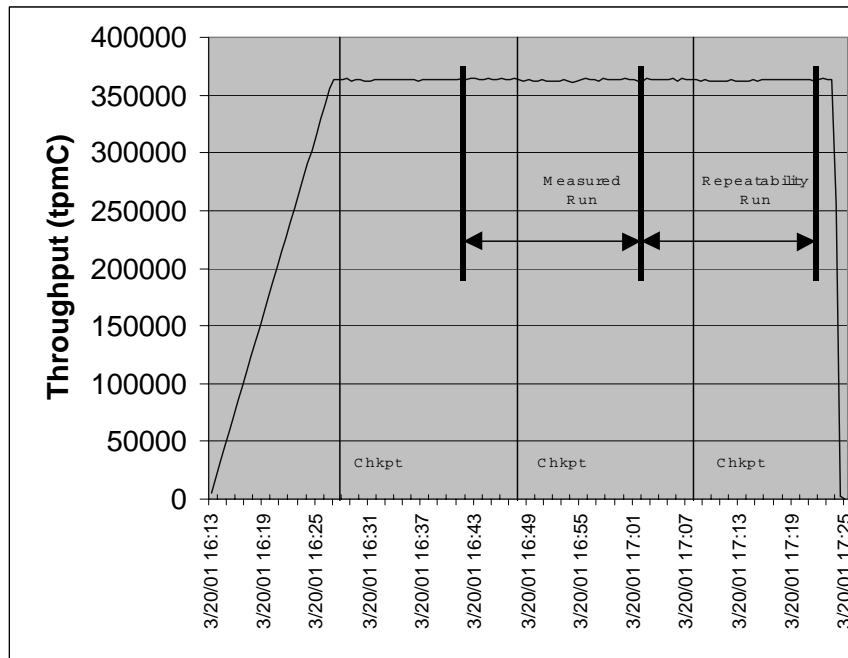
Figure 5-7. New-Order Think Time Distribution



Throughput vs. Elapsed Time

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

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



Steady State Methodology

The method used to determine that the SUT had reached a steady state prior to commencing the measurement interval (see Clause 5.5) must be described. (8.1.6.9)

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

Work Performed during Steady State

A description of how the work normally performed during a sustained test (e.g., checkpointing, writing redo/undo log records) actually occurred during the measurement interval must be reported. (8.1.6.10)

Transaction Flow

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

The RTE generated the required input data for the chosen transaction. It waited to complete the minimum required key time before transmitting the input screen. The transmission was time-stamped. The return of the screen with the required response data was time-stamped. The difference between these two time-stamps was the response time for that transaction and was logged in the RTE log.

The RTE then waited the required think time interval before repeating the process starting at selecting another transaction from the menu.

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

Checkpoints

Checkpoints were executed on the server during the ramp-up phase and at 20-minute intervals. Each measured run contained one checkpoint. SQL Server was started with trace flag 3502, which causes it to log the occurrence of the checkpoint. This information was used to verify that the checkpoints occurred at the appropriate times during the test run.

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

Reproducibility Methodology

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

A repeatability measurement was taken on the xSeries 370 server for the same length of time as the measured run. The repeatability measurement was 363,015.60 tpmC.

Measurement Interval

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

The measurement interval was 20 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

New Order	Value (%)
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
Payment	
Home warehouse payment transactions	84.99
Remote warehouse payment transactions	15.01
Non-Primary Key Access	
Payment transactions using C_LAST	60.02
Order-Status transactions using C_LAST	60.02
Delivery	
Delivery transactions skipped	0
Transaction Mix	
New-Order	44.84
Payment	43.06
Delivery	4.03
Stock Level	4.03
Order Status	4.04

Clause 6: SUT, Driver and Communication Definition Related Items

Description of RTE

The RTE input parameters, code fragments, functions, etc., used to generate each transaction input field must be disclosed. (8.1.7.1)

The RTE used was Microsoft BenchCraft V2.0 RTE. Benchcraft is a proprietary tool provided by Microsoft and is not commercially available. The RTE input is listed in Appendix C.

Emulated Components

It must be demonstrated that the functionality and performance of the components being emulated in the Driver System are equivalent to that of the priced system. The results of the test described in Clause 6.6.3.4 must be disclosed. (8.1.7.2)

No components were emulated.

Benchmarked and Targeted System Configuration Diagrams

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

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

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

Network Configuration

The network configurations of both the tested services and the proposed (target) services which are being represented and a thorough explanation of exactly which parts of the proposed configuration are being replaced with the Driver System must be disclosed (see Clause 6.6.4). (8.1.7.4)

The client-to-server LAN connections used 100Mbps switched Ethernet for both the measured and priced configurations. The server-to-server connections used one Gigaset cLAN-5300 switch for both the measured and priced configurations. The server-to-server protocol was VIA.

In the measured configuration, the six LAN segment connections between most of the RTE-client pairs used Ethernet cross-over cables. The link speed for each Ethernet adapter port was set at 10Mbps. On one RTE-client pair, the six LAN segment connections used Type 5 Ethernet cables and 10Mbps Ethernet hubs. The priced configuration included 10Mbps hubs. Benchcraft was used to generate separate transaction reports for an RTE-client pair that used cross-over cables and an RTE-client pair that used Ethernet hubs. The response time data indicated no difference in performance between the use of hubs vs. cross-over cables. The data was submitted to and reviewed by the auditor.

Network Bandwidth

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

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

Operator Intervention

If the configuration requires operator intervention (see Clause 6.6.6), the mechanism and the frequency of this intervention must be disclosed. (8.1.7.6)

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

Clause 7: Pricing Related Items

Hardware and Software Components

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

Pricing source(s) and effective date(s) must also be reported. (8.1.8.1)

The total 5-year price of the entire configuration must be reported, including: hardware, software, and maintenance charges. Separate component pricing is recommended. The basis of all discounts used must be disclosed. (8.1.8.2)

A detailed list of all hardware and software, including the 5-year price, is provided in the Executive Summary at the front of this report. All third-party quotations are included in Appendix E at the end of this document.

Availability Date

The committed delivery date for general availability (availability date) of products used in the price calculations must be reported. When the priced system includes products with different availability dates, the reported availability for the priced system must be the date at which all components are committed to be available. (8.1.8.3)

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

Measured tpmC

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

- Maximum Qualified Throughput: 363,129.75 tpmC
- Price per tpmC: \$28.10 per tpmC
- Five-year cost of ownership: \$10,204,816

Country-Specific Pricing

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

The configuration is priced for the United States of America.

Usage Pricing

For any usage pricing, the sponsor must disclose:

- Usage level at which the component was priced.
- A statement of the company policy allowing such pricing. (8.1.8.6)

The component pricing based on usage is shown below:

- 16 copies of Microsoft Windows 2000 Datacenter Server
- 1 copies of Microsoft Windows 2000 Advanced Server
- 48 copies of Microsoft Windows 2000 Server

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

System Pricing

System pricing should include subtotals for the following components: Server Hardware, Server Software, Client Hardware, Client Software, and Network Components used for terminal connection (see Clause 7.2.2.3). (8.1.8.7)
System pricing must include line item indication where non-sponsoring companies' brands are used. System pricing must also include line item indication of third-party pricing. (8.1.8.8)

A detailed list of all hardware and software, including the 5-year price, is provided in the Executive Summary at the front of this report. All third-party quotations are included in Appendix E at the end of this document.

Clause 9: Audit Related Items

Auditor

The auditor's name, address, phone number, and a copy of the auditor's attestation letter indicating compliance must be included in the Full Disclosure Report. (8.1.9.1)

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

Availability of the Full Disclosure Report

The Full Disclosure Report must be readily available to the public at a reasonable charge, similar to the charges for similar documents by the test sponsor. The report must be made available when results are made public. In order to use the phrase "TPC BenchmarkTMC," the Full Disclosure Report must have been submitted to the TPC Administrator as well as written permission obtained to distribute same. (8.2)

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

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

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

March 21, 2001

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

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

The results were:

CPU's Speed	Memory	Disks(total)	NewOrder 90% Response Time	tpmC
16 Servers: IBM @server xSeries 370 each with:				
8 x Pentium III Xeon (900 MHz)	16 GB Main (2MB L2 Cache per processor)	218 x 18.2 GB 2 x 9.1 GB	0.31Seconds	363,129.75
48 Clients: IBM Netfinity 5100 each with:				
2 x Pentium III (733 MHz)	512 MB Main (256KB L2 Cache per processor)	1 x 9.1 GB	n/a	
1 DTC Server: IBM @server xSeries 370 with:				
8 x Pentium III Xeon (900 MHz)	512 MB Main (2MB L2 Cache per processor)	6 x 9.1 GB	n/a	

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


- The database records were the proper size
- The database was properly scaled and populated
- The required ACID properties were met
- The transactions were correctly implemented
- Input data was generated according to the specified percentages
- The transaction cycle times included the required keying and think times
- The reported response times were correctly measured.
- All 90% response times were under the specified maximums
- At least 90% of all delivery transactions met the 80 Second completion time limit
- The reported measurement interval was 20 minutes (1200 seconds)
- The reported measurement interval was representative of steady state conditions
- One checkpoint was taken during the reported measurement interval
- The repeatability of the measured performance was verified
- The 180 day storage requirement was correctly computed
- The ability to update primary keys with the same syntax as other columns was verified.
- The system pricing was verified for major components and maintenance

Additional Audit Notes:

The measured system included 3450 18.2 GB 10K UltraWide SCSI Drives P/N 36L9749 and 38 18.2 GB 10K Wide Ultra160 SCSI Drives P/N 37L7205 that were substituted by 3488 18.2 GB 10K Wide Ultra160 SCSI Drives P/N 37L7205 in the priced configuration. Based on the specifications of these disks and on additional performance data collected on these disks, it is my opinion that this substitution does not have a material effect on the reported performance.

In the measured system, 47 out of 48 clients systems were driven using a direct LAN connection between the RTE and the client systems, while the 48th was driven using the priced hubs. Based on measurements comparing the direct connect to the hubs, it is my opinion that there is no effect on performance by using the direct connect instead of the priced hubs.

Respectfully Yours,



François Raab, President



Bradley J. Askins, Auditor

Appendix A: Source Code

Web Client Source Code

dllldata.c

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

DO NOT ALTER THIS FILE

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

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

*****

#include <rpcproxy.h>

#ifdef __cplusplus
extern "C" {
#endif

EXTERN_PROXY_FILE( tpcc_com_remote_ps )

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

DLLDATA_ROUTINES(aProxyFileList, GET_DLL_CLSID)

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

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

error.h

```
/*
FILE:          ERROR.H
*              Microsoft TPC-C Kit Ver.
4.20.000
*              Copyright Microsoft,
1999
*              All Rights Reserved
*
*              Version 4.10.000 audited
by Richard Gimarc, Performance Metrics, 3/17/99
*
*              PURPOSE:      Header file for error exception
classes.
*
*              Change history:
*              4.20.000 - updated rev number to match kit
```

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

#pragma once

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

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

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

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

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



```

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

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

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

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

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

```

```

        else
            m_szLoc = NULL;

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

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

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

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

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

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

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

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

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

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

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

    int m_idMsg;

```

```

};

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

    CSocketErr(Action eAction, LPCTSTR szLocation =
NULL);

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

    Action    m_eAction;
    char     *m_szErrorText;

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

};

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

```

```

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

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

    Action    m_eAction;

private:
    char m_szMsg[ERR_MSG_BUF_SIZE];
};

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

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

```

install.c

```

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

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

#include "resource.h"

#define WM_INITTEXT WM_USER+100

HICON hIcon;

```

```

HINSTANCE      hInst;

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

// TPC-C registry settings
TPCCREGISTRYDATA  Reg;

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

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

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

BOOL install_com(char *szDllPath);

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

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

    hInst = hInstance;

    InitCommonControls();

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

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

```

```

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

        DestroyIcon(hIcon);
        return 0;
    }

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

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

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

            case WM_DESTROY:
                DeleteObject(hFont);
                return TRUE;

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

```

<pre> default: break; } return FALSE; } } BOOL CALLBACK UpdatedDlgProc(HWND hwnd, UINT uMsg, WPARAM wParam, LPARAM lParam) { switch(uMsg) { case WM_INITDIALOG: switch(lParam) { case 1: case 2: } SetDlgItemText(hwnd, IDC_RESULTS, "TPC-C Web Client Installed"); break; } return TRUE; case WM_COMMAND: if (wParam == IDOK) EndDialog(hwnd, TRUE); break; default: break; } return FALSE; } } BOOL CALLBACK MainDlgProc(HWND hwnd, UINT uMsg, WPARAM wParam, LPARAM lParam) { PAINTSTRUCT ps; MEMORYSTATUS memoryStatus; OSVERSIONINFO VI; char szTmp[256]; static char szDllPath[256]; static char szExePath[256]; switch(uMsg) { case WM_INITDIALOG: GlobalMemoryStatus(&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)); ReadTPCCRegistrySettings(&Reg); ReadRegistrySettings(); Reg.dwNumberOfDeliveryThreads= 7; Reg.dwMaxConnections = 10000; </pre>	<pre> 1000; "tpcc"); "sa"); strcpy(Reg.szDbUser, strcpy(Reg.szDbPassword, ""); // iMaxPhysicalMemory * 2; iPoolThreadLimit = iPoolThreadLimit = 190; iThreadTimeout = 86400; iListenBackLog = 150; iAcceptExOutstanding = 40; GetModuleFileName(hInst, szExePath, sizeof(szExePath)); GetVersionInfo(szDllPath, szExePath); wsprintf(szTmp, "Version %d.%2.2d.%3.3d", versionExeMS, versionExeMM, versionExeLS); SetDlgItemText(hwnd, IDC_VERSION, szTmp); SetDlgItemText(hwnd, IDC_PATH, szDllPath); SetDlgItemText(hwnd, ED_DB_SERVER, Reg.szDbServer); SetDlgItemText(hwnd, ED_DB_USER_ID, Reg.szDbUser); SetDlgItemText(hwnd, ED_DB_PASSWORD, Reg.szDbPassword); SetDlgItemText(hwnd, ED_DB_NAME, Reg.szDbName); SetDlgItemInt(hwnd, ED_THREADS, Reg.dwNumberOfDeliveryThreads, FALSE); SetDlgItemInt(hwnd, ED_MAXCONNECTION, Reg.dwMaxConnections, FALSE); SetDlgItemInt(hwnd, ED_MAXDELIVERIES, Reg.dwMaxPendingDeliveries, FALSE); SetDlgItemInt(hwnd, ED_IIS_MAX_THREAD_POOL_LIMIT, iPoolThreadLimit, FALSE); SetDlgItemInt(hwnd, ED_IIS_THREAD_TIMEOUT, iThreadTimeout, FALSE); SetDlgItemInt(hwnd, ED_IIS_LISTEN_BACKLOG, iListenBackLog, FALSE); SetDlgItemInt(hwnd, ED_WEB_SERVICE_BACKLOG_QUEUE_SIZE, iAcceptExOutstanding, FALSE); CheckDlgButton(hwnd, IDC_DBLIB, 0); CheckDlgButton(hwnd, IDC_ODBC, 0); if (Reg.eDB_Protocol == DBLIB) CheckDlgButton(hwnd, IDC_DBLIB, 1); else if (Reg.eDB_Protocol == ODBC) </pre>
---	--

<pre> IDC_ODBC, 1); CheckDlgButton(hwnd, // check OS version level for COM. Must be at least Windows 2000 VI.dwOSVersionInfoSize = sizeof(VI); GetVersionEx(&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; } IDC_TM_NONE, 0); IDC_TM_TUXEDO, 0); IDC_TM_MTS, 0); IDC_TM_ENCINA, 0); switch (Reg.eTxnMon) { case None: CheckDlgButton(hwnd, IDC_TM_NONE, 1); break; case TUXEDO: CheckDlgButton(hwnd, IDC_TM_TUXEDO, 1); break; case ENCINA: CheckDlgButton(hwnd, IDC_TM_ENCINA, 1); break; case COM: CheckDlgButton(hwnd, IDC_TM_MTS, 1); break; } return TRUE; case WM_PAINT: if (!IsIconic(hwnd)) { BeginPaint(hwnd, &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: </pre>	<pre> return TRUE; case IDC_ODBC: return TRUE; case IDOK: ProcessOK(hwnd, szDllPath); return TRUE; case IDC_CANCEL: EndDialog(hwnd, FALSE); return TRUE; default: return FALSE; } } default: break; } } return FALSE; } static void ProcessOK(HWND hwnd, char *szDllPath) { int d; HWND hDlg; int rc; char szFullName[256]; char szErrTxt[128]; // read settings from dialog Reg.dwNumberOfDeliveryThreads = GetDlgItemInt(hwnd, ED_THREADS, &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)) </pre>
--	--

```

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

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

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

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

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

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

```

```

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

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

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

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

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

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

```

```

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

        RegCloseKey(hKey);
    }

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

        RegCloseKey(hKey);
    }
}

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

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

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

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

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

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

```

```

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

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

        RegFlushKey(hKey);
        RegCloseKey(hKey);
    }

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

        RegFlushKey(hKey);
        RegCloseKey(hKey);
    }

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

        RegFlushKey(hKey);
        RegCloseKey(hKey);
    }

    return;
}

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

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

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

```

```

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

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

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

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

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

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

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

    CloseHandle(hFile);

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

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

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

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

```

```

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

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

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

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

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

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

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

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

```



```

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

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

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

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

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

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

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

return 1;
}

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

```

```

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

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

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

            RegCloseKey(hKey);
        }
    }

    return bRc;
}

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

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

```

```

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

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

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

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

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

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

    CloseServiceHandle(schService);
    return TRUE;

ServiceNotRunning:
    CloseServiceHandle(schService);
    return FALSE;
}

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

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

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

```

```

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

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

    CloseServiceHandle(schService);
    return TRUE;

StartWWWebErr:
    CloseServiceHandle(schService);
    return FALSE;
}

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

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

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

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

```

```

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

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

    CloseServiceHandle(schService);
    return TRUE;

StopWWWebErr:
    CloseServiceHandle(schService);
    return FALSE;
}

static void UpdateDialog(HWND hDlg)
{
    MSG msg;

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

```

install.h

```

//{{NO_DEPENDENCIES}}
// Microsoft Developer Studio generated include file.
// Used by install.rc
//

#define IDD_DIALOG1          101
#define IDI_ICON1           102
#define IDR_TPCCDLL         103
#define IDD_DIALOG2        105
#define IDI_ICON2           106
#define IDR_DELIVERY        107
#define IDD_DIALOG3        108

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

```

```

#define ED_USER_CONNECT_DELAY_TIME
1024

// Next default values for new objects
//

install.rc

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

#define APSTUDIO_READONLY_SYMBOLS
//
// Generated from the TEXTINCLUDE 2 resource.
//
#include "afxres.h"

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

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

//
// Dialog
//

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

```

```

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

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

```

```

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

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

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

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

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

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

```

```

END

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

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

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

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

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

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

#endif // APSTUDIO_INVOKED

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

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

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

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

```

```

#endif

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

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

#endif // !_MAC

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

IDR_LICENSE1      LICENSE DISCARDABLE  "license.txt"

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

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

```

```

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

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

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

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

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

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

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

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

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

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

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

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

```

```

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

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

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

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

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

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

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

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

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

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

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

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

install_com.cpp

/*      FILE:          INSTALL_COM.CPP
 *
 *      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,
bstrTemp5, bstrTemp6, bstrTemp7;
    _bstr_t                bstrDllPath =
szDllPath;
    _variant_t             vTmp, vKey;
    long                   lActProp,
ICount, ICountCo, ICountItf, ICountMethod;
    bool                   bTmp;
    bool
bRemote=FALSE;

    CoInitializeEx(NULL, COINIT_MULTITHREADED);

    HRESULT hr =
CoCreateInstance(CLSID_COMAdminCatalog,

        NULL,

        CLSCTX_INPROC_SERVER,

        IID_ICOMAdminCatalog,

        (void**) &pCOMAdminCat);

    if (!SUCCEEDED(hr)) goto Error;

```

```

bstrTemp = "Applications";

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

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

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

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

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

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

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

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

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

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

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

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

```

```

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

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

pCatalogObjectApp->Release();
pCatalogObjectApp = NULL;

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

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

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

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

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

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

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

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

```

```

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

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

if (!SUCCEEDED(hr)) goto Error;

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

if (!SUCCEEDED(hr)) goto Error;

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

if (!SUCCEEDED(hr)) goto Error;

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

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

if (!SUCCEEDED(hr)) goto Error;

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

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

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

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

```



```

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

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

        if (!SUCCEEDED(hr)) goto Error;

        bstrTemp = "EventTrackingEnabled";
        vTmp.Clear();
        bTmp = FALSE;
        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;

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

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

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

```

```

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

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

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

            bTmp = TRUE;
            vTmp = bTmp;
            hr =
pCatalogObjectMethod->put_Value(bstrTemp, vTmp);
            if (!SUCCEEDED(hr))
                goto Error;
        }

        pCatalogObjectMethod->Release();
        pCatalogObjectMethod =
NULL;

        ICountMethod--;
    }

    // save changes
    hr =
pCatalogCollectionMethod->SaveChanges(&lActProp);
    if (!SUCCEEDED(hr)) goto Error;

    pCatalogObjectItf->Release();
    pCatalogObjectItf = NULL;

    ICountItf--;
}

pCatalogObjectCo->Release();
pCatalogObjectCo = NULL;

ICountCo--;
}

// save changes
hr = pCatalogCollectionCo->SaveChanges(&lActProp);
if (!SUCCEEDED(hr)) goto Error;

pCatalogCollectionApp->Release();
pCatalogCollectionApp = NULL;

pCatalogCollectionCo->Release();
pCatalogCollectionCo = NULL;

pCatalogCollectionItf->Release();
pCatalogCollectionItf = NULL;

```

```

    pCatalogCollectionMethod->Release();
    pCatalogCollectionMethod = NULL;

Error:
    CoUninitialize();

    if (!SUCCEEDED(hr))
    {
        LPTSTR lpBuf;
        DWORD dwRes =
FormatMessage(FORMAT_MESSAGE_ALLOCATE_BUFFER|
FORMAT_MESSAGE_FROM_SYSTEM,

        NULL,

        hr,

        MAKELANGID(LANG_NEUTRAL,
        SUBLANG_DEFAULT),

        (LPTSTR) &lpBuf,

        0,

        NULL);
//
        _tprintf(_T("Error adding components.
HRESULT: 0x%x\n%s"), hr, lpBuf);
        return TRUE;
    }
    else
        return FALSE;
}

```

install_resource.h

```

//{{NO_DEPENDENCIES}}
// Microsoft Developer Studio generated include file.
// Used by install.rc
//
#define IDD_DIALOG1 101
#define IDI_ICON1 102
#define IDR_TPCCDLL 103
#define IDD_DIALOG2 105
#define IDI_ICON2 106
#define IDR_DELIVERY 107
#define IDD_DIALOG3 108
#define IDR_LICENSE1 112
#define IDD_DIALOG4 113
#define IDR_TPCCOBJ1 117
#define IDR_TPCCSTUB1 118
#define IDR_DBLIB_DLL 122
#define IDR_ODBC_DLL 123
#define IDR_TUXEDO_APP 124
#define IDR_TUXEDO_DLL 125
#define IDR_COM_DLL 126
#define IDR_COMPS_DLL 127
#define IDR_COMALL_DLL 128
#define IDR_COMTYPLIB_DLL 129
#define IDR_REMOTE_DLL 131
#define IDR_COMREMTYPLIB_DLL 132
#define IDR_COMREMPS_DLL 133
#define BN_LOG 1001
#define ED_KEEP 1002
#define ED_THREADS 1003
#define ED_THREADS2 1004

```

```

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

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

```

isapi_dll_resource.h

```

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

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

```

license.txt

```

END-USER LICENSE AGREEMENT FOR
MICROSOFT TPC-C BENCHMARK KIT

IMPORTANT READ CAREFULLY: This Microsoft End-

```

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

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

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

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

2. RESTRICTIONS.

--You must maintain all copyright notices on all copies of the SOFTWARE PRODUCT.

--You may not distribute copies of the SOFTWARE PRODUCT to third parties.

--You may not rent, lease or lend the SOFTWARE PRODUCT.

--You may not use the SOFTWARE PRODUCT or any derivative works thereof to internally test database management system software other than Microsoft SQL Server and/or operating system software other than Microsoft Windows NT.

-- You may not disclose the results of any benchmark tests using the SOFTWARE PRODUCT to any third party without Microsoft's prior written approval.

-- You may not disclose or provide the SOFTWARE PRODUCT or any derivative works thereof, or any information relating to the SOFTWARE PRODUCT (including the existence of the SOFTWARE PRODUCT or the results of use and testing or benchmark testing), to any third party without Microsoft's written permission.

3. TERMINATION. Without prejudice to any other rights, Microsoft may terminate this EULA if you fail to comply with the terms and conditions of this EULA. In such event, you must destroy all copies of the SOFTWARE PRODUCT.

4. COPYRIGHT. All title and copyrights in and to the SOFTWARE PRODUCT and any copies thereof are owned by Microsoft or its suppliers. All title and intellectual property rights in and to the content which may be accessed through use of the SOFTWARE PRODUCT is the property of the respective content owner and may be protected by applicable copyright or other intellectual property laws and treaties. This EULA grants you no rights to use such content.

5. UPGRADES. If the SOFTWARE PRODUCT is labeled as an upgrade, you must be properly licensed to use a product identified by Microsoft as being eligible for the upgrade in order to use the SOFTWARE PRODUCT. A SOFTWARE PRODUCT labeled as an upgrade replaces and/or supplements the product that formed the basis for your eligibility for the upgrade. You may use the resulting upgraded product only in

accordance with the terms of this EULA.

6. U.S. GOVERNMENT RESTRICTED RIGHTS.

The SOFTWARE PRODUCT is provided with RESTRICTED RIGHTS. Use, duplication, or disclosure by the Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 or subparagraphs (c)(1) and (2) of the Commercial Computer Software Restricted Rights at 48 CFR 52.227-19, as applicable. Manufacturer is Microsoft Corporation/One Microsoft Way/Redmond, WA 98052-6399.

7. EXPORT RESTRICTIONS.

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

8. NO WARRANTY. ANY USE OF THE SOFTWARE PRODUCT IS AT YOUR OWN RISK. THE SOFTWARE PRODUCT IS PROVIDED FOR USE ONLY WITH MICROSOFT SQL SERVER AND/OR MICROSOFT WINDOWS NT SERVER SOFTWARE. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, MICROSOFT AND ITS SUPPLIERS DISCLAIM ALL WARRANTIES AND CONDITIONS, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT.

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

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

11. MISCELLANEOUS

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

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

Way/Redmond, WA 98052-6399.

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

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

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

ABSENCE DE RESPONSABILITÉ POUR LES DOMMAGES INDIRECTS.

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

methods.h

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

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

```
ERR_LOADDLL_FAILED,
ERR_GETPROCADDR_FAILED,
ERR_UNKNOWN_DB_PROTOCOL
};

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

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

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

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

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

static void WriteMessageToEventLog(LPTSTR lpszMsg);

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

```

        CTPCC_Common();
        ~CTPCC_Common();

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

        HRESULT __stdcall CallSetComplete();

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

// IObjectConstruct
    STDMETHODIMP Construct(IDispatch * pUnk);

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

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

};

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

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

```

```

        COM_INTERFACE_ENTRY_CHAIN(CTPCC_Common)
    END_COM_MAP()
};

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

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

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

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

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

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

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

```

```

{
public:
DECLARE_REGISTRY_RESOURCEID(IDR_PAYMENT)

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

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

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

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

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

```

ReadRegistry.cpp

```

/* FILE:          READREGISTRY.CPP
 *               Microsoft TPC-C Kit Ver.
4.20.000
 *               Copyright Microsoft,
1999
 *               All Rights Reserved
 *
 *               not yet audited
 *
 * PURPOSE:      Implementation for TPC-C Tuxedo
class.
 * Contact:     Charles Levine (clevine@microsoft.com)
 *
 * Change history:

```

```

 *               4.20.000 - first version
 */

/* FUNCTION: ReadTPCCRegistrySettings
 *
 * PURPOSE:      This function reads the NT registry for startup
 *               parameters. There parameters are
 *               under the TPCC key.
 *
 * RETURNS      FALSE = no errors
 *               TRUE  = error reading registry
 */
BOOL ReadTPCCRegistrySettings( TPCCREGISTRYDATA *pReg )
{
    HKEY   hKey;
    DWORD  size;
    DWORD  type;
    DWORD  dwTmp;
    char    szTmp[256];

    if ( RegOpenKeyEx(HKEY_LOCAL_MACHINE,
    "SOFTWARE\\Microsoft\\TPCC", 0, KEY_READ, &hKey) !=
    ERROR_SUCCESS )
        return TRUE;

    // determine database protocol to use; may be either ODBC
    or DBLIB
    pReg->eDB_Protocol = Unspecified;
    size = sizeof(szTmp);
    if ( RegQueryValueEx(hKey, "DB_Protocol", 0, &type,
    (BYTE *)&szTmp, &size) == ERROR_SUCCESS )
    {
        if ( !strcmp(szTmp, szDBNames[ODBC]) )
            pReg->eDB_Protocol = ODBC;
        else if ( !strcmp(szTmp, szDBNames[DBLIB]) )
            pReg->eDB_Protocol = DBLIB;
    }

    pReg->eTxnMon = None;
    // determine txn monitor to use; may be either TUXEDO, or
    blank
    size = sizeof(szTmp);
    if ( RegQueryValueEx(hKey, "TxnMonitor", 0, &type,
    (BYTE *)&szTmp, &size) == ERROR_SUCCESS )
    {
        if ( !strcmp(szTmp,
    szTxnMonNames[TUXEDO]) )
            pReg->eTxnMon = TUXEDO;
        else if ( !strcmp(szTmp,
    szTxnMonNames[ENCINA]) )
            pReg->eTxnMon = ENCINA;
        else if ( !strcmp(szTmp,
    szTxnMonNames[COM]) )
            pReg->eTxnMon = COM;
    }

    pReg->bCOM_SinglePool = FALSE;
    size = sizeof(szTmp);
    if ( RegQueryValueEx(hKey, "COM_SinglePool", 0, &type,
    (BYTE *)&szTmp, &size) == ERROR_SUCCESS )
    {
        if ( !strcmp(szTmp, "YES") )
            pReg->bCOM_SinglePool = TRUE;
    }

    pReg->dwMaxConnections = 0;
    size = sizeof(dwTmp);

```

```

    if ( ( RegQueryValueEx(hKey, "MaxConnections", 0,
&type, (LPBYTE)&dwTmp, &size) == ERROR_SUCCESS )
        && (type == REG_DWORD) )
        pReg->dwMaxConnections = dwTmp;

    pReg->dwMaxPendingDeliveries = 0;
    size = sizeof(dwTmp);
    if ( ( RegQueryValueEx(hKey, "MaxPendingDeliveries", 0,
&type, (LPBYTE)&dwTmp, &size) == ERROR_SUCCESS )
        && (type == REG_DWORD) )
        pReg->dwMaxPendingDeliveries = dwTmp;

    pReg->dwNumberOfDeliveryThreads = 0;
    size = sizeof(dwTmp);
    if ( ( RegQueryValueEx(hKey,
"NumberOfDeliveryThreads", 0, &type, (LPBYTE)&dwTmp, &size)
== ERROR_SUCCESS )
        && (type == REG_DWORD) )
        pReg->dwNumberOfDeliveryThreads = dwTmp;

    size = sizeof( pReg->szPath );
    if ( RegQueryValueEx(hKey, "Path", 0, &type, (BYTE
*)&pReg->szPath, &size) != ERROR_SUCCESS )
        pReg->szPath[0] = 0;

    size = sizeof( pReg->szDbServer );
    if ( RegQueryValueEx(hKey, "DbServer", 0, &type, (BYTE
*)&pReg->szDbServer, &size) != ERROR_SUCCESS )
        pReg->szDbServer[0] = 0;

    size = sizeof( pReg->szDbName );
    if ( RegQueryValueEx(hKey, "DbName", 0, &type, (BYTE
*)&pReg->szDbName, &size) != ERROR_SUCCESS )
        pReg->szDbName[0] = 0;

    size = sizeof( pReg->szDbUser );
    if ( RegQueryValueEx(hKey, "DbUser", 0, &type, (BYTE
*)&pReg->szDbUser, &size) != ERROR_SUCCESS )
        pReg->szDbUser[0] = 0;

    size = sizeof( pReg->szDbPassword );
    if ( RegQueryValueEx(hKey, "DbPassword", 0, &type,
(BYTE *)&pReg->szDbPassword, &size) != ERROR_SUCCESS )
        pReg->szDbPassword[0] = 0;

    RegCloseKey(hKey);

    return FALSE;
}

```

ReadRegistry.h

```

/*      FILE:          ReadRegistry.h
*
4.22.000          Microsoft TPC-C Kit Ver.
*
*                  Copyright Microsoft,
2000
*
*                  All Rights Reserved
*
*                  not audited
*
*      PURPOSE:      Header for registry related code.
*
*      Change history:
*
*                  4.20.000 - first version
*/

```

```

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

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

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

BOOL ReadTPCCRegistrySettings( TPCCREGISTRYDATA *pReg );

```

ReadWHouse.cpp

```

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

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

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

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

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

// forward decl of routine

```

```

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

// begin code

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

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

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

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

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

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

```

```

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

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

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

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

}

// find out how large the array needs to be

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

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

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

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

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

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

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

```



```

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

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

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

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

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

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

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

if (piTotalWhs)
    *piTotalWhs = iTotalWhs;

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

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

// select the data and copy to the array
rc = SQLAllocStmt(hdbc, &hstmt);
if (!SQLSUCCEEDED(rc))
{
    free(pArray);
    *ppWarehouseArray = NULL;
    return ProcessError(henv,hdbc,hstmt);
}

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

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

```

```

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

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

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

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

*piCount = ii;

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

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

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

// cleanup

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

```

```

        return (ERROR_SUCCESS);
    }

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

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

        // quit if there isn't enough room to concatenate
error text
        if ( ( strlen(szMsg) + 2) > (sizeof(szTmp) -
strlen(szTmp)) )
            break;

        // include line break after first error msg
        if (szTmp[0] != 0)
            strcat( szTmp, "\n");
        strcat( szTmp, szMsg );
    }

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

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

    return (E_FAIL);
}

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

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

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

    if (hEventSource != NULL)
    {

```

```

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

        (VOID) DeregisterEventSource(hEventSource);
    }
}

```

ReadWHouse.h

```

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

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

```

rtetime.h

```

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

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

```

```

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

```

SetAuditTime.cpp

```

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

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

#include <stdlib.h> // db2

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

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

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

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

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

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

```

```

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

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

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

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

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

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

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

//
// Connect to db
//
{

```

```

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

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

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

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

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

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

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

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

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

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

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

}

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

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

// cleanup

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

return (ERROR_SUCCESS);
}

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

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

// quit if there isn't enough room to concatenate
error text
if ( ( strlen(szMsg) + 2) > (sizeof(szTmp) -
strlen(szTmp)) )
break;

// include line break after first error msg
if (szTmp[0] != 0)
strcat( szTmp, "\n");
strcat( szTmp, szMsg );
}

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

```

```

        WriteMessageToEventLog(szTmp);
    }

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

    return (E_FAIL);
}

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

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

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

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

        (VOID) DeregisterEventSource(hEventSource);
    }
}

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

    SYSTEMTIME sysTime;

    comboDate.i64Date=iTime;

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

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

    return;
}

```

```

}

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

    SYSTEMTIME sysTime;

    GetLocalTime(&sysTime);

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

```

spinlock.h

```

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

#ifndef _INC_Spinlock

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

    /**
     *
     * Spinlock and Semaphore locking.
     *
     * This class provides a very conservative locking scheme.
     *
     * The assumption behind the code is that locks will be
     * held for a very short time. When a lock is taken a
     * memory
     * location is exchanged. All other threads that want this
     * semaphore
     * lock wait by spinning and sometimes sleeping on a
     * cache
     * 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
     * aligned memory in minimize cache line misses.
     *
     */

    /**
     */

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

```

```

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

public:
    // Public functions.

    Spinlock( void );

    inline BOOL ClaimLock( BOOL
Wait = TRUE );

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

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

};

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

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

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

```

```

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

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

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

#define _INC_Spinlock

#endif

tpcc.cpp

/*      FILE:          TPCC.C
 *
 *      Microsoft TPC-C Kit Ver.
4.20.000
 *
 *      Copyright Microsoft,
1999
 *
 *      All Rights Reserved
 *
 *
 *
 *
 *      Version 4.10.000 audited
by Richard Gimarc, Performance Metrics, 3/17/99
 *
 *
 *      PURPOSE:          Main module for TPCC.DLL which
is an ISAPI service dll.
 *
 *      Contact:   Charles Levine (clevine@microsoft.com)
 *
 *
 *      Change history:
 *
 *      4.20.000 - reworked error handling; added
options for COM and Encina txn monitors
 */

#include <windows.h>
#include <process.h>
#include <tchar.h>
#include <stdio.h>
#include <stdarg.h>
#include <malloc.h>
#include <stdlib.h>
#include <string.h>
#include <time.h>
#include <sys\timeb.h>
#include <io.h>
#include <assert.h>

```

```

#include <sqltypes.h>

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

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

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

// Database layer includes
#include "..\..\db_dblib_dll\src\tpcc_dblib.h" //
DBLIB implementation of TPC-C txns
#include "..\..\db_odbc_dll\src\tpcc_odbc.h" // ODBC
implementation of TPC-C txns

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

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

#define LEN_ERR_STRING 256

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

char
szMyComputerName[MAX_COMPUTERNAME_LENGTH+1];

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

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

static CRITICAL_SECTION
TermCriticalSection;

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

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

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

// For deferred Delivery txns:

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

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

// configuration settings from registry
TPCCREGISTRYDATA Reg;

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

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

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

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

```

```

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

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

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

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

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

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

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

```

```

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

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

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

// get function
pointer to wrapper for class constructor

```



```

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

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

GetWareHouseTable(
                (void
***)&pWareHouseArray,// array of ints
                &iCount,
                &iTotalWhs,

Reg.szDbUser,          // user name for login

Reg.szDbPassword,    // password for login

Reg.szDbName);       // name of DSN to use

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

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

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

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

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

```

```

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

//
get function pointer to wrapper for class constructor

pCTPCC_DBLIB_new = (TYPE_CTPCC_DBLIB*)
GetProcAddress(hLibInstanceDb,"CTPCC_DBLIB_new");
if
(pCTPCC_DBLIB_new == NULL)
throw new CWEBCLNT_ERR( ERR_GETPROCADDR_FAILED,
szDllName, GetLastError());
}
else if
(Reg.eDB_Protocol == ODBC)
{
strcpy( szDllName, Reg.szPath );
strcat( szDllName, "tpcc_odbc.dll");
hLibInstanceDb = LoadLibrary( szDllName );
if
(hLibInstanceDb == NULL)
throw new CWEBCLNT_ERR( ERR_LOADDLL_FAILED,
szDllName, GetLastError());

//
get function pointer to wrapper for class constructor

pCTPCC_ODBC_new = (TYPE_CTPCC_ODBC*)
GetProcAddress(hLibInstanceDb,"CTPCC_ODBC_new");
if
(pCTPCC_ODBC_new == NULL)
throw new CWEBCLNT_ERR( ERR_GETPROCADDR_FAILED,
szDllName, GetLastError());
}
}
if
(dwNumDeliveryThreads)
{
// for deferred
delivery txns:
hDoneEvent =
CreateEvent( NULL, TRUE /* manual reset */, FALSE /* initially not
signalled */, NULL );
InitializeCriticalSection(&DelBuffCriticalSection);
hWorkerSemaphore = CreateSemaphore( NULL, 0, dwDelBuffSize,
NULL );
dwDelBuffFreeCount = dwDelBuffSize;

InitJulianTime(NULL);

// create unique
log file name based on delilog-yyymmdd-hhmm.log

```

SYSTEMTIME Time;		pDelBuff;	delete []
&Time);	GetLocalTime(hWorkerSemaphore);	CloseHandle(
szLogFile, "%sdelivery-%2.2d%2.2d%2.2d-%2.2d%2.2d.log",	wsprintf(hDoneEvent);	CloseHandle(
Reg.szPath, Time.wYear % 100, Time.wMonth, Time.wDay,		DeleteCriticalSection(&DelBuffCriticalSection);	
Time.wHour, Time.wMinute);	txnDelilog =	}	
new CTxnLog(szLogFile, TXN_LOG_WRITE);		DeleteCriticalSection(&TermCriticalSection);	
	//write event		if (hLibInstanceTm !=
into txn log for START			FreeLibrary(
txnDelilog->WriteCtrlRecToLog(TXN_EVENT_START,		NULL)	hLibInstanceTm = NULL;
szMyComputerName, sizeof(szMyComputerName));		hLibInstanceTm);	
	// allocate	NULL)	if (hLibInstanceDb !=
structures for delivery buffers and thread mgmt	pDeliHandles	hLibInstanceDb);	FreeLibrary(
= new HANDLE[dwNumDeliveryThreads];	pDelBuff =	hLibInstanceDb = NULL;	
new DELIVERY_TRANSACTION[dwDelBuffSize];		Sleep(500);	break;
DeliveryWorkerThread to perform actual delivery txns	// launch	break;	
i<dwNumDeliveryThreads; i++)	for(i=0;	default:	/* nothing */;
	{	}	
pDeliHandles[i] = (HANDLE) _beginthread(DeliveryWorkerThread,		catch (CBaseErr *e)	
0, NULL);	if	{	WriteMessageToEventLog(e->ErrorText());
(pDeliHandles[i] == INVALID_HANDLE_VALUE)		{	delete e;
throw new CWEBCLNT_ERR({	TerminateExtension(0);
ERR_DELIVERY_THREAD_FAILED);		{	return FALSE;
	break;	catch (...)	
		{	WriteMessageToEventLog(TEXT("Unhandled
	case DLL_PROCESS_DETACH:	{	exception. DLL could not load."));
	if	{	TerminateExtension(0);
(dwNumDeliveryThreads)		{	return FALSE;
	{		
!= NULL)	if (txnDelilog		return TRUE;
	{		
//write event into txn log for STOP			
txnDelilog->WriteCtrlRecToLog(TXN_EVENT_STOP,		/* FUNCTION: GetExtensionVersion	
szMyComputerName, sizeof(szMyComputerName));		*	
	//	* PURPOSE: This function is called by the inet service when	
This will do a clean shutdown of the delivery log file		the DLL is first loaded.	
CTxnLog *txnDelilogLocal = txnDelilog;		*	
txnDelilog= NULL;		* ARGUMENTS: HSE_VERSION_INFO *pVer	
delete txnDelilogLocal;		passed in structure in which to place expected version number.	
		*	
		* RETURNS: TRUE inet service expected	
		return value.	
		*/	
	}	BOOL WINAPI GetExtensionVersion(HSE_VERSION_INFO *pVer)	
pDeliHandles;	delete []	{	
		{	pVer->dwExtensionVersion =
		{	MAKELONG(HSE_VERSION_MINOR, HSE_VERSION_MAJOR);

```

        Istrcpy(pVer->lpszExtensionDesc, "TPC-C Server.",
HSE_MAX_EXT_DLL_NAME_LEN);

// TODO: why do we need this here instead of in the DLL
attach?
if (Reg.eTxnMon == ENCINA)
    pCTPCC_ENCINA_post_init();

return TRUE;
}

/* FUNCTION: TerminateExtension
 *
 * PURPOSE: This function is called by the inet service when
the DLL is about to be unloaded.
 * Release all resources in anticipation
of being unloaded.
 *
 * RETURNS: TRUE inet service expected
return value.
 */

BOOL WINAPI TerminateExtension( DWORD dwFlags )
{
    if (pDeliHandles)
    {
        SetEvent( hDoneEvent );
        for(DWORD i=0; i<dwNumDeliveryThreads;
i++)
            WaitForSingleObject(
pDeliHandles[i], INFINITE );
    }

    TermDeleteAll();
    return TRUE;
}

/* FUNCTION: HttpExtensionProc
 *
 * PURPOSE: This function is the main entry point for the
TPCC DLL. The internet service
 * calls this function passing in the http
string.
 *
 * ARGUMENTS: EXTENSION_CONTROL_BLOCK
 * pECB structure pointer to passed in internet
service
information.
 *
 * RETURNS: DWORD
HSE_STATUS_SUCCESS
connection can be dropped if error
 *
HSE_STATUS_SUCCESS_AND_KEEP_CONN keep connect
valid comment sent
 *
 * COMMENTS: None
 */

DWORD WINAPI
HttpExtensionProc(EXTENSION_CONTROL_BLOCK *pECB)
{
    int iCmd, FormId, TermId,
iSyncId;
    char szBuffer[4096];

```

```

    int lpbSize;
    static char szHeader[] = "200 Ok";
    DWORD dwSize = 6; //
    initial value is strlen(szHeader)
    char szHeader1[4096];

#ifdef ICECAP
    StartCAP();
#endif

    try
    {
        //process http query
        ProcessQueryString(pECB, &iCmd, &FormId,
&TermId, &iSyncId);

        if (TermId != 0)
        {
            if ( TermId < 0 || TermId >=
Term.iNumEntries || Term.pClientData[TermId].iNextFree != -1 )
            {
                // debugging...
                char szTmp[128];
                wsprintf( szTmp, "Invalid
term ID; TermId = %d", TermId);
                WriteMessageToEventLog( szTmp );
                throw new
CWEBCLNT_ERR( ERR_INVALID_TERMID);
            }

            //must have a valid syncid here since
termid is valid
            if (iSyncId !=
Term.pClientData[TermId].iSyncId)
                throw new
CWEBCLNT_ERR( ERR_INVALID_SYNC_CONNECTION);

            //set use time
            Term.pClientData[TermId].iTickCount = GetTickCount();
        }

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

```

```

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

```

```

}
}

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

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

    DELIVERY_TRANSACTION    delivery;
    PDELIVERY_DATA
pDeliveryData;
    TXN_RECORD_TPCC_DELIV_DEF    txnDeliRec;

    DWORD
index;
    HANDLE
handles[2];

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

    assert(txnDeliLog != NULL);

    assert(txnDeliLog != NULL);

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

```

```

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

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

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

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

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

LeaveCriticalSection(&DelBuffCriticalSection);

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

```

```

        txnDeliRec.w_id =
pDeliveryData->w_id;
        txnDeliRec.o_carrier_id =
pDeliveryData->o_carrier_id;
        txnDeliRec.TxnStartT0 =
Get64BitTime(&delivery.queue);

        GetLocalTime(
&trans_start );
        pTxn->Delivery();
        GetLocalTime(
&trans_end );

        //log txn
        txnDeliRec.TxnStatus =
ERR_SUCCESS;
        for (int i=0; i<10; i++)

        txnDeliRec.o_id[i] = pDeliveryData->o_id[i];
        txnDeliRec.DeltaT4 =
(int)(Get64BitTime(&trans_end) - txnDeliRec.TxnStartT0);
        txnDeliRec.DeltaTxnExec
= (int)(Get64BitTime(&trans_end) - Get64BitTime(&trans_start));
        if (txnDelilog != NULL)

        txnDelilog->WriteToLog(&txnDeliRec);
    }
    catch (CBaseErr *e)
    {
        char szTmp[1024];
        wsprintf( szTmp, "Error in Delivery
Txn thread. %s", e->ErrorText() );
        WriteMessageToEventLog( szTmp );

        // log the error txn
        txnDeliRec.TxnStatus =
e->ErrorType();
        if (txnDelilog != NULL)

        txnDelilog->WriteToLog(&txnDeliRec);

        delete e;
    }
    catch (...)
    {
        // unhandled exception; shouldn't
        happen; not much we can do...

        WriteMessageToEventLog(TEXT("Unhandled exception caught in
        DeliveryWorkerThread. "));
    }
}

ErrorExit:
    delete pTxn;
    _endthread();
}

/* FUNCTION: PostDeliveryInfo
*
* PURPOSE:      This function enters the delivery txn into the
deferred delivery buffer.
*
* RETURNS:      BOOL    FALSE    delivery
information posted successfully
*/
TRUE    error cannot post delivery info
*/
BOOL PostDeliveryInfo(long w_id, short o_carrier_id)
{
    BOOL bError;

    EnterCriticalSection(&DelBuffCriticalSection);
    if (dwDelBuffFreeCount > 0)
    {
        bError = FALSE;
        (pDelBuff+dwDelBuffFreeIndex)->w_id
        = w_id;
        (pDelBuff+dwDelBuffFreeIndex)->o_carrier_id
        = o_carrier_id;
        GetLocalTime(&(pDelBuff+dwDelBuffFreeIndex)->queue);
        dwDelBuffFreeCount--;
        dwDelBuffFreeIndex++;
        if (dwDelBuffFreeIndex == dwDelBuffSize)
            dwDelBuffFreeIndex = 0;
        // wrap-around if at end of buffer
    }
    else
        // No free buffers. Return an error, which
        indicates that the delivery buffer is full.
        // Most likely, the number of delivery worker
        threads needs to be increased to keep up
        // with the txn rate.
        bError = TRUE;
        LeaveCriticalSection(&DelBuffCriticalSection);

        if (!bError)
            // increment worker semaphore to wake up a
            worker thread
            ReleaseSemaphore( hWorkerSemaphore, 1,
            NULL );

        return bError;
    }

/* FUNCTION: ProcessQueryString
*
* PURPOSE:      This function extracts the relevent information
out of the http command passed in from
*
*               the browser.
*
* COMMENTS:     If this is the initial connection i.e. client is at
welcome screen then
*
*               there will not be a
terminal id or current form id. If this is the case
*
*               then the pTermid and
pFormid return values are undefined.
*/
void ProcessQueryString(EXTENSION_CONTROL_BLOCK *pECB,
int *pCmd, int *pFormId, int *pTermId, int *pSyncId)
{
    char *ptr = pECB->lpszQueryString;
    char szBuffer[25];
    int i;

    //allowable client command strings i.e. CMD=command
    static char *szCmds[] =
    {

```

```

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

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

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

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

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

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

/* FUNCTION: void WelcomeForm
*
*/

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

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

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

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

"Compiled: __DATE__, __TIME__ <BR>"

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

"</PRE></font>"

"<FORM ACTION='tpcc.dll' METHOD='GET'>"

```

```

"<INPUT TYPE='hidden' NAME='STATUSID' VALUE='0'>"
"<INPUT TYPE='hidden' NAME='ERROR' VALUE='0'>"
"<INPUT TYPE='hidden' NAME='FORMID' VALUE='1'>"
"<INPUT TYPE='hidden' NAME='TERMIID' VALUE='0'>"
"<INPUT TYPE='hidden' NAME='SYNCID' VALUE='0'>"
"<INPUT TYPE='hidden' NAME='VERSION' VALUE=''"
WEBCLIENT_VERSION"'>"
    );

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

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

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

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

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

```

```

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

"<font face=\"Courier New\" color=\"blue\"><PRE>"

Server      = <B>%s</B><BR>"

User ID     = <B>%s</B><BR>"

Password   = <B>%s</B><BR>"

Name       = <B>%s</B><BR>"

"</PRE></font>"

, Reg.szDbServer,
Reg.szDbUser, Reg.szDbPassword, Reg.szDbName);
strcat( szBuffer, szTmp);

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

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

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

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

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

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

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

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

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

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

    if (Reg.eTxnMon == None)
    {
        // parse Server name

```

```

        GetKeyValue(&ptr, "db_server", szServer,
sizeof(szServer), ERR_NO_SERVER_SPECIFIED);
        // parse User name
        GetKeyValue(&ptr, "db_user", szUser,
sizeof(szUser), NO_ERR);
        // parse Password
        GetKeyValue(&ptr, "db_passwd", szPassword,
sizeof(szPassword), NO_ERR);
        // parse Database name
        GetKeyValue(&ptr, "db_name", szDatabase,
sizeof(szDatabase), NO_ERR);
    }

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

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

    iNewTerm = TermAdd();

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

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

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

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

```



```

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

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

    EnterCriticalSection(&TermCriticalSection);

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

    LeaveCriticalSection(&TermCriticalSection);

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

char *CWEBCLNT_ERR::ErrorText()
{
    static SERRORMSG errorMsgs[] =
    {
        { ERR_COMMAND_UNDEFINED,
          "Command undefined."
        },
        { ERR_D_ID_INVALID,
          "Invalid District ID Must
be 1 to 10."
        },
        { ERR_DELIVERY_CARRIER_ID_RANGE,
          "Delivery Carrier ID out of range must be 1 - 10."
        },
        { ERR_DELIVERY_CARRIER_INVALID,
          "Delivery Carrier ID invalid must be numeric 1 - 10."
        },
        { ERR_DELIVERY_MISSING_OCD_KEY,
          "Delivery missing Carrier ID key\"OCD*\"."
        },
        { ERR_DELIVERY_THREAD_FAILED,
          "Could not start delivery worker thread."
        },
        { ERR_GETPROCADDR_FAILED,
          "Could not map proc in
DLL. GetProcAddr error. DLL="
        },
        { ERR_HTML_ILL_FORMED,
          "Required key field is missing from
HTML string."
        },
        { ERR_INVALID_SYNC_CONNECTION,

```

```

"Invalid Terminal Sync ID."
    },
    { ERR_INVALID_TERMID,
      "Invalid Terminal ID."
    },
    { ERR_LOADDLL_FAILED,
      "Load of DLL failed.
DLL="
    },
    { ERR_MAX_CONNECTIONS_EXCEEDED,
      "No connections available. Max Connections is probably too low."
    },
    { ERR_MISSING_REGISTRY_ENTRIES,
      "Required registry entries are missing. Rerun INSTALL to correct."
    },
    { ERR_NEWORDER_CUSTOMER_INVALID,
      "New Order customer id invalid data type, range = 1 to 3000."
    },
    { ERR_NEWORDER_CUSTOMER_KEY,
      "New Order missing Customer key\"CID*\"."
    },
    { ERR_NEWORDER_DISTRICT_INVALID,
      "New Order District ID Invalid range 1 - 10."
    },
    { ERR_NEWORDER_FORM_MISSING_DID,
      "New Order missing District key\"DID*\"."
    },
    { ERR_NEWORDER_ITEMID_INVALID,
      "New Order Item Id is wrong data type, must be numeric."
    },
    { ERR_NEWORDER_ITEMID_RANGE,
      "New Order Item Id is out of range. Range = 1 to 999999."
    },
    { ERR_NEWORDER_ITEMID_WITHOUT_SUPPW,
      "New Order Item_Id field entered without a corresponding Supp_W."
    },
    { ERR_NEWORDER_MISSING_IID_KEY,
      "New Order missing Item Id key\"IID*\"."
    },
    { ERR_NEWORDER_MISSING_QTY_KEY,
      "New Order Missing Qty key\"Qty##*\"."
    },
    { ERR_NEWORDER_MISSING_SUPPW_KEY,
      "New Order missing Supp_W key\"SP##*\"."
    },
    { ERR_NEWORDER_NOITEMS_ENTERED,
      "New Order No order lines entered."
    },
    { ERR_NEWORDER_QTY_INVALID,
      "New Order Qty invalid must be numeric range 1 - 99."
    },
    { ERR_NEWORDER_QTY_RANGE,
      "New Order Qty is out of

```

```

range. Range = 1 to 99."
},
    {
ERR_NEWORDER_QTY_WITHOUT_SUPPW,
"New Order Qty field entered without a corresponding Supp_W."
},
    {
ERR_NEWORDER_SUPPW_INVALID,
"New Order Supp_W invalid data type must be numeric."
    },
    {
ERR_NO_SERVER_SPECIFIED,
"No Server name specified."
    },
},
{
ERR_ORDERSTATUS_CID_AND_CLT,
"Order Status Only Customer ID or Last Name may be entered, not
both." },
{
ERR_ORDERSTATUS_CID_INVALID,
"Order Status Customer ID invalid, range must be numeric 1 - 3000."
},
{
ERR_ORDERSTATUS_CLT_RANGE,
"Order Status Customer last name longer than 16 characters."
},
{
ERR_ORDERSTATUS_DID_INVALID,
"Order Status District invalid, value must be numeric 1 - 10."
},
{
ERR_ORDERSTATUS_MISSING_CID_CLT,
"Order Status Either Customer ID or Last Name must be entered."
},
{
ERR_ORDERSTATUS_MISSING_CID_KEY,
"Order Status missing Customer key \"CID*\"."
    },
{
ERR_ORDERSTATUS_MISSING_CLT_KEY,
"Order Status missing Customer Last Name key \"CLT*\"."
    },
{
ERR_ORDERSTATUS_MISSING_DID_KEY,
"Order Status missing District key \"DID*\"."
    },
{
ERR_PAYMENT_CDI_INVALID,
"Payment Customer district invalid
must be numeric."
    },
{
ERR_PAYMENT_CID_AND_CLT,
"Payment Only Customer ID or Last
Name may be entered, not both." },
{
ERR_PAYMENT_CUSTOMER_INVALID,
"Payment Customer data type invalid, must be numeric."
    },
{
ERR_PAYMENT_CWI_INVALID,
"Payment Customer Warehouse
invalid, must be numeric."
    },
{
ERR_PAYMENT_DISTRICT_INVALID,
"Payment District ID is invalid, must be 1 - 10."
    },
{
ERR_PAYMENT_HAM_INVALID,
"Payment Amount invalid data type
must be numeric."
    },
{
ERR_PAYMENT_HAM_RANGE,
"Payment Amount out of

```

```

range, 0 - 9999.99."
    },
    {
ERR_PAYMENT_LAST_NAME_TO_LONG,
"Payment Customer last name longer than 16 characters."
    },
    {
ERR_PAYMENT_MISSING_CDI_KEY,
"Payment missing Customer district key \"CDI*\"."
    },
    {
ERR_PAYMENT_MISSING_CID_CLT,
"Payment Either Customer ID or Last Name must be entered."
    },
    {
ERR_PAYMENT_MISSING_CID_KEY,
"Payment missing Customer Key \"CID*\"."
    },
    {
ERR_PAYMENT_MISSING_CLT_KEY,
"Payment missing Customer Last Name key \"CLT*\"."
    },
    {
ERR_PAYMENT_MISSING_CWI_KEY,
"Payment missing Customer Warehouse key \"CWI*\"."
    },
    {
ERR_PAYMENT_MISSING_DID_KEY,
"Payment missing District Key \"DID*\"."
    },
    {
ERR_PAYMENT_MISSING_HAM_KEY,
"Payment missing Amount key \"HAM*\"."
    },
{
ERR_STOCKLEVEL_MISSING_THRESHOLD_KEY,
"Stock Level; missing Threshold key \"TT*\"."
    },
{
ERR_STOCKLEVEL_THRESHOLD_INVALID,
"Stock Level; Threshold value must be in the range = 1 - 99."
},
{
ERR_STOCKLEVEL_THRESHOLD_RANGE,
"Stock Level Threshold out of range, range must be 1 - 99."
    },
{
ERR_VERSION_MISMATCH,
"Invalid version field. RTE and Web
Client are probably out of sync." },
{
ERR_W_ID_INVALID,
"Invalid Warehouse ID."
    },
{
0,
""
    },
};

char szTmp[256];
int i = 0;
while (TRUE)
{
    if (errorMsgs[i].szMsg[0] == 0)
    {
        strcpy( szTmp, "Unknown error
number.");
    }
}

```

```

                break;
            }
            if (m_Error == errorMsgs[i].iError)
            {
                strcpy( szTmp, errorMsgs[i].szMsg );
                break;
            }
            i++;
        }

        if (m_szTextDetail)
            strcat( szTmp, m_szTextDetail );
        if (m_SystemErr)
            sprintf( szTmp+strlen(szTmp), " Error=%d",
m_SystemErr );

        m_szErrorText = new char[strlen(szTmp)+1];
        strcpy( m_szErrorText, szTmp );
        return m_szErrorText;
    }

/* FUNCTION: GetKeyValue
*
* PURPOSE:      This function parses a http formatted string for
specific key values.
*
* ARGUMENTS:   char                *pQueryString
http string from client browser
*
*              char
* pKey          key value to look for
*              char
* pValue       character array into which to place
key's value
*
*              int
iMax          maximum length of key
value array.
*
*              WEBERROR
err           error value to throw
*
* RETURNS:     nothing.
*
* ERROR:       if (the pKey value is not found) then
if (err == 0)
*
return (empty string)
*
*              else
*
throw CWEBCLNT_ERR(err)
*
* COMMENTS:    http keys are formatted either KEY=value& or
KEY=value\0. This DLL formats
*
*              TPC-C input fields in
such a manner that the keys can be extracted in the
*
*              above manner.
*/

void GetKeyValue(char **pQueryString, char *pKey, char *pValue,
int iMax, WEBERROR err)
{
    char *ptr;

    if ( !(ptr=strstr(*pQueryString, pKey)) )
        goto ErrorExit;
    ptr += strlen(pKey);
    if ( *ptr != '=' )
        goto ErrorExit;
    ptr++;

```

```

iMax--; // one position is for terminating null
while( *ptr && *ptr != '&' && iMax)
{
    *pValue++ = *ptr++;
    iMax--;
}
*pValue = 0; // terminating null

*pQueryString = ptr;
return;

ErrorExit:
if (err != NO_ERR)
    throw new CWEBCLNT_ERR( err );
*pValue = 0; // return empty result string
}

/* FUNCTION: GetIntKeyValue
*
* PURPOSE:      This function parses a http formatted string for a
specific key value.
*
* ARGUMENTS:   char                *pQueryString
http string from client browser
*
*              char
* pKey          key value to look for
*              WEBERROR
NoKeyErr       error value to throw if key not found
*              WEBERROR
NotIntErr      error value to throw if value not numeric
*
* RETURNS:     integer
*
* ERROR:       if (the pKey value is not found) then
if (NoKeyErr
!= NO_ERR)
*
throw CWEBCLNT_ERR(err)
*
*              else
*
return 0
*
*              else if (non-numeric char
found) then
*
*              if (NotIntErr !=
NO_ERR) then
*
throw CWEBCLNT_ERR(err)
*
*              else
*
return 0
*
* COMMENTS:    http keys are formatted either KEY=value& or
KEY=value\0. This DLL formats
*
*              TPC-C input fields in
such a manner that the keys can be extracted in the
*
*              above manner.
*/

int GetIntKeyValue(char **pQueryString, char *pKey, WEBERROR
NoKeyErr, WEBERROR NotIntErr)
{
    char *ptr0;
    char *ptr;

    if ( !(ptr=strstr(*pQueryString, pKey)) )
        goto ErrorNoKey;
    ptr += strlen(pKey);

```

```

if ( *ptr != '=' )
    goto ErrorNoKey;
ptr++;

ptr0 = ptr;          // remember starting point
// scan string until a terminator (null or &) or a non-digit
while( *ptr && *ptr != '&' && isdigit(*ptr) )
    ptr++;

// make sure we stopped scanning for the right reason
if ((ptr0 == ptr) || (*ptr && *ptr != '&'))
{
    if (NotIntErr != NO_ERR)
        throw new CWEBCLNT_ERR(
NoKeyErr);
    return 0;
}

*pQueryString = ptr;
return atoi(ptr0);

ErrorNoKey:
if (NoKeyErr != NO_ERR)
    throw new CWEBCLNT_ERR( NoKeyErr );
return 0;
}

/* FUNCTION: TermInit
*
* PURPOSE:      This function initializes the client terminal
structure; it is called when the TPCC.DLL
*
*               is first loaded by the inet service.
*
*/

void TermInit(void)
{
    EnterCriticalSection(&TermCriticalSection);

    Term.iMasterSyncId = 1;
    Term.iNumEntries = Reg.dwMaxConnections+1;

    Term.pClientData = NULL;
    Term.pClientData =
(PCLIENTDATA)malloc(Term.iNumEntries*
sizeof(CLIENTDATA));
    if (Term.pClientData == NULL)
    {
        LeaveCriticalSection(&TermCriticalSection);
        throw new CWEBCLNT_ERR(
ERR_MEM_ALLOC_FAILED);
    }

    ZeroMemory( Term.pClientData, Term.iNumEntries *
sizeof(CLIENTDATA));

    Term.iFreeList = Term.iNumEntries-1;
    // build free list
    // note: Term.pClientData[0].iNextFree gets set to -1, which
marks it as "in use".
    // This is intentional, as the zero entry is used as an
anchor and never
    // allocated as an actual terminal.
    for(int i=0; i<Term.iNumEntries; i++)
        Term.pClientData[i].iNextFree = i-1;

    LeaveCriticalSection(&TermCriticalSection);
}

```

```

/* FUNCTION: TermDeleteAll
*
* PURPOSE:      This function frees allocated resources
associated with the terminal structure.
*
* ARGUMENTS:   none
*
* RETURNS:     None
*
* COMMENTS:    This function is called only when the inet service
unloads the TPCC.DLL
*/

void TermDeleteAll(void)
{
    EnterCriticalSection(&TermCriticalSection);

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

    Term.iFreeList = 0;
    Term.iNumEntries = 0;
    if ( Term.pClientData )
        free(Term.pClientData);
    Term.pClientData = NULL;

    LeaveCriticalSection(&TermCriticalSection);
}

/* FUNCTION: TermAdd
*
* PURPOSE:      This function assigns a terminal id which is used
to identify a client browser.
*
* RETURNS:     int assigned
terminal id
*/

int TermAdd(void)
{
    DWORD i;
    int iNewTerm, iTickCount;

    if (Term.iNumEntries == 0)
        return -1;

    EnterCriticalSection(&TermCriticalSection);
    if (Term.iFreeList != 0)
    {
        // position is available
        iNewTerm = Term.iFreeList;
        Term.iFreeList =
Term.pClientData[iNewTerm].iNextFree;
        Term.pClientData[iNewTerm].iNextFree = -1; //
indicates this position is in use
    }
    else
    {
        // no open slots, so find the slot that hasn't been
used in the longest time and reuse it
        for(iNewTerm=1, i=1,
iTickCount=0x7FFFFFFF; i<Reg.dwMaxConnections; i++)
        {

```

```

        if (iTickCount >
Term.pClientData[i].iTickCount)
        {
            iTickCount =
Term.pClientData[i].iTickCount;
            iNewTerm = i;
        }
        // if oldest term is less than one minute old, it
probably means that more connections
        // are being attempted than were specified as
"Max Connections" at install. In this case,
        // do not bump existing connection; instead,
return error to requestor.
        if ((GetTickCount() - iTickCount) < 60000)
        {
LeaveCriticalSection(&TermCriticalSection);
            throw new CWEBCLNT_ERR(
ERR_MAX_CONNECTIONS_EXCEEDED);
        }

        Term.pClientData[iNewTerm].iTickCount =
GetTickCount();
        Term.pClientData[iNewTerm].iSyncId =
Term.iMasterSyncId++;
        Term.pClientData[iNewTerm].pTxn = NULL;

        LeaveCriticalSection(&TermCriticalSection);
        return iNewTerm;
}

/* FUNCTION: TermDelete
*
* PURPOSE:      This function makes a terminal entry in the Term
array available for reuse.
*
* ARGUMENTS:   int
                id                Terminal id of client exiting
*
*/

void TermDelete(int id)
{
    if ( id > 0 && id < Term.iNumEntries )
    {
        delete Term.pClientData[id].pTxn;

        // put onto free list
        EnterCriticalSection(&TermCriticalSection);

        Term.pClientData[id].iNextFree =
Term.iFreeList;
        Term.iFreeList = id;

        LeaveCriticalSection(&TermCriticalSection);
    }
}

/* FUNCTION: MakeErrorForm
*/

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

```

```

        wsprintf(szBuffer,
                "<HTML><HEAD><TITLE>TPC-C
Error</TITLE></HEAD><BODY>"
                "<FORM ACTION='tpcc.dll'"
METHOD="GET">"
                "<INPUT TYPE='hidden'"
NAME="STATUSID" VALUE="%d">"
                "<INPUT TYPE='hidden'" NAME="ERROR"
VALUE="%d">"
                "<INPUT TYPE='hidden'"
NAME="FORMID" VALUE="%d">"
                "<INPUT TYPE='hidden'"
NAME="TERMID" VALUE="%d">"
                "<INPUT TYPE='hidden'"
NAME="SYNCID" VALUE="%d">"
                "<BOLD>An Error
Occurred</BOLD><BR><BR>"
                "%s"
                "<BR><BR><HR>"
                "<INPUT TYPE='submit'" NAME="CMD"
VALUE="..NewOrder..">"
                "<INPUT TYPE='submit'" NAME="CMD"
VALUE="..Payment..">"
                "<INPUT TYPE='submit'" NAME="CMD"
VALUE="..Delivery..">"
                "<INPUT TYPE='submit'" NAME="CMD"
VALUE="..Order-Status..">"
                "<INPUT TYPE='submit'" NAME="CMD"
VALUE="..Stock-Level..">"
                "<INPUT TYPE='submit'" NAME="CMD"
VALUE="..Exit..">"
                "</FORM></BODY></HTML>"
                , iType, iErrorNum, MAIN_MENU_FORM,
iTermId, iSyncId, szErrorText );
}

/* FUNCTION: MakeMainMenuForm
*/

void MakeMainMenuForm(int iTermId, int iSyncId, char *szForm)
{
    wsprintf(szForm,
            "<HTML><HEAD><TITLE>TPC-CMain
Menu</TITLE></HEAD><BODY>"
            "Select Desired Transaction.<BR><HR>"
            "<FORM ACTION='tpcc.dll'"
METHOD="GET">"
            "<INPUT TYPE='hidden'"
NAME="STATUSID" VALUE="0">"
            "<INPUT TYPE='hidden'" NAME="ERROR"
VALUE="0">"
            "<INPUT TYPE='hidden'"
NAME="FORMID" VALUE="%d">"
            "<INPUT TYPE='hidden'"
NAME="TERMID" VALUE="%d">"
            "<INPUT TYPE='hidden'"
NAME="SYNCID" VALUE="%d">"
            "<INPUT TYPE='submit'" NAME="CMD"
VALUE="..NewOrder..">"
            "<INPUT TYPE='submit'" NAME="CMD"
VALUE="..Payment..">"
            "<INPUT TYPE='submit'" NAME="CMD"
VALUE="..Delivery..">"
            "<INPUT TYPE='submit'" NAME="CMD"
VALUE="..Order-Status..">"
            "<INPUT TYPE='submit'" NAME="CMD"
VALUE="..Stock-Level..">"

```

```

        "<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"..Exit..\">"
    "</FORM></BODY></HTML>"
    , MAIN_MENU_FORM, iTermId, iSyncId);
}

/* FUNCTION: MakeStockLevelForm
*
* PURPOSE:      This function constructs the Stock Level HTML
page.
*
* COMMENTS:    The internal client buffer is created when the
terminal id is assigned and should not
*
*              be freed except when the
client terminal id is no longer needed.
*/

void MakeStockLevelForm(int iTermId, STOCK_LEVEL_DATA
*pStockLevelData, BOOL bInput, char *szForm)
{
    int    c;

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

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

```

```

        "<BR> <BR> <BR> <BR> <BR>
<BR> <BR> <BR></PRE><HR>"
        "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..NewOrder..\">"
        "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..Payment..\">"
        "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..Delivery..\">"
        "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..Order-Status..\">"
        "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..Stock-Level..\">"
        "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..Exit..\">"
        "</FORM></HTML>"
        , pStockLevelData->threshold,
pStockLevelData->low_stock);
    }
}

/* FUNCTION: MakeNewOrderForm
*
* COMMENTS:    The internal client buffer is created when the
terminal id is assigned and should not
*
*              be freed except when the
client terminal id is no longer needed.
*/

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

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

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

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

    if ( bInput )
    {

```

```

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

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

```

```

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

                if ( bValid )
                {
                        c += sprintf(szForm+c,
"%2.2d-%2.2d-%4.4d %2.2d:%2.2d:%2.2d",
pNewOrderData->o_entry_d.day,
pNewOrderData->o_entry_d.month,
pNewOrderData->o_entry_d.year,
pNewOrderData->o_entry_d.hour,
pNewOrderData->o_entry_d.minute,
pNewOrderData->o_entry_d.second);
                }

                c += sprintf(szForm+c, "<BR>Customer:
%4.4d Name: %-16s Credit: %-2s ",
                pNewOrderData->c_id,
                pNewOrderData->c_last, pNewOrderData->c_credit);

                if ( bValid )
                {
                        c += sprintf(szForm+c,
"%%Disc: %5.2f
                <BR>"
                "Order Number: %8.8d Number of Lines: %2.2d
                W_tax: %5.2f
                D_tax: %5.2f <BR> <BR>"
                "
                Supp_W Item_Id Item Name
                Qty Stock B/G Price
                Amount<BR>",
                100.0*pNewOrderData->c_discount,
                pNewOrderData->o_id,
                pNewOrderData->o_ol_cnt,
                100.0 *
                pNewOrderData->w_tax,
                100.0 *
                pNewOrderData->d_tax);

                        for(i=0;
i<pNewOrderData->o_ol_cnt; i++)
                        {
                                c += sprintf(szForm+c, "
%6.6d %6.6d %-24s %2.2d %3.3d %1.1s
                $%6.2f $%7.2f
                <BR>",
                pNewOrderData->OL[i].ol_supply_w_id,
                pNewOrderData->OL[i].ol_i_id,
                pNewOrderData->OL[i].ol_i_name,
                pNewOrderData->OL[i].ol_quantity,
                pNewOrderData->OL[i].ol_stock,

```

```

pNewOrderData->OL[i].ol_brand_generic,
pNewOrderData->OL[i].ol_i_price,
pNewOrderData->OL[i].ol_amount );
    }
    else
    {
        c += sprintf(szForm+c,
            "% Disc:<BR>"
            "Order Number: %8.8d
Number of Lines:      W_tax:      D_tax:<BR> <BR>"
            " Supp_W Item_Id Item
Name          Qty Stock B/G Price  Amount<BR>"
            , pNewOrderData->o_id);

        i = 0;
    }

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

if ( bValid )
    c += sprintf(szForm+c, "Execution
Status: Transaction committed.      Total: $%8.2f ",

pNewOrderData->total_amount);
    else
    c += sprintf(szForm+c, "Execution
Status: Item number is not valid.      Total:");

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

/* FUNCTION: MakePaymentForm
*
* COMMENTS:      The internal client buffer is created when the
terminal id is assigned and should not
*                  be freed except when the
client terminal id is no longer needed.
*/

void MakePaymentForm(int iTermId, PAYMENT_DATA
*pPaymentData, BOOL bInput, char *szForm)
{
    int c;

    c = sprintf(szForm,
        "<HTML><HEAD><TITLE>TPC-C
Payment</TITLE></HEAD><BODY>"

```

```

        "<FORM ACTION=\"tpcc.dll\"
METHOD=\"GET\">"
        "<INPUT TYPE=\"hidden\"
NAME=\"STATUSID\" VALUE=\"0\">"
        "<INPUT TYPE=\"hidden\" NAME=\"ERROR\"
VALUE=\"0\">"
        "<INPUT TYPE=\"hidden\"
NAME=\"FORMID\" VALUE=\"%d\">"
        "<INPUT TYPE=\"hidden\"
NAME=\"TERMINID\" VALUE=\"%d\">"
        "<INPUT TYPE=\"hidden\"
NAME=\"SYNCID\" VALUE=\"%d\">"
        "<PRE><font face=\"Courier\">
Payment<BR>"
        "Date: "
        , PAYMENT_FORM, iTermId,
Term.pClientData[iTermId].iSyncId);

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

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

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

```



```

"%-20s %-2s %5.5s-%4.4s  %-20s
%-2s %5.5s-%4.4s<BR> <BR>"
"Customer: %4.4d Cust-Warehouse:
%6.6d Cust-District: %2.2d<BR>"
"Name: %-16s %-2s %-16s
Since: %2.2d-%2.2d-%4.4d<BR>"
" %-20s          Credit:
%-2s<BR>"

, Term.pClientData[iTermId].w_id,
pPaymentData->d_id
, pPaymentData->w_street_1,
pPaymentData->d_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>"
<BR>"
"Credit Limit:  $%13.2f<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;
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='TERMINID' VALUE='%d'>"
"<INPUT TYPE='hidden'
NAME='SYNCID' VALUE='%d'>"
"<PRE><font face='Courier'>
Order-Status<BR>"
"Warehouse: %6.6d ",
ORDER_STATUS_FORM, iTermId,
Term.pClientData[iTermId].iSyncId,
Term.pClientData[iTermId].w_id);

if ( bInput )
{
strcpy(szForm+c,

```

```

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

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

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

        pOrderStatusData->o_entry_d.minute,

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

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

```

```

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

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

/* FUNCTION: MakeDeliveryForm
*
* COMMENTS: The internal client buffer is created when the
terminal id is assigned and should not
* be freed except when the
client terminal id is no longer needed.
*/

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

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

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

```

```

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

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

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

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

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

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

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

```

```

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

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

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

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

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

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

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

```

```

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

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

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

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

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

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

    PDELIVERY_DATA pDelivery;

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

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

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

```

```

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

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

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

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

    PSTOCK_LEVEL_DATA    pStockLevel;

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

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

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

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

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

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

```

```

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

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

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

    for(i=0, items=0; i<MAX_OL_NEW_ORDER_ITEMS;
i++)
    {
        GetKeyValue(&ptr, szSP[i], szTmp,
sizeof(szTmp), ERR_NEWORDER_MISSING_SUPPW_KEY);
        if ( szTmp[0] )
        {
            if ( !IsNumeric(szTmp) )
                throw new
CWEBCLNT_ERR( ERR_NEWORDER_SUPPW_INVALID);

            pNewOrderData->OL[items].ol_supply_w_id = (long)atoi(szTmp);
            if (!fRemoteFlag)
            {
                if
(!pbLocalWhsTbl[pNewOrderData->OL[items].ol_supply_w_id])
                    fRemoteFlag=TRUE;
            }

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

```

```

                throw new
CWEBCLNT_ERR( ERR_NEWORDER_ITEMID_RANGE);

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

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

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

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

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

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

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

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

```

```

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

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

fRemoteFlag=TRUE;

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

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

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

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

```

```

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

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

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

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

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

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

```

```

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

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

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

    if ( *ptr == 0 )
        return FALSE;

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

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

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

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

```

tpc.def

```

LIBRARY TPCC.DLL

EXPORTS

    GetExtensionVersion @1
    HttpExtensionProc  @2
    TerminateExtension @3

```

tpcc.h

```

/*      FILE:          TPCC.H
 *
 *      Microsoft TPC-C Kit Ver.
4.20.000
 *
 *      Copyright Microsoft,
1999
 *
 *      All Rights Reserved
 *
 *
 *      Version 4.10.000 audited
by Richard Gimarc, Performance Metrics, 3/17/99
 *
 *      PURPOSE:      Header file for ISAPI TPCC.DLL,
defines structures and functions used in the isapi tpcc.dll.
 *
 */

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

#define TP_MAX_RETRIES
    50

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

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

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

```

```

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

CTPCC_BASE *pTxn;
} CLIENTDATA, *PCLIENTDATA;

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

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

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

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

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

```

```

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

ERR_STOCKLEVEL_MISSING_THRESHOLD_KEY,

ERR_STOCKLEVEL_THRESHOLD_INVALID,
ERR_STOCKLEVEL_THRESHOLD_RANGE,
ERR_VERSION_MISMATCH,
ERR_W_ID_INVALID
};

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

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

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

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

int ErrorType() {return
ERR_TYPE_WEBDLL;};

```



```

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

};

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

//function prototypes

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

```

```

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

```

tpcc.rc

```

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

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

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

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

```

```

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

#endif // !_MAC

#ifdef APSTUDIO_INVOKED
//
// TEXTINCLUDE
//

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

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

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

#endif // APSTUDIO_INVOKED

//
// Dialog
//

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

//
// DESIGNINFO
//

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

```

```

#endif // APSTUDIO_INVOKED

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

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

#endif // not APSTUDIO_INVOKED

tpcc_com.cpp

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

// needed for CoInitializeEx
#define _WIN32_WINNT 0x0400

#include <windows.h>

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

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

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

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

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

CTPCC_COM::CTPCC_COM(BOOL bSinglePool)
{

```

```

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

m_bSinglePool = bSinglePool;

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

    ReleaseInterface(m_pNewOrder);

    ReleaseInterface(m_p_remNewOrder);

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

```

```

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

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

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

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

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

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

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

```

```

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

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

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

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

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

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

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

```

tpcc_com.h

```

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

```

```

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

#pragma once

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

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

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

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

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

    int          m_hr;
    int          m_iErrorType;
    int          m_iError;

```

```

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

    int ErrorNum() {return m_hr;}

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

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

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

    struct COM_DATA

```

```

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

        VARIANT m_vTxn;

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

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

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

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

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

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

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

inline void ReleaseInterface(IUnknown *pUnk)

```

```

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

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

typedef CTPCC_COM* (TYPE_CTPCC_COM)(BOOL);

tpcc_com_all.cpp

/*      FILE:          TPCC_COM_ALL.CPP
 *      Microsoft TPC-C Kit Ver.
4.20.000
 *      Copyright Microsoft,
1999
 *      All Rights Reserved
 *
 *      Version 4.10.000 audited
by Richard Gimarc, Performance Metrics, 3/17/99
 *
 *      PURPOSE:      Implementation for TPC-C Tuxedo
class.
 *      Contact:    Charles Levine (clevine@microsoft.com)
 *
 *      Change history:
 *      4.20.000 - updated rev number to match kit
 */

#define STRICT
#define _WIN32_WINNT 0x0400
#define _ATL_APARTMENT_THREADED

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

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

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

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

```

```

#include "..\..\common\src\txn_base.h"
#include "..\..\common\src\error.h"
#include "..\..\common\src\ReadRegistry.h"
#include "..\..\db_dblib_dll\src\tpcc_dblib.h" //
DBLIB implementation of TPC-C txns
#include "..\..\db_odbc_dll\src\tpcc_odbc.h" // ODBC
implementation of TPC-C txns

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

CComModule _Module;

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

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

static HINSTANCE hLibInstanceDb = NULL;

TYPE_CTPCC_DBLIB *pCTPCC_DBLIB_new;
TYPE_CTPCC_ODBC *pCTPCC_ODBC_new;

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

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

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

DisableThreadLibraryCalls(hInstance);

            DWORD dwSize =
MAX_COMPUTERNAME_LENGTH+1;

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

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

            if (Reg.eDB_Protocol == DBLIB)

```

```

        {
            strcpy( szDllName,
Reg.szPath );

            strcat( szDllName,
"tpcc_dblib.dll");

LoadLibrary( szDllName );

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

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

            strcat( szDllName,
"tpcc_odbc.dll");

LoadLibrary( szDllName );

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

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

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

```

```

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

    return TRUE;    // OK
}

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

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

////////////////////////////////////
// Returns a class factory to create an object of the requested type

STDAPI DllGetClassObject(REFCLSID rclsid, REFIID riid, LPVOID*
ppv)
{
    return _Module.GetClassObject(rclsid, riid, ppv);
}

////////////////////////////////////
// DllRegisterServer - Adds entries to the system registry

STDAPI DllRegisterServer(void)
{
    // registers object, typelib and all interfaces in typelib
    return _Module.RegisterServer(TRUE);
}

////////////////////////////////////
// DllUnregisterServer - Removes entries from the system registry

STDAPI DllUnregisterServer(void)
{
    _Module.UnregisterServer();
    return S_OK;
}

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

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

    _stprintf(szMsg, TEXT("Error in COM+ TPC-C Component: "));
    lpszStrings[0] = szMsg;
    lpszStrings[1] = lpszMsg;

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

```

```

(LPCTSTR *)lpszStrings, // array of error strings
NULL); // no raw data

(VOID) DeregisterEventSource(hEventSource);
}

inline void ReleaseInterface(IUnknown *pUnk)
{
    if (pUnk)
    {
        pUnk->Release();
        pUnk = NULL;
    }
}

/* FUNCTION: CCOMPONENT_ERR::ErrorText
*
*/
char* CCOMPONENT_ERR::ErrorText(void)
{
    static SERRORMSG errorMsgs[] =
    {
        { ERR_MISSING_REGISTRY_ENTRIES,
"Required entries missing from registry."
},
        { ERR_LOADDLL_FAILED,
"Load of DLL failed. DLL="
},
        { ERR_GETPROCADDR_FAILED,
"Could not map proc in DLL. 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* pObjectContext = NULL;

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

//
// called by the ctor activator
//
STDMETHODIMP CTPCC_Common::Construct(IDispatch * pUnk)
{
    // Code to access construction string, if needed later...
    // if (!pUnk)
    //     return E_UNEXPECTED;
    // IObjectConstructString * pString = NULL;
    // HRESULT hr =
pUnk->QueryInterface(IID_IObjectConstructString, (void
**)&pString);
    // pString->Release();
    try
    {
        if (Reg.eDB_Protocol == ODBC)
            m_pTxn = pCTPCC_ODBC_new(
Reg.szDbServer, Reg.szDbUser, Reg.szDbPassword,
szMyComputerName, Reg.szDbName );
        else if (Reg.eDB_Protocol == DBLIB)
            m_pTxn = pCTPCC_DBLIB_new(
Reg.szDbServer, Reg.szDbUser, Reg.szDbPassword,
szMyComputerName, Reg.szDbName );
    }
    catch (CBaseErr *e)
    {
        WriteMessageToEventLog(e->ErrorText());
        delete e;
        return E_FAIL;
    }
    catch (...)
    {
        WriteMessageToEventLog(TEXT("Unhandled
exception in object ::Construct"));
        return E_FAIL;
    }

    return S_OK;
}

```

```

HRESULT CTPCC_Common::NewOrder(VARIANT txn_in,
VARIANT* txn_out)
{
    PNEW_ORDER_DATA pNewOrder;
    COM_DATA *pData;
    try
    {
        pData = (COM_DATA*)txn_in.parray->pvData;
        pNewOrder = m_pTxn->BuffAddr_NewOrder();

        memcpy(pNewOrder, &pData->u.NewOrder,
sizeof(NEW_ORDER_DATA));

        VariantInit(txn_out);
        txn_out->vt = VT_SAFEARRAY;
        txn_out->parray = SafeArrayCreateVector(
VT_UI1,
txn_in.parray->rgsabound->cElements,
txn_in.parray->rgsabound->cElements);
        pData = (COM_DATA*)
txn_out->parray->pvData;

        m_pTxn->NewOrder(TRUE);

        memcpy(&pData->u.NewOrder, pNewOrder,
sizeof(NEW_ORDER_DATA));

        pData->retval = ERR_SUCCESS;
        pData->error = 0;
        return S_OK;
    }
    catch (CBaseErr *e)
    {
        // check for lost database connection; if yes,
component is toast
        if ( ((e->ErrorType() == ERR_TYPE_DBLIB)
&& (e->ErrorNum() == 10005)) ||
((e->ErrorType() ==
ERR_TYPE_ODBC) && (e->ErrorNum() == 10054)) )
            m_bCanBePooled = FALSE;

        pData->retval = e->ErrorType();
        pData->error = e->ErrorNum();
        delete e;
        return E_FAIL;
    }
    catch (...)
    {
        WriteMessageToEventLog(TEXT("Unhandled
exception."));

        pData->retval = ERR_TYPE_LOGIC;
        pData->error = 0;
        m_bCanBePooled = FALSE;
        return E_FAIL;
    }
}

HRESULT CTPCC_Common::Payment(VARIANT txn_in,
VARIANT* txn_out)
{
    PPAYMENT_DATA pPayment;
    COM_DATA *pData;
    try
    {
        pData = (COM_DATA*)txn_in.parray->pvData;

```

```

        pPayment = m_pTxn->BuffAddr_Payment();

        memcpy(pPayment, &pData->u.Payment,
sizeof(PAYMENT_DATA));

        VariantInit(txn_out);
        txn_out->vt = VT_SAFEARRAY;
        txn_out->parray = SafeArrayCreateVector(
VT_UI1,

txn_in.parray->rgsabound->cElements,

txn_in.parray->rgsabound->cElements);
        pData = (COM_DATA*)
txn_out->parray->pvData;

        m_pTxn->Payment(TRUE);

        memcpy( &pData->u.Payment, pPayment,
sizeof(PAYMENT_DATA));

        pData->retval = ERR_SUCCESS;
        pData->error = 0;
        return S_OK;
    }
    catch (CBaseErr *e)
    {
        // check for lost database connection; if yes,
component is toast
        if ( ((e->ErrorType() == ERR_TYPE_DBLIB)
&& (e->ErrorNum() == 10005)) ||
            ((e->ErrorType() ==
ERR_TYPE_ODBC) && (e->ErrorNum() == 10054)) )
            m_bCanBePooled = FALSE;

        pData->retval = e->ErrorType();
        pData->error = e->ErrorNum();
        delete e;
        return E_FAIL;
    }
    catch (...)
    {
        WriteMessageToEventLog(TEXT("Unhandled
exception."));

        pData->retval = ERR_TYPE_LOGIC;
        pData->error = 0;
        m_bCanBePooled = FALSE;
        return E_FAIL;
    }
}

HRESULT CTPCC_Common::StockLevel(VARIANT txn_in,
VARIANT* txn_out)
{
    PSTOCK_LEVEL_DATA    pStockLevel;
    COM_DATA              *pData;

    try
    {
        pData = (COM_DATA*)txn_in.parray->pvData;
        pStockLevel =
m_pTxn->BuffAddr_StockLevel();

        memcpy(pStockLevel, &pData->u.StockLevel,
sizeof(STOCK_LEVEL_DATA));

        VariantInit(txn_out);

```

```

        txn_out->vt = VT_SAFEARRAY;
        txn_out->parray = SafeArrayCreateVector(
VT_UI1,

txn_in.parray->rgsabound->cElements,

txn_in.parray->rgsabound->cElements);
        pData =
(COM_DATA*)txn_out->parray->pvData;

        m_pTxn->StockLevel();

        memcpy( &pData->u.StockLevel, pStockLevel,
sizeof(STOCK_LEVEL_DATA));

        pData->retval = ERR_SUCCESS;
        pData->error = 0;
        return S_OK;
    }
    catch (CBaseErr *e)
    {
        // check for lost database connection; if yes,
component is toast
        if ( ((e->ErrorType() == ERR_TYPE_DBLIB)
&& (e->ErrorNum() == 10005)) ||
            ((e->ErrorType() ==
ERR_TYPE_ODBC) && (e->ErrorNum() == 10054)) )
            m_bCanBePooled = FALSE;

        pData->retval = e->ErrorType();
        pData->error = e->ErrorNum();
        delete e;
        return E_FAIL;
    }
    catch (...)
    {
        WriteMessageToEventLog(TEXT("Unhandled
exception."));

        pData->retval = ERR_TYPE_LOGIC;
        pData->error = 0;
        m_bCanBePooled = FALSE;
        return E_FAIL;
    }
}

HRESULT CTPCC_Common::OrderStatus(VARIANT txn_in,
VARIANT* txn_out)
{
    PORDER_STATUS_DATA    pOrderStatus;
    COM_DATA              *pData;
    try
    {
        pData = (COM_DATA*)txn_in.parray->pvData;
        pOrderStatus =
m_pTxn->BuffAddr_OrderStatus();

        memcpy(pOrderStatus, &pData->u.OrderStatus,
sizeof(ORDER_STATUS_DATA));

        VariantInit(txn_out);
        txn_out->vt = VT_SAFEARRAY;
        txn_out->parray = SafeArrayCreateVector(
VT_UI1,

txn_in.parray->rgsabound->cElements,

```

```

txn_in.parray->rgsabound->cElements);
    pData =
(COM_DATA*)txn_out->parray->pvData;

    m_pTxn->OrderStatus();

    memcpy( &pData->u.OrderStatus, pOrderStatus,
sizeof(ORDER_STATUS_DATA));

    pData->retval = ERR_SUCCESS;
    pData->error = 0;
    return S_OK;
}
catch (CBaseErr *e)
{
    // check for lost database connection; if yes,
component is toast
    if ( ((e->ErrorType() == ERR_TYPE_DBLIB)
&& (e->ErrorNum() == 10005)) ||
        ((e->ErrorType() ==
ERR_TYPE_ODBC) && (e->ErrorNum() == 10054)) )
        m_bCanBePooled = FALSE;

    pData->retval = e->ErrorType();
    pData->error = e->ErrorNum();
    delete e;
    return E_FAIL;
}
catch (...)
{
    WriteMessageToEventLog(TEXT("Unhandled
exception."));

    pData->retval = ERR_TYPE_LOGIC;
    pData->error = 0;
    m_bCanBePooled = FALSE;
    return E_FAIL;
}
}

```

tpcc_com_all.def

: tpcc_com_all.def : Declares the module parameters.

```

LIBRARY "tpcc_com_all.dll"

EXPORTS
    DllCanUnloadNow @1 PRIVATE
    DllGetClassObject @2 PRIVATE
    DllRegisterServer @3 PRIVATE
    DllUnregisterServer @4 PRIVATE

```

tpcc_com_all.h

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

```

/* File created by MIDL compiler version 5.01.0164 */
/* at Fri Jan 19 11:14:55 2001
*/
/* Compiler settings for .\src\tpcc_com_all.idl:
    Oicf (OptLev=i2), W1, Zp8, env=Win32, ms_ext, c_ext
    error checks: allocation ref bounds_check enum stub_data
*/
//@@@MIDL_FILE_HEADING( )

```

/* verify that the <rpcndr.h> version is high enough to compile this file*/

```

#ifndef __REQUIRED_RPCNDR_H_VERSION__
#define __REQUIRED_RPCNDR_H_VERSION__ 440
#endif

```

```

#include "rpc.h"
#include "rpcndr.h"

```

```

#ifndef __tpcc_com_all_h__
#define __tpcc_com_all_h__

```

```

#ifdef __cplusplus
extern "C" {
#endif

```

/* Forward Declarations */

```

#ifndef __TPCC_FWD_DEFINED__
#define __TPCC_FWD_DEFINED__

```

```

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

```

```

#endif /* __TPCC_FWD_DEFINED__ */

```

```

#ifndef __NewOrder_FWD_DEFINED__
#define __NewOrder_FWD_DEFINED__

```

```

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

```

```

#endif /* __NewOrder_FWD_DEFINED__ */

```

```

#ifndef __OrderStatus_FWD_DEFINED__
#define __OrderStatus_FWD_DEFINED__

```

```

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

```

```

#endif /* __OrderStatus_FWD_DEFINED__ */

```

```

#ifndef __Payment_FWD_DEFINED__
#define __Payment_FWD_DEFINED__

```

```

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

```

```

#endif /* __Payment_FWD_DEFINED__ */

```

```

#ifndef __StockLevel_FWD_DEFINED__
#define __StockLevel_FWD_DEFINED__

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

#ifdef __StockLevel_FWD_DEFINED__

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

void __RPC_FAR * __RPC_USER MIDL_user_allocate(size_t);
void __RPC_USER MIDL_user_free(void __RPC_FAR *);

/* interface __MIDL_itf_tpcc_com_all_0000 */
/* [local] */

extern RPC_IF_HANDLE
__MIDL_itf_tpcc_com_all_0000_v0_0_c_ifspec;
extern RPC_IF_HANDLE
__MIDL_itf_tpcc_com_all_0000_v0_0_s_ifspec;

#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 /* __TPCCLib_LIBRARY_DEFINED__ */

/* Additional Prototypes for ALL interfaces */
/* end of Additional Prototypes */

#ifdef __cplusplus
}
#endif

#endif

tpcc_com_all.idl

/* FILE: TPCC.IDL
* Microsoft TPC-C Kit Ver.
4.20.000
* Copyright Microsoft,
1999
* All Rights Reserved
* not yet audited
* PURPOSE: IDL source for TPCC.dll. This file is
processed by the MIDL tool to
* produce the type library
(TPCC.tlb) and marshalling code.
*
* Change history:
* 4.20.000 - first version
*/

interface TPCC;
interface NewOrder;
interface OrderStatus;
interface Payment;
interface StockLevel;

```

```

import "oidl.idl";
import "ocidl.idl";
import "..\tpcc_com_ps\src\tpcc_com_ps.idl";

[
    uuid(122A3117-2520-11D3-BA71-00C04FBFE08B),
    version(1.0),
    helpstring("TPC-C 1.0 Type Library")
]
library TPCCLib
{
    importlib("stdole32.tlb");
    importlib("stdole2.tlb");

    [
        uuid(122A3128-2520-11D3-BA71-00C04FBFE08B),
        helpstring("All Txns Class")
    ]
    coclass TPCC
    {
        [default] interface ITPCC;
    };

    [
        uuid(975BAABF-84A7-11D2-BA47-00C04FBFE08B),
        helpstring("NewOrder Class")
    ]
    coclass NewOrder
    {
        [default] interface ITPCC;
    };

    [
        uuid(266836AD-A50D-11D2-BA4E-00C04FBFE08B),
        helpstring("OrderStatus Class")
    ]
    coclass OrderStatus
    {
        [default] interface ITPCC;
    };

    [
        uuid(CD02F7EF-A4FA-11D2-BA4E-00C04FBFE08B),
        helpstring("Payment Class")
    ]
    coclass Payment
    {
        [default] interface ITPCC;
    };

    [
        uuid(2668369E-A50D-11D2-BA4E-00C04FBFE08B),
        helpstring("StockLevel Class")
    ]
    coclass StockLevel
    {
        [default] interface ITPCC;
    };
}

```

```

};

};

tpcc_com_all.rc

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

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

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

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

#ifndef _AFX_RESOURCE_DLL || defined(_AFX_TARG_ENU)
#ifdef _WIN32
LANGUAGE LANG_ENGLISH, SUBLANG_ENGLISH_US
#pragma code_page(1252)
#endif // _WIN32

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

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

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

3 TEXTINCLUDE DISCARDABLE
BEGIN
    "1 TYPELIB ""tpcc_com_all.tlb""\r\n"
    "\0"
END

#endif // APSTUDIO_INVOKED

#ifdef _MAC
////////////////////////////////////
//
// Version
//

VS_VERSION_INFO VERSIONINFO
FILEVERSION 1,0,0,1
PRODUCTVERSION 1,0,0,1
FILEFLAGSMASK 0x3fL
#ifdef _DEBUG

```

```

FILEFLAGS0x1L
#else
FILEFLAGS0x0L
#endif
FILEOS 0x4L
FILETYPE 0x2L
FILESUBTYPE 0x0L
BEGIN
    BLOCK "StringFileInfo"
    BEGIN
        BLOCK "040904B0"
        BEGIN
            VALUE "CompanyName", "\0"
            VALUE "FileDescription", "tpcc_com_all Module\0"
            VALUE "FileVersion", "1, 0, 0, 1\0"
            VALUE "InternalName", "TPCCNEWORDER\0"
            VALUE "LegalCopyright", "Copyright 1997\0"
            VALUE "OriginalFilename", "tpcc_com_all.DLL\0"
            VALUE "ProductName", "tpcc_com_all Module\0"
            VALUE "ProductVersion", "1, 0, 0, 1\0"
            VALUE "OLESelfRegister", "\0"
        END
    END
    BLOCK "VarFileInfo"
    BEGIN
        VALUE "Translation", 0x409, 1200
    END
END

#endif // !_MAC

////////////////////////////////////
//
// REGISTRY
//

IDR_TPCC          REGISTRY DISCARDABLE
"tpcc_com_all.rgs"
IDR_NEWORDER      REGISTRY DISCARDABLE
"tpcc_com_no.rgs"
IDR_ORDERSTATUS  REGISTRY DISCARDABLE
"tpcc_com_os.rgs"
IDR_PAYMENT       REGISTRY DISCARDABLE
"tpcc_com_pay.rgs"
IDR_STOCKLEVEL   REGISTRY DISCARDABLE
"tpcc_com_sl.rgs"

////////////////////////////////////
//
// String Table
//

STRINGTABLEDISCARDABLE
BEGIN
    IDS_PROJNAME      "tpcc_com_all"
END

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

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

```

```

1 TYPELIB "tpcc_com_all.tlb"

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

tpcc_com_all.rgs

HKCR
{
    TPCC.AllTxns.1 = s 'All Txns Class'
    {
        CLSID = s
' {122A3128-2520-11D3-BA71-00C04FBFE08B}'
    }
    TPCC.AllTxns = s 'TPCC Class'
    {
        CurVer = s 'TPCC.AllTxns.1'
    }
    NoRemove CLSID
    {
        ForceRemove
' {122A3128-2520-11D3-BA71-00C04FBFE08B}' = s 'TPCC Class'
    {
        ProgID = s 'TPCC.AllTxns.1'
        VersionIndependentProgID = s
'TPCC.AllTxns'
        InprocServer32 = s '%MODULE%'
    {
        val ThreadingModel = s
'Both'
    }
    }
}

tpcc_com_all.i.c

/* this file contains the actual definitions of */
/* the IIDs and CLSIDs */

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

/* File created by MIDL compiler version 5.01.0164 */
/* at Fri Jan 19 11:14:55 2001
*/
/* Compiler settings for .\src\tpcc_com_all.idl:
    Oicf (OptLev=i2), W1, Zp8, env=Win32, ms_ext, c_ext
    error checks: allocation ref bounds_check enum stub_data
*/
//@@@MIDL_FILE_HEADING( )
#ifdef __cplusplus
extern "C" {
#endif

#ifdef __IID_DEFINED__
#define __IID_DEFINED__

typedef struct _IID
{
    unsigned long x;
    unsigned short s1;
    unsigned short s2;
    unsigned char c[8];

```

```

} IID;

#endif // __IID_DEFINED__

#ifndef CLSID_DEFINED
#define CLSID_DEFINED
typedef IID CLSID;
#endif // CLSID_DEFINED

const IID LIBID_TPCCLib =
{0x122A3117,0x2520,0x11D3,{0xBA,0x71,0x00,0xC0,0x4F,0xBF,0xE0,0x8B}};

const CLSID CLSID_TPCC =
{0x122A3128,0x2520,0x11D3,{0xBA,0x71,0x00,0xC0,0x4F,0xBF,0xE0,0x8B}};

const CLSID CLSID_NewOrder =
{0x975BAABF,0x84A7,0x11D2,{0xBA,0x47,0x00,0xC0,0x4F,0xBF,0xE0,0x8B}};

const CLSID CLSID_OrderStatus =
{0x266836AD,0xA50D,0x11D2,{0xBA,0x4E,0x00,0xC0,0x4F,0xBF,0xE0,0x8B}};

const CLSID CLSID_Payment =
{0xCD02F7EF,0xA4FA,0x11D2,{0xBA,0x4E,0x00,0xC0,0x4F,0xBF,0xE0,0x8B}};

const CLSID CLSID_StockLevel =
{0x2668369E,0xA50D,0x11D2,{0xBA,0x4E,0x00,0xC0,0x4F,0xBF,0xE0,0x8B}};

#ifdef __cplusplus
}
#endif

```

tpcc_com_no.rgs

```

HKCR
{
    TPCC.NewOrder.1 = s 'NewOrder Class'
    {
        CLSID = s
        '{975BAABF-84A7-11D2-BA47-00C04FBFE08B}'
    }
    TPCC.NewOrder = s 'NewOrder Class'
    {
        CurVer = s 'TPCC.NewOrder.1'
    }
    NoRemove CLSID
    {
        ForceRemove
        {975BAABF-84A7-11D2-BA47-00C04FBFE08B} = s 'NewOrder
        Class'
    }
    ProgID = s 'TPCC.NewOrder.1'
    VersionIndependentProgID = s
    'TPCC.NewOrder'
    InprocServer32 = s '%MODULE%'
}

```

```

{
    val ThreadingModel = s
}
}
}
}
}

tpcc_com_os.rgs

HKCR
{
    TPCC.OrderStatus.1 = s 'OrderStatus Class'
    {
        CLSID = s
        '{266836AD-A50D-11D2-BA4E-00C04FBFE08B}'
    }
    TPCC.OrderStatus = s 'OrderStatus Class'
    {
        CurVer = s 'TPCC.OrderStatus.1'
    }
    NoRemove CLSID
    {
        ForceRemove
        {266836AD-A50D-11D2-BA4E-00C04FBFE08B} = s 'OrderStatus
        Class'
    }
    ProgID = s 'TPCC.OrderStatus.1'
    VersionIndependentProgID = s
    'TPCC.OrderStatus'
    InprocServer32 = s '%MODULE%'
    {
        val ThreadingModel = s
    }
}
}
}
}
}

```

tpcc_com_pay.rgs

```

HKCR
{
    TPCC.Payment.1 = s 'Payment Class'
    {
        CLSID = s
        '{CD02F7EF-A4FA-11D2-BA4E-00C04FBFE08B}'
    }
    TPCC.Payment = s 'Payment Class'
    {
        CurVer = s 'TPCC.Payment.1'
    }
    NoRemove CLSID
    {
        ForceRemove
        {CD02F7EF-A4FA-11D2-BA4E-00C04FBFE08B} = s 'Payment
        Class'
    }
    ProgID = s 'TPCC.Payment.1'
    VersionIndependentProgID = s
    'TPCC.Payment'
    InprocServer32 = s '%MODULE%'
    {
        val ThreadingModel = s
    }
}
}
}
}
}

```

```

    }
}

tpcc_com_ps.def

```

```
LIBRARY "tpcc_com_ps"
```

```
DESCRIPTION 'Proxy/Stub DLL'
```

```
EXPORTS
```

```

    DllGetClassObject    @1    PRIVATE
    DllCanUnloadNow     @2    PRIVATE
    GetProxyDllInfo     @3    PRIVATE
    DllRegisterServer   @4    PRIVATE
    DllUnregisterServer @5    PRIVATE

```

tpcc_com_ps.h

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

```
/* File created by MIDL compiler version 5.01.0164 */
```

```
/* at Fri Jan 19 11:14:46 2001
```

```
*/
```

```
/* Compiler settings for .\src\tpcc_com_ps.idl:
```

```
    Oicf (OptLev=i2), W1, Zp8, env=Win32, ms_ext, c_ext
```

```
error checks: allocation ref bounds_check enum stub_data
```

```
*/
```

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

```
/* verify that the <rpcndr.h> version is high enough to compile this
file*/
```

```
#ifndef __REQUIRED_RPCNDR_H_VERSION__
```

```
#define __REQUIRED_RPCNDR_H_VERSION__ 440
```

```
#endif
```

```
#include "rpc.h"
```

```
#include "rpcndr.h"
```

```
#ifndef __RPCNDR_H_VERSION__
```

```
#error this stub requires an updated version of <rpcndr.h>
```

```
#endif // __RPCNDR_H_VERSION__
```

```
#ifndef COM_NO_WINDOWS_H
```

```
#include "windows.h"
```

```
#include "ole2.h"
```

```
#endif /*COM_NO_WINDOWS_H*/
```

```
#ifndef __tpcc_com_ps_h__
```

```
#define __tpcc_com_ps_h__
```

```
#ifdef __cplusplus
```

```
extern "C"{
```

```
#endif
```

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

```
#ifndef __ITPCC_FWD_DEFINED__
```

```
#define __ITPCC_FWD_DEFINED__
```

```
typedef interface ITPCC ITPCC;
```

```
#endif /* __ITPCC_FWD_DEFINED__ */
```

```
/* header files for imported files */
```

```
#include "oidl.h"
```

```
#include "ocidl.h"
```

```
void __RPC_FAR * __RPC_USER MIDL_user_allocate(size_t);
```

```
void __RPC_USER MIDL_user_free( void __RPC_FAR * );
```

```
/* interface __MIDL_itf_tpcc_com_ps_0000 */
```

```
/* [local] */
```

```
extern RPC_IF_HANDLE
```

```
__MIDL_itf_tpcc_com_ps_0000_v0_0_c_ifspec;
```

```
extern RPC_IF_HANDLE
```

```
__MIDL_itf_tpcc_com_ps_0000_v0_0_s_ifspec;
```

```
#ifndef __ITPCC_INTERFACE_DEFINED__
```

```
#define __ITPCC_INTERFACE_DEFINED__
```

```
/* interface ITPCC */
```

```
/* [unique][helpstring][uuid][oleautomation][object] */
```

```
EXTERN_C const IID IID_ITPCC;
```

```
#if defined(__cplusplus) && !defined(CINTERFACE)
```

```
MIDL_INTERFACE("FEEE6AA2-84B1-11d2-BA47-00C04FBFE08B")
```

```
ITPCC : public IUnknown
```

```
{
```

```
public:
```

```
virtual HRESULT __stdcall NewOrder(
```

```
    /* [in] */ VARIANT txn_in,
```

```
    /* [out] */ VARIANT __RPC_FAR *txn_out) = 0;
```

```
virtual HRESULT __stdcall Payment(
```

```
    /* [in] */ VARIANT txn_in,
```

```
    /* [out] */ VARIANT __RPC_FAR *txn_out) = 0;
```

```
virtual HRESULT __stdcall Delivery(
```

```
    /* [in] */ VARIANT txn_in) = 0;
```

```
virtual HRESULT __stdcall StockLevel(
```

```
    /* [in] */ VARIANT txn_in,
```

```
    /* [out] */ VARIANT __RPC_FAR *txn_out) = 0;
```

```
virtual HRESULT __stdcall OrderStatus(
```

```
    /* [in] */ VARIANT txn_in,
```

```
    /* [out] */ VARIANT __RPC_FAR *txn_out) = 0;
```

```
virtual HRESULT __stdcall CallSetComplete( void) = 0;
```

```
};
```

```
#else /* C style interface */
```

```
typedef struct ITPCCVtbl
```

```
{
```

```
    BEGIN_INTERFACE
```

```
        HRESULT ( STDMETHODCALLTYPE __RPC_FAR
```

```
*QueryInterface)(
```



```

        ITPCC__RPC_FAR * This,
        /* [in] */ REFIID riid,
        /* [iid_is][out] */ void __RPC_FAR * __RPC_FAR
*ppvObject);

        ULONG ( STDMETHODCALLTYPE __RPC_FAR *AddRef)(
        ITPCC__RPC_FAR * This);

        ULONG ( STDMETHODCALLTYPE __RPC_FAR *Release)(
        ITPCC__RPC_FAR * This);

        HRESULT ( STDMETHODCALLTYPE __RPC_FAR *NewOrder)(
        ITPCC__RPC_FAR * This,
        /* [in] */ VARIANT txn_in,
        /* [out] */ VARIANT __RPC_FAR *txn_out);

        HRESULT ( STDMETHODCALLTYPE __RPC_FAR *Payment)(
        ITPCC__RPC_FAR * This,
        /* [in] */ VARIANT txn_in,
        /* [out] */ VARIANT __RPC_FAR *txn_out);

        HRESULT ( STDMETHODCALLTYPE __RPC_FAR *Delivery)(
        ITPCC__RPC_FAR * This,
        /* [in] */ VARIANT txn_in);

        HRESULT ( STDMETHODCALLTYPE __RPC_FAR *StockLevel)(
        ITPCC__RPC_FAR * This,
        /* [in] */ VARIANT txn_in,
        /* [out] */ VARIANT __RPC_FAR *txn_out);

        HRESULT ( STDMETHODCALLTYPE __RPC_FAR *OrderStatus)(
        ITPCC__RPC_FAR * This,
        /* [in] */ VARIANT txn_in,
        /* [out] */ VARIANT __RPC_FAR *txn_out);

        HRESULT ( STDMETHODCALLTYPE __RPC_FAR *CallSetComplete)(
        ITPCC__RPC_FAR * This);

        END_INTERFACE
    } ITPCCVtbl;

    interface ITPCC
    {
        CONST_VTBL struct ITPCCVtbl __RPC_FAR *lpVtbl;
    };

#ifdef COBJMACROS

#define ITPCC_QueryInterface(This,riid,ppvObject) \
    (This)->lpVtbl->QueryInterface(This,riid,ppvObject)

#define ITPCC_AddRef(This) \
    (This)->lpVtbl->AddRef(This)

#define ITPCC_Release(This) \
    (This)->lpVtbl->Release(This)

#define ITPCC_NewOrder(This,txn_in,txn_out) \
    (This)->lpVtbl->NewOrder(This,txn_in,txn_out)

#define ITPCC_Payment(This,txn_in,txn_out) \
    (This)->lpVtbl->Payment(This,txn_in,txn_out)

#define ITPCC_Delivery(This,txn_in) \

```

```

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

#define ITPCC_StockLevel(This,txn_in,txn_out) \
    (This)->lpVtbl->StockLevel(This,txn_in,txn_out)

#define ITPCC_OrderStatus(This,txn_in,txn_out) \
    (This)->lpVtbl->OrderStatus(This,txn_in,txn_out)

#define ITPCC_CallSetComplete(This) \
    (This)->lpVtbl->CallSetComplete(This)

#endif /* COBJMACROS */

#endif /* C style interface */

HRESULT STDMETHODCALLTYPE ITPCC_NewOrder_Proxy(
    ITPCC__RPC_FAR * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT __RPC_FAR *txn_out);

void __RPC_STUB ITPCC_NewOrder_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *pRpcChannelBuffer,
    PRPC_MESSAGE_pRpcMessage,
    DWORD *_pdwStubPhase);

HRESULT STDMETHODCALLTYPE ITPCC_Payment_Proxy(
    ITPCC__RPC_FAR * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT __RPC_FAR *txn_out);

void __RPC_STUB ITPCC_Payment_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *pRpcChannelBuffer,
    PRPC_MESSAGE_pRpcMessage,
    DWORD *_pdwStubPhase);

HRESULT STDMETHODCALLTYPE ITPCC_Delivery_Proxy(
    ITPCC__RPC_FAR * This,
    /* [in] */ VARIANT txn_in);

void __RPC_STUB ITPCC_Delivery_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *pRpcChannelBuffer,
    PRPC_MESSAGE_pRpcMessage,
    DWORD *_pdwStubPhase);

HRESULT STDMETHODCALLTYPE ITPCC_StockLevel_Proxy(
    ITPCC__RPC_FAR * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT __RPC_FAR *txn_out);

void __RPC_STUB ITPCC_StockLevel_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *pRpcChannelBuffer,
    PRPC_MESSAGE_pRpcMessage,
    DWORD *_pdwStubPhase);

```

```

HRESULT __stdcall ITPCC_OrderStatus_Proxy(
    ITPCC __RPC_FAR * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT __RPC_FAR *txn_out);

void __RPC_STUB ITPCC_OrderStatus_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *_pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,
    DWORD *_pdwStubPhase);

HRESULT __stdcall ITPCC_CallSetComplete_Proxy(
    ITPCC __RPC_FAR * This);

void __RPC_STUB ITPCC_CallSetComplete_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *_pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,
    DWORD *_pdwStubPhase);

#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 "oidl.idl";
import "ocidl.idl";

[
    object,
    oleautomation,
    uuid(FEEE6AA2-84B1-11d2-BA47-00C04FBFE08B),
    helpstring("ITPCC Interface"),
    pointer_default(unique)
]
interface ITPCC : IUnknown
{
    HRESULT __stdcall NewOrder
    (
    [in]
    VARIANT txn_in,
    [out] VARIANT *txn_out
    );
    HRESULT __stdcall Payment
    (
    [in]
    VARIANT txn_in,
    [out] VARIANT *txn_out
    );
    HRESULT __stdcall Delivery
    (
    [in]
    VARIANT txn_in
    );
    HRESULT __stdcall StockLevel
    (
    [in]
    VARIANT txn_in,
    [out] VARIANT *txn_out
    );
    HRESULT __stdcall OrderStatus
    (
    [in]
    VARIANT txn_in,
    [out] VARIANT *txn_out
    );
    HRESULT __stdcall CallSetComplete
    (
    );
}; // interface ITPCC

tpcc_com_ps.i.c

/* this file contains the actual definitions of */

```

```

/* the IIDs and CLSIDs */

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

/* File created by MIDL compiler version 5.01.0164 */
/* at Fri Jan 19 11:14:46 2001
*/
/* Compiler settings for .\src\tpcc_com_ps.idl:
Oicf (OptLev=i2), W1, Zp8, env=Win32, ms_ext, c_ext
error checks: allocation ref bounds_check enum stub_data
*/
//@@@MIDL_FILE_HEADING( )
#ifdef __cplusplus
extern "C"{
#endif

#ifdef __IID_DEFINED__
#define __IID_DEFINED__

typedef struct _IID
{
    unsigned long x;
    unsigned short s1;
    unsigned short s2;
    unsigned char c[8];
} IID;

#endif // __IID_DEFINED__

#ifdef CLSID_DEFINED
#define CLSID_DEFINED
typedef IID CLSID;
#endif // CLSID_DEFINED

const IID IID_ITPCC =
{0xFEEE6AA2,0x84B1,0x11d2,{0xBA,0x47,0x00,0xC0,0x4F,0xBF,0xE0,0x8B}};

#ifdef __cplusplus
}
#endif

tpcc_com_ps_p.c

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

/* File created by MIDL compiler version 5.01.0164 */
/* at Fri Jan 19 11:14:46 2001
*/
/* Compiler settings for .\src\tpcc_com_ps.idl:
Oicf (OptLev=i2), W1, Zp8, env=Win32, ms_ext, c_ext
error checks: allocation ref bounds_check enum stub_data
*/
//@@@MIDL_FILE_HEADING( )

#define USE_STUBLESS_PROXY

/* verify that the <rpcproxy.h> version is high enough to compile this
file*/
#ifdef __REDQ_RPCPROXY_H_VERSION__
#define __REQUIRED_RPCPROXY_H_VERSION__ 440

```

```

#endif

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

#include "tpcc_com_ps.h"

#define TYPE_FORMAT_STRING_SIZE 997
#define PROC_FORMAT_STRING_SIZE 187

typedef struct _MIDL_TYPE_FORMAT_STRING
{
    short Pad;
    unsigned char Format[ TYPE_FORMAT_STRING_SIZE];
} MIDL_TYPE_FORMAT_STRING;

typedef struct _MIDL_PROC_FORMAT_STRING
{
    short Pad;
    unsigned char Format[ PROC_FORMAT_STRING_SIZE];
} MIDL_PROC_FORMAT_STRING;

extern const MIDL_TYPE_FORMAT_STRING
__MIDL_TypeFormatString;
extern const MIDL_PROC_FORMAT_STRING
__MIDL_ProcFormatString;

/* Standard interface: __MIDL_itf_tpcc_com_ps_0000, ver. 0.0,
GUID={0x00000000,0x0000,0x0000,{0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00}} */

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

/* Object interface: ITPCC, ver. 0.0,
GUID={0xFEEE6AA2,0x84B1,0x11d2,{0xBA,0x47,0x00,0xC0,0x4F,0xBF,0xE0,0x8B}} */

extern const MIDL_STUB_DESC Object_StubDesc;

extern const MIDL_SERVER_INFO ITPCC_ServerInfo;

#pragma code_seg(".orpc")
extern const USER_MARSHAL_ROUTINE_QUADRUPLE
UserMarshalRoutines[1];

static const MIDL_STUB_DESC Object_StubDesc =
{
    0,
    NdrOleAllocate,
    NdrOleFree,
    0,
    0,
    0,
    0,
    0,
    0,

```

```

0,
0,
__MIDL_TypeFormatString.Format,
1, /* -error bounds_check flag */
0x20000, /* Ndr library version */
0,
0x50100a4, /* MIDL Version 5.1.164 */
0,
UserMarshalRoutines,
0, /* notify & notify_flag routine table */
1, /* Flags */
0, /* Reserved3 */
0, /* Reserved4 */
0 /* Reserved5 */
};

static const unsigned short ITPCC_FormatStringOffsetTable[] =
{
0,
34,
68,
96,
130,
164
};

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

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

CINTERFACE_PROXY_VTABLE(9)_ITPCCProxyVtbl =
{
&ITPCC_ProxyInfo,
&IID_ITPCC,
IUnknown_QueryInterface_Proxy,
IUnknown_AddRef_Proxy,
IUnknown_Release_Proxy,
(void *)-1 /* ITPCC::NewOrder */,
(void *)-1 /* ITPCC::Payment */,
(void *)-1 /* ITPCC::Delivery */,
(void *)-1 /* ITPCC::StockLevel */,
(void *)-1 /* ITPCC::OrderStatus */,
(void *)-1 /* ITPCC::CallSetComplete */
};

const CInterfaceStubVtbl_ITPCCStubVtbl =
{
&IID_ITPCC,
&ITPCC_ServerInfo,
9,
0, /* pure interpreted */
CStdStubBuffer_METHODS
};

#pragma data_seg(".rdata")

static const USER_MARSHAL_ROUTINE_QUADRUPLE
UserMarshalRoutines[1] =
{
{
VARIANT_UserSize
,VARIANT_UserMarshal
,VARIANT_UserUnmarshal
,VARIANT_UserFree
}
};

#if !defined(__RPC_WIN32__)
#error Invalid build platform for this stub.
#endif

#if !(TARGET_IS_NT40_OR_LATER)
#error You need a Windows NT 4.0 or later to run this stub because it
uses these features:
#error -Oif or -Oicf, [wire_marshall] or [user_marshall] attribute, more
than 32 methods in the interface.
#error However, your C/C++ compilation flags indicate you intend to
run this app on earlier systems.
#error This app will die there with the
RPC_X_WRONG_STUB_VERSION error.
#endif

static const MIDL_PROC_FORMAT_STRING
__MIDL_ProcFormatString =
{
0,
{
/* Procedure NewOrder */
FC_AUTO_HANDLE /* 0x33, /*
0x6c, /* Old Flags:
object, Oi2 */
/* 2 */ NdrFcLong( 0x0 ), /* 0 */
/* 6 */ NdrFcShort( 0x3 ), /* 3 */
#ifdef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 8 */ NdrFcShort( 0x1c ), /* x86 Stack size/offset = 28 */
#else
NdrFcShort( 0x20 ), /* MIPS &
PPC Stack size/offset = 32 */
#endif
#else
NdrFcShort( 0x28 ), /* Alpha Stack
size/offset = 40 */
#endif
/* 10 */ NdrFcShort( 0x0 ), /* 0 */
/* 12 */ NdrFcShort( 0x8 ), /* 8 */
/* 14 */ 0x7, /* Oi2 Flags: srv must size, clt must
size, has return, */
0x3, /* 3 */
/* Parameter txn_in */
}
}
};

```

```

/* 16 */  NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by
val, */
#ifdef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 18 */  NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
#else
NdrFcShort( 0x8 ), /* MIPS &
PPC Stack size/offset = 8 */
#endif
#else
NdrFcShort( 0x8 ), /* Alpha Stack
size/offset = 8 */
#endif
/* 20 */  NdrFcShort( 0x3c8 ), /* Type Offset=968 */

/* Parameter txn_out */

/* 22 */  NdrFcShort( 0x4113 ), /* Flags: must size, must
free, out, simple ref, srv alloc size=16 */
#ifdef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 24 */  NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
#else
NdrFcShort( 0x18 ), /* MIPS &
PPC Stack size/offset = 24 */
#endif
#else
NdrFcShort( 0x18 ), /* Alpha Stack
size/offset = 24 */
#endif
/* 26 */  NdrFcShort( 0x3da ), /* Type Offset=986 */

/* Return value */

/* 28 */  NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifdef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 30 */  NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
#else
NdrFcShort( 0x1c ), /* MIPS &
PPC Stack size/offset = 28 */
#endif
#else
NdrFcShort( 0x20 ), /* Alpha Stack
size/offset = 32 */
#endif
/* 32 */  0x8, /* FC_LONG */
0x0, /* 0 */

/* Procedure Payment */

/* 34 */  0x33, /* FC_AUTO_HANDLE */
0x6c, /* Old Flags:
object, Oi2 */
/* 36 */  NdrFcLong( 0x0 ), /* 0 */
/* 40 */  NdrFcShort( 0x4 ), /* 4 */
#ifdef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 42 */  NdrFcShort( 0x1c ), /* x86 Stack size/offset = 28 */
#else
NdrFcShort( 0x20 ), /* MIPS &
PPC Stack size/offset = 32 */
#endif
#else
NdrFcShort( 0x28 ), /* Alpha Stack
size/offset = 40 */
#endif

/* 44 */  NdrFcShort( 0x0 ), /* 0 */
/* 46 */  NdrFcShort( 0x8 ), /* 8 */
/* 48 */  0x7, /* Oi2 Flags: srv must size, clt must
size, has return, */
0x3, /* 3 */

/* Parameter txn_in */

/* 50 */  NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by
val, */
#ifdef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 52 */  NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
#else
NdrFcShort( 0x8 ), /* MIPS &
PPC Stack size/offset = 8 */
#endif
#else
NdrFcShort( 0x8 ), /* Alpha Stack
size/offset = 8 */
#endif
/* 54 */  NdrFcShort( 0x3c8 ), /* Type Offset=968 */

/* Parameter txn_out */

/* 56 */  NdrFcShort( 0x4113 ), /* Flags: must size, must
free, out, simple ref, srv alloc size=16 */
#ifdef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 58 */  NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
#else
NdrFcShort( 0x18 ), /* MIPS &
PPC Stack size/offset = 24 */
#endif
#else
NdrFcShort( 0x18 ), /* Alpha Stack
size/offset = 24 */
#endif
/* 60 */  NdrFcShort( 0x3da ), /* Type Offset=986 */

/* Return value */

/* 62 */  NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifdef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 64 */  NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
#else
NdrFcShort( 0x1c ), /* MIPS &
PPC Stack size/offset = 28 */
#endif
#else
NdrFcShort( 0x20 ), /* Alpha Stack
size/offset = 32 */
#endif
/* 66 */  0x8, /* FC_LONG */
0x0, /* 0 */

/* Procedure Delivery */

/* 68 */  0x33, /* FC_AUTO_HANDLE */
0x6c, /* Old Flags:
object, Oi2 */
/* 70 */  NdrFcLong( 0x0 ), /* 0 */
/* 74 */  NdrFcShort( 0x5 ), /* 5 */
#ifdef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 76 */  NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
#endif
#endif

```

```

PPC Stack size/offset = 28 */
#endif
#else
    NdrFcShort( 0x1c ), /* MIPS &
/* 110 */ 0x7, /* Oi2 Flags: srv must size, clt must
size, has return, */
    0x3, /* 3 */
    /* Parameter txn_in */
    /* 112 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by
val, */
#ifdef _ALPHA_
#ifndef _MIPS_ && !defined(_PPC_)
/* 114 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
#else
    NdrFcShort( 0x8 ), /* MIPS &
PPC Stack size/offset = 8 */
#endif
#else
    NdrFcShort( 0x8 ), /* Alpha Stack
size/offset = 8 */
#endif
/* 116 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */
/* Parameter txn_out */
/* 118 */ NdrFcShort( 0x4113 ), /* Flags: must size, must
free, out, simple ref, srv alloc size=16 */
#ifdef _ALPHA_
#ifndef _MIPS_ && !defined(_PPC_)
/* 120 */ NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
#else
    NdrFcShort( 0x18 ), /* MIPS &
PPC Stack size/offset = 24 */
#endif
#else
    NdrFcShort( 0x18 ), /* Alpha Stack
size/offset = 24 */
#endif
/* 122 */ NdrFcShort( 0x3da ), /* Type Offset=986 */
/* Return value */
/* 124 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifdef _ALPHA_
#ifndef _MIPS_ && !defined(_PPC_)
/* 126 */ NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
#else
    NdrFcShort( 0x1c ), /* MIPS &
PPC Stack size/offset = 28 */
#endif
#else
    NdrFcShort( 0x20 ), /* Alpha Stack
size/offset = 32 */
#endif
/* 128 */ 0x8, /* FC_LONG */
0x0, /* 0 */
/* Procedure OrderStatus */
/* 130 */ 0x33, /* FC_AUTO_HANDLE */
0x6c, /* Old Flags:
object, Oi2 */
/* 132 */ NdrFcLong( 0x0 ), /* 0 */
/* 136 */ NdrFcShort( 0x7 ), /* 7 */
#ifdef _ALPHA_
#ifndef _MIPS_ && !defined(_PPC_)
/* 138 */ NdrFcShort( 0x1c ), /* x86 Stack size/offset = 28 */
#else
    NdrFcShort( 0x20 ), /* MIPS &
PPC Stack size/offset = 32 */
#endif
#endif
#endif
#endif
NdrFcShort( 0x1c ), /* MIPS &
NdrFcShort( 0x20 ), /* Alpha Stack
size/offset = 32 */
#endif
/* 78 */ NdrFcShort( 0x0 ), /* 0 */
/* 80 */ NdrFcShort( 0x8 ), /* 8 */
/* 82 */ 0x6, /* Oi2 Flags: clt must size, has
return, */
0x2, /* 2 */
/* Parameter txn_in */
/* 84 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by
val, */
#ifdef _ALPHA_
#ifndef _MIPS_ && !defined(_PPC_)
/* 86 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
#else
    NdrFcShort( 0x8 ), /* MIPS &
PPC Stack size/offset = 8 */
#endif
#else
    NdrFcShort( 0x8 ), /* Alpha Stack
size/offset = 8 */
#endif
/* 88 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */
/* Return value */
/* 90 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifdef _ALPHA_
#ifndef _MIPS_ && !defined(_PPC_)
/* 92 */ NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
#else
    NdrFcShort( 0x18 ), /* MIPS &
PPC Stack size/offset = 24 */
#endif
#else
    NdrFcShort( 0x18 ), /* Alpha Stack
size/offset = 24 */
#endif
/* 94 */ 0x8, /* FC_LONG */
0x0, /* 0 */
/* Procedure StockLevel */
/* 96 */ 0x33, /* FC_AUTO_HANDLE */
0x6c, /* Old Flags:
object, Oi2 */
/* 98 */ NdrFcLong( 0x0 ), /* 0 */
/* 102 */ NdrFcShort( 0x6 ), /* 6 */
#ifdef _ALPHA_
#ifndef _MIPS_ && !defined(_PPC_)
/* 104 */ NdrFcShort( 0x1c ), /* x86 Stack size/offset = 28 */
#else
    NdrFcShort( 0x20 ), /* MIPS &
PPC Stack size/offset = 32 */
#endif
#else
    NdrFcShort( 0x28 ), /* Alpha Stack
size/offset = 40 */
#endif
/* 106 */ NdrFcShort( 0x0 ), /* 0 */
/* 108 */ NdrFcShort( 0x8 ), /* 8 */

```

```

#endif
#else
    NdrFcShort( 0x28 ), /* Alpha Stack
size/offset = 40 */
#endif
/* 140 */ NdrFcShort( 0x0 ), /* 0 */
/* 142 */ NdrFcShort( 0x8 ), /* 8 */
/* 144 */ 0x7, /* Oi2 Flags: srv must size, clt must
size, has return, */
    0x3, /* 3 */
    /* Parameter txn_in */
/* 146 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by
val, */
#ifndef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 148 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
#else
    NdrFcShort( 0x8 ), /* MIPS &
PPC Stack size/offset = 8 */
#endif
#else
    NdrFcShort( 0x8 ), /* Alpha Stack
size/offset = 8 */
#endif
/* 150 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */
    /* Parameter txn_out */
/* 152 */ NdrFcShort( 0x4113 ), /* Flags: must size, must
free, out, simple ref, srv alloc size=16 */
#ifndef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 154 */ NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
#else
    NdrFcShort( 0x18 ), /* MIPS &
PPC Stack size/offset = 24 */
#endif
#else
    NdrFcShort( 0x18 ), /* Alpha Stack
size/offset = 24 */
#endif
/* 156 */ NdrFcShort( 0x3da ), /* Type Offset=986 */
    /* Return value */
/* 158 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifndef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 160 */ NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
#else
    NdrFcShort( 0x1c ), /* MIPS &
PPC Stack size/offset = 28 */
#endif
#else
    NdrFcShort( 0x20 ), /* Alpha Stack
size/offset = 32 */
#endif
/* 162 */ 0x8, /* FC_LONG */
    0x0, /* 0 */
    /* Procedure CallSetComplete */
/* 164 */ 0x33, /* FC_AUTO_HANDLE */
    0x6c, /* Old Flags:
object, Oi2 */
/* 166 */ NdrFcLong( 0x0 ), /* 0 */
/* 170 */ NdrFcShort( 0x8 ), /* 8 */
#ifndef _ALPHA_
/* 172 */ NdrFcShort( 0x8 ), /* x86, MIPS, PPC Stack size/offset
= 8 */
#else
    NdrFcShort( 0x10 ), /* Alpha Stack
size/offset = 16 */
#endif
/* 174 */ NdrFcShort( 0x0 ), /* 0 */
/* 176 */ NdrFcShort( 0x8 ), /* 8 */
/* 178 */ 0x4, /* Oi2 Flags: has return, */
    0x1, /* 1 */
    /* Return value */
/* 180 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifndef _ALPHA_
/* 182 */ NdrFcShort( 0x4 ), /* x86, MIPS, PPC Stack size/offset
= 4 */
#else
    NdrFcShort( 0x8 ), /* Alpha Stack
size/offset = 8 */
#endif
/* 184 */ 0x8, /* FC_LONG */
    0x0, /* 0 */
    0x0
}
};
static const MIDL_TYPE_FORMAT_STRING
__MIDL_TypeFormatString =
{
    0,
    {
        NdrFcShort( 0x0 ), /* 0 */
        0x12, 0x0, /* FC_UP */
        NdrFcShort( 0x3b0 ), /* Offset= 944 (948) */
        0x2b, /*
FC_NON_ENCAPSULATED_UNION */
        0x9, /* FC_ULONG */
        0x7, /* Corr desc: FC_USHORT */
        0x0, /* */
        NdrFcShort( 0xffff ), /* -8 */
        NdrFcShort( 0x2 ), /* Offset= 2 (14) */
        NdrFcShort( 0x10 ), /* 16 */
        NdrFcShort( 0x2b ), /* 43 */
        NdrFcLong( 0x3 ), /* 3 */
        NdrFcShort( 0x8008 ), /* Simple arm type:
FC_LONG */
        NdrFcLong( 0x11 ), /* 17 */
        NdrFcShort( 0x8001 ), /* Simple arm type:
FC_BYTE */
        NdrFcLong( 0x2 ), /* 2 */
        NdrFcShort( 0x8006 ), /* Simple arm type:
FC_SHORT */
        NdrFcLong( 0x4 ), /* 4 */
        NdrFcShort( 0x800a ), /* Simple arm type:
FC_FLOAT */
        NdrFcLong( 0x5 ), /* 5 */
        NdrFcShort( 0x800c ), /* Simple arm type:
FC_DOUBLE */
        NdrFcLong( 0xb ), /* 11 */
        NdrFcShort( 0x8006 ), /* Simple arm type:
FC_SHORT */
    }
}

```

```

/* 54 */ NdrFcLong( 0xa ), /* 10 */
/* 58 */ NdrFcShort( 0x8008 ), /* Simple arm type:
FC_LONG */
/* 60 */ NdrFcLong( 0x6 ), /* 6 */
/* 64 */ NdrFcShort( 0xd6 ), /* Offset= 214 (278) */
/* 66 */ NdrFcLong( 0x7 ), /* 7 */
/* 70 */ NdrFcShort( 0x800c ), /* Simple arm type:
FC_DOUBLE */
/* 72 */ NdrFcLong( 0x8 ), /* 8 */
/* 76 */ NdrFcShort( 0xd0 ), /* Offset= 208 (284) */
/* 78 */ NdrFcLong( 0xd ), /* 13 */
/* 82 */ NdrFcShort( 0xe2 ), /* Offset= 226 (308) */
/* 84 */ NdrFcLong( 0x9 ), /* 9 */
/* 88 */ NdrFcShort( 0xee ), /* Offset= 238 (326) */
/* 90 */ NdrFcLong( 0x2000 ), /* 8192 */
/* 94 */ NdrFcShort( 0xfa ), /* Offset= 250 (344) */
/* 96 */ NdrFcLong( 0x24 ), /* 36 */
/* 100 */ NdrFcShort( 0x308 ), /* Offset= 776 (876) */
/* 102 */ NdrFcLong( 0x4024 ), /* 16420 */
/* 106 */ NdrFcShort( 0x302 ), /* Offset= 770 (876) */
/* 108 */ NdrFcLong( 0x4011 ), /* 16401 */
/* 112 */ NdrFcShort( 0x300 ), /* Offset= 768 (880) */
/* 114 */ NdrFcLong( 0x4002 ), /* 16386 */
/* 118 */ NdrFcShort( 0x2fe ), /* Offset= 766 (884) */
/* 120 */ NdrFcLong( 0x4003 ), /* 16387 */
/* 124 */ NdrFcShort( 0x2fc ), /* Offset= 764 (888) */
/* 126 */ NdrFcLong( 0x4004 ), /* 16388 */
/* 130 */ NdrFcShort( 0x2fa ), /* Offset= 762 (892) */
/* 132 */ NdrFcLong( 0x4005 ), /* 16389 */
/* 136 */ NdrFcShort( 0x2f8 ), /* Offset= 760 (896) */
/* 138 */ NdrFcLong( 0x400b ), /* 16395 */
/* 142 */ NdrFcShort( 0x2e6 ), /* Offset= 742 (884) */
/* 144 */ NdrFcLong( 0x400a ), /* 16394 */
/* 148 */ NdrFcShort( 0x2e4 ), /* Offset= 740 (888) */
/* 150 */ NdrFcLong( 0x4006 ), /* 16390 */
/* 154 */ NdrFcShort( 0x2ea ), /* Offset= 746 (900) */
/* 156 */ NdrFcLong( 0x4007 ), /* 16391 */
/* 160 */ NdrFcShort( 0x2e0 ), /* Offset= 736 (896) */
/* 162 */ NdrFcLong( 0x4008 ), /* 16392 */
/* 166 */ NdrFcShort( 0x2e2 ), /* Offset= 738 (904) */
/* 168 */ NdrFcLong( 0x400d ), /* 16397 */
/* 172 */ NdrFcShort( 0x2e0 ), /* Offset= 736 (908) */
/* 174 */ NdrFcLong( 0x4009 ), /* 16393 */
/* 178 */ NdrFcShort( 0x2de ), /* Offset= 734 (912) */
/* 180 */ NdrFcLong( 0x6000 ), /* 24576 */
/* 184 */ NdrFcShort( 0x2dc ), /* Offset= 732 (916) */
/* 186 */ NdrFcLong( 0x400c ), /* 16396 */
/* 190 */ NdrFcShort( 0x2da ), /* Offset= 730 (920) */
/* 192 */ NdrFcLong( 0x10 ), /* 16 */
/* 196 */ NdrFcShort( 0x8002 ), /* Simple arm type:
FC_CHAR */
/* 198 */ NdrFcLong( 0x12 ), /* 18 */
/* 202 */ NdrFcShort( 0x8006 ), /* Simple arm type:
FC_SHORT */
/* 204 */ NdrFcLong( 0x13 ), /* 19 */
/* 208 */ NdrFcShort( 0x8008 ), /* Simple arm type:
FC_LONG */
/* 210 */ NdrFcLong( 0x16 ), /* 22 */
/* 214 */ NdrFcShort( 0x8008 ), /* Simple arm type:
FC_LONG */
/* 216 */ NdrFcLong( 0x17 ), /* 23 */
/* 220 */ NdrFcShort( 0x8008 ), /* Simple arm type:
FC_LONG */
/* 222 */ NdrFcLong( 0xe ), /* 14 */
/* 226 */ NdrFcShort( 0x2be ), /* Offset= 702 (928) */
/* 228 */ NdrFcLong( 0x400e ), /* 16398 */
/* 232 */ NdrFcShort( 0x2c4 ), /* Offset= 708 (940) */
/* 234 */ NdrFcLong( 0x4010 ), /* 16400 */

/* 238 */ NdrFcShort( 0x2c2 ), /* Offset= 706 (944) */
/* 240 */ NdrFcLong( 0x4012 ), /* 16402 */
/* 244 */ NdrFcShort( 0x280 ), /* Offset= 640 (884) */
/* 246 */ NdrFcLong( 0x4013 ), /* 16403 */
/* 250 */ NdrFcShort( 0x27e ), /* Offset= 638 (888) */
/* 252 */ NdrFcLong( 0x4016 ), /* 16406 */
/* 256 */ NdrFcShort( 0x278 ), /* Offset= 632 (888) */
/* 258 */ NdrFcLong( 0x4017 ), /* 16407 */
/* 262 */ NdrFcShort( 0x272 ), /* Offset= 626 (888) */
/* 264 */ NdrFcLong( 0x0 ), /* 0 */
/* 268 */ NdrFcShort( 0x0 ), /* Offset= 0 (268) */
/* 270 */ NdrFcLong( 0x1 ), /* 1 */
/* 274 */ NdrFcShort( 0x0 ), /* Offset= 0 (274) */
/* 276 */ NdrFcShort( 0xffffffff ), /* Offset= -1 (275) */
/* 278 */

0x15, /*
FC_STRUCT */
0x7, /* 7 */
/* 280 */ NdrFcShort( 0x8 ), /* 8 */
/* 282 */ 0xb, /* FC_HYPER */
0x5b, /* FC_END */
/* 284 */
0x12, 0x0, /* FC_UP */
/* 286 */ NdrFcShort( 0xc ), /* Offset= 12 (298) */
/* 288 */
0x1b, /*
FC_CARRAY */
0x1, /* 1 */
/* 290 */ NdrFcShort( 0x2 ), /* 2 */
/* 292 */ 0x9, /* Corr desc: FC_ULONG */
0x0, /*
/* 294 */ NdrFcShort( 0xffc ), /* -4 */
/* 296 */ 0x6, /* FC_SHORT */
0x5b, /* FC_END */
/* 298 */
0x17, /*
FC_CSTRUCT */
0x3, /* 3 */
/* 300 */ NdrFcShort( 0x8 ), /* 8 */
/* 302 */ NdrFcShort( 0xffffffff2 ), /* Offset= -14 (288) */
/* 304 */ 0x8, /* FC_LONG */
0x8, /* FC_LONG
*/
/* 306 */ 0x5c, /* FC_PAD */
0x5b, /* FC_END */
/* 308 */
0x2f, /* FC_IP */
0x5a, /*
FC_CONSTANT_IID */
/* 310 */ NdrFcLong( 0x0 ), /* 0 */
/* 314 */ NdrFcShort( 0x0 ), /* 0 */
/* 316 */ NdrFcShort( 0x0 ), /* 0 */
/* 318 */ 0xc0, /* 192 */
0x0, /* 0 */
/* 320 */ 0x0, /* 0 */
0x0, /* 0 */
/* 322 */ 0x0, /* 0 */
0x0, /* 0 */
/* 324 */ 0x0, /* 0 */
0x46, /* 70 */
/* 326 */
0x2f, /* FC_IP */
0x5a, /*
FC_CONSTANT_IID */
/* 328 */ NdrFcLong( 0x20400 ), /* 132096 */
/* 332 */ NdrFcShort( 0x0 ), /* 0 */
/* 334 */ NdrFcShort( 0x0 ), /* 0 */
/* 336 */ 0xc0, /* 192 */

```


/* 338 */	0x0,	/* 0 */	/*	0x8,	/* FC_LONG
/* 340 */	0x0,	/* 0 */	/* 448 */	0x5c,	/* FC_PAD */
/* 342 */	0x0,	/* 0 */	/* 450 */	0x5b,	/* FC_END */
/* 344 */	0x46,	/* 70 */	FC_PSTRUCT */	0x16,	/*
/* 346 */	NdrFcShort(0x2),	/* FC_UP */	/* 452 */	0x3,	/* 3 */
/* 348 */	/* Offset= 2 (348) */	/*	/* 454 */	0x5c,	/* FC_PP */
/* 350 */	NdrFcShort(0x1fc),	/* FC_UP */	/* 456 */	0x4b,	/* FC_PAD */
/* 352 */	/* Offset= 508 (858) */	/*	FC_NO_REPEAT */	0x5c,	/*
FC_ENCAPSULATED_UNION */	0x2a,	/*	/* 458 */	0x5c,	/* FC_PAD */
/* 354 */	NdrFcShort(0x18),	/* 24 */	/* 460 */	NdrFcShort(0x4),	/* 4 */
/* 356 */	NdrFcShort(0xa),	/* 10 */	/* 462 */	NdrFcShort(0x4),	/* 4 */
/* 358 */	NdrFcLong(0x8),	/* 8 */	/* 464 */	0x11, 0x0, /* FC_RP */	/*
/* 362 */	NdrFcShort(0x58),	/* Offset= 88 (450) */	/* 466 */	NdrFcShort(0xffffd4),	/* Offset= -44 (420) */
/* 364 */	NdrFcLong(0xd),	/* 13 */	/*	0x5b,	/* FC_END */
/* 368 */	NdrFcShort(0x78),	/* Offset= 120 (488) */	/*	0x8,	/* FC_LONG
/* 370 */	NdrFcLong(0x9),	/* 9 */	/*	/*	/*
/* 374 */	NdrFcShort(0x94),	/* Offset= 148 (522) */	/* 468 */	0x8,	/* FC_LONG */
/* 376 */	NdrFcLong(0xc),	/* 12 */	/* 470 */	0x5b,	/* FC_END */
/* 380 */	NdrFcShort(0xbc),	/* Offset= 188 (568) */	FC_BOGUS_ARRAY */	0x21,	/*
/* 382 */	NdrFcLong(0x24),	/* 36 */	/* 472 */	0x3,	/* 3 */
/* 386 */	NdrFcShort(0x114),	/* Offset= 276 (662) */	/* 474 */	NdrFcShort(0x0),	/* 0 */
/* 388 */	NdrFcLong(0x800d),	/* 32781 */	/* 476 */	0x19,	/* Corr desc: field pointer,
/* 392 */	NdrFcShort(0x130),	/* Offset= 304 (696) */	FC_ULONG */	0x0,	/* */
/* 394 */	NdrFcLong(0x10),	/* 16 */	/* 478 */	NdrFcShort(0x0),	/* 0 */
/* 398 */	NdrFcShort(0x148),	/* Offset= 328 (726) */	/* 482 */	NdrFcLong(0xffffffff),	/* -1 */
/* 400 */	NdrFcLong(0x2),	/* 2 */	/* 484 */	0x4c,	/* FC_EMBEDDED_COMPLEX */
/* 404 */	NdrFcShort(0x160),	/* Offset= 352 (756) */	/* 486 */	0x0,	/* 0 */
/* 406 */	NdrFcLong(0x3),	/* 3 */	/* 488 */	NdrFcShort(0xffff50),	/* Offset= -176 (308) */
/* 410 */	NdrFcShort(0x178),	/* Offset= 376 (786) */	/*	0x5c,	/* FC_PAD */
/* 412 */	NdrFcLong(0x14),	/* 20 */	/*	0x5b,	/* FC_END */
/* 416 */	NdrFcShort(0x190),	/* Offset= 400 (816) */	FC_BOGUS_STRUCT */	0x1a,	/*
/* 418 */	NdrFcShort(0xffffffff),	/* Offset= -1 (417) */	/* 490 */	0x3,	/* 3 */
/* 420 */	0x1b,	/*	/* 492 */	NdrFcShort(0x8),	/* 8 */
FC_CARRAY */	0x3,	/* 3 */	/* 494 */	NdrFcShort(0x0),	/* 0 */
/* 422 */	NdrFcShort(0x4),	/* 4 */	/* 496 */	NdrFcShort(0x6),	/* Offset= 6 (500) */
/* 424 */	0x19,	/* Corr desc: field pointer,	/*	0x8,	/* FC_LONG */
FC_ULONG */	0x0,	/* */	FC_POINTER */	0x36,	/*
/* 426 */	NdrFcShort(0x0),	/* 0 */	/* 498 */	0x5c,	/* FC_PAD */
/* 428 */	0x4b,	/* FC_PP */	/* 500 */	0x5b,	/* FC_END */
/* 430 */	0x5c,	/* FC_PAD */	/* 502 */	0x11, 0x0, /* FC_RP */	/*
FC_VARIABLE_REPEAT */	0x48,	/*	/* 504 */	NdrFcShort(0xffffe0),	/* Offset= -32 (470) */
FC_FIXED_OFFSET */	0x49,	/*	/*	0x21,	/*
/* 432 */	NdrFcShort(0x4),	/* 4 */	FC_BOGUS_ARRAY */	0x3,	/* 3 */
/* 434 */	NdrFcShort(0x0),	/* 0 */	/* 506 */	NdrFcShort(0x0),	/* 0 */
/* 436 */	NdrFcShort(0x1),	/* 1 */	/* 508 */	0x19,	/* Corr desc: field pointer,
/* 438 */	NdrFcShort(0x0),	/* 0 */	FC_ULONG */	0x0,	/* */
/* 440 */	NdrFcShort(0x0),	/* 0 */	/* 510 */	NdrFcShort(0x0),	/* 0 */
/* 442 */	0x12, 0x0, /* FC_UP */	/*	/* 512 */	NdrFcLong(0xffffffff),	/* -1 */
/* 444 */	NdrFcShort(0xffff6e),	/* Offset= -146 (298) */	/* 514 */	0x4c,	/* FC_EMBEDDED_COMPLEX */
/* 446 */	0x5b,	/* FC_END */	/*	/*	/*

0x0, /* 0 */	FC_CONSTANT_IID */	0x5a, /*
/* 518 */ NdrFcShort(0xfffff40), /* Offset= -192 (326) */	/* 586 */ NdrFcLong(0x2f), /* 47 */	
/* 520 */ 0x5c, /* FC_PAD */	/* 590 */ NdrFcShort(0x0), /* 0 */	
/* 522 */	/* 592 */ NdrFcShort(0x0), /* 0 */	
0x1a, /*	/* 594 */ 0xc0, /* 192 */	
FC_BOGUS_STRUCT */	0x0, /* 0 */	
0x3, /* 3 */	/* 596 */ 0x0, /* 0 */	
/* 524 */ NdrFcShort(0x8), /* 8 */	/* 598 */ 0x0, /* 0 */	
/* 526 */ NdrFcShort(0x0), /* 0 */	/* 600 */ 0x0, /* 0 */	
/* 528 */ NdrFcShort(0x6), /* Offset= 6 (534) */	/* 602 */	
/* 530 */ 0x8, /* FC_LONG */	0x46, /* 70 */	
0x36, /*		
FC_POINTER */		
/* 532 */ 0x5c, /* FC_PAD */	0x1b, /*	
/* 534 */	FC_CARRY */	
0x11, 0x0, /* FC_RP */	0x0, /* 0 */	
/* 536 */ NdrFcShort(0xfffffe0), /* Offset= -32 (504) */	/* 604 */ NdrFcShort(0x1), /* 1 */	
/* 538 */	/* 606 */ 0x19, /* Corr desc: field pointer,	
0x1b, /*	FC_ULONG */	
FC_CARRY */	0x0, /*	
0x3, /* 3 */	/* 608 */ NdrFcShort(0x4), /* 4 */	
/* 540 */ NdrFcShort(0x4), /* 4 */	/* 610 */ 0x1, /* FC_BYTE */	
/* 542 */ 0x19, /* Corr desc: field pointer,	0x5b, /* FC_END */	
FC_ULONG */	/* 612 */	
0x0, /*	0x1a, /*	
/* 544 */ NdrFcShort(0x0), /* 0 */	FC_BOGUS_STRUCT */	
/* 546 */	0x3, /* 3 */	
0x4b, /* FC_PP */	/* 614 */ NdrFcShort(0x10), /* 16 */	
0x5c, /* FC_PAD */	/* 616 */ NdrFcShort(0x0), /* 0 */	
/* 548 */	/* 618 */ NdrFcShort(0xa), /* Offset= 10 (628) */	
0x48, /*	/* 620 */ 0x8, /* FC_LONG */	
FC_VARIABLE_REPEAT */	0x8, /* FC_LONG	
0x49, /*	*/	
FC_FIXED_OFFSET */	/* 622 */ 0x4c, /* FC_EMBEDDED_COMPLEX */	
/* 550 */ NdrFcShort(0x4), /* 4 */	0x0, /* 0 */	
/* 552 */ NdrFcShort(0x0), /* 0 */	/* 624 */ NdrFcShort(0xfffffd8), /* Offset= -40 (584) */	
/* 554 */ NdrFcShort(0x1), /* 1 */	/* 626 */ 0x36, /* FC_POINTER */	
/* 556 */ NdrFcShort(0x0), /* 0 */	0x5b, /* FC_END */	
/* 558 */ NdrFcShort(0x0), /* 0 */	/* 628 */	
/* 560 */ 0x12, 0x0, /* FC_UP */	0x12, 0x0, /* FC_UP */	
/* 562 */ NdrFcShort(0xffffdce), /* Offset= -562 (0) */	/* 630 */ NdrFcShort(0xfffffe4), /* Offset= -28 (602) */	
/* 564 */	/* 632 */	
0x5b, /* FC_END */	0x1b, /*	
0x8, /* FC_LONG	FC_CARRY */	
/*	0x3, /* 3 */	
/* 566 */ 0x5c, /* FC_PAD */	/* 634 */ NdrFcShort(0x4), /* 4 */	
/* 568 */	/* 636 */ 0x19, /* Corr desc: field pointer,	
0x1a, /*	FC_ULONG */	
FC_BOGUS_STRUCT */	0x0, /*	
0x3, /* 3 */	/* 638 */ NdrFcShort(0x0), /* 0 */	
/* 570 */ NdrFcShort(0x8), /* 8 */	/* 640 */	
/* 572 */ NdrFcShort(0x0), /* 0 */	0x4b, /* FC_PP */	
/* 574 */ NdrFcShort(0x6), /* Offset= 6 (580) */	0x5c, /* FC_PAD */	
/* 576 */ 0x8, /* FC_LONG */	0x48, /*	
0x36, /*	FC_VARIABLE_REPEAT */	
FC_POINTER */	0x49, /*	
/* 578 */ 0x5c, /* FC_PAD */	FC_FIXED_OFFSET */	
/* 580 */	/* 644 */ NdrFcShort(0x4), /* 4 */	
0x11, 0x0, /* FC_RP */	/* 646 */ NdrFcShort(0x0), /* 0 */	
/* 582 */ NdrFcShort(0xfffffd4), /* Offset= -44 (538) */	/* 648 */ NdrFcShort(0x1), /* 1 */	
/* 584 */	/* 650 */ NdrFcShort(0x0), /* 0 */	
0x2f, /* FC_IP */	/* 652 */ NdrFcShort(0x0), /* 0 */	
	/* 654 */ 0x12, 0x0, /* FC_UP */	
	/* 656 */ NdrFcShort(0xfffffd4), /* Offset= -44 (612) */	
	/* 658 */	
	0x5b, /* FC_END */	

0x8,	/* FC_LONG	/* 722 */ NdrFcShort(0x0),	/* 0 */
/* 660 */ 0x5c,	/* FC_PAD */	/* 724 */ 0x1,	/* FC_BYTE */
/* 662 */	0x5b,	/* 726 */	0x5b,
FC_BOGUS_STRUCT */	/* FC_END */	FC_PSTRUCT */	/* FC_END */
0x1a,	/*	0x16,	/*
0x3,	/* 3 */	0x3,	/* 3 */
/* 664 */ NdrFcShort(0x8),	/* 8 */	/* 728 */ NdrFcShort(0x8),	/* 8 */
/* 666 */ NdrFcShort(0x0),	/* 0 */	/* 730 */	0x4b,
/* 668 */ NdrFcShort(0x6),	/* Offset= 6 (674) */	/* 732 */	0x5c,
/* 670 */ 0x8,	/* FC_LONG */	/* 734 */	/* FC_PP */
FC_POINTER */	0x36,	FC_NO_REPEAT */	/* FC_PAD */
/* 672 */ 0x5c,	/* FC_PAD */	0x5c,	/* FC_PAD */
/* 674 */	0x5b,	/* 734 */ NdrFcShort(0x4),	/* 4 */
0x11, 0x0, /* FC_RP */	/* FC_END */	/* 736 */ NdrFcShort(0x4),	/* 4 */
/* 676 */ NdrFcShort(0xfffffd4),	/* Offset= -44 (632) */	/* 738 */ 0x12, 0x0, /* FC_UP */	/*
/* 678 */	0x1d,	/* 740 */ NdrFcShort(0xfffffe8),	/* Offset= -24 (716) */
FC_SMFARRAY */	/*	/* 742 */	0x5b,
0x0,	/* 0 */	0x8,	/* FC_END */
/* 680 */ NdrFcShort(0x8),	/* 8 */	/*	0x8,
/* 682 */ 0x2,	/* FC_CHAR */	/* 744 */ 0x8,	/* FC_LONG */
/* 684 */	0x5b,	/* 746 */	0x5b,
FC_STRUCT */	/* FC_END */	FC_CARRAY */	/* FC_END */
0x15,	/*	0x1b,	/*
/* 686 */ NdrFcShort(0x10),	/* 3 */	0x1,	/* 1 */
/* 688 */ 0x8,	/* 16 */	/* 748 */ NdrFcShort(0x2),	/* 2 */
0x6,	/* FC_SHORT	/* 750 */ 0x19,	/* Corr desc: field pointer,
/*	/* FC_SHORT */	FC_ULONG */	/*
/* 690 */ 0x6,	/* FC_SHORT */	0x0,	/* *
FC_EMBEDDED_COMPLEX */	/*	/* 752 */ NdrFcShort(0x0),	/* 0 */
/* 692 */ 0x0,	/* 0 */	/* 754 */ 0x6,	/* FC_SHORT */
Offset= -15 (678) */	NdrFcShort(0xffffffff),	/* 756 */	0x5b,
/* 696 */	/* FC_END */	FC_PSTRUCT */	/* FC_END */
0x1a,	/*	0x16,	/*
FC_BOGUS_STRUCT */	0x5b,	/* 758 */ NdrFcShort(0x8),	/* 3 */
0x3,	/* 3 */	/* 760 */	0x3,
/* 698 */ NdrFcShort(0x18),	/* 24 */	/* 762 */	0x4b,
/* 700 */ NdrFcShort(0x0),	/* 0 */	/* 764 */ NdrFcShort(0x4),	/* FC_PP */
/* 702 */ NdrFcShort(0xa),	/* Offset= 10 (712) */	/* 766 */ NdrFcShort(0x4),	/* FC_PAD */
/* 704 */ 0x8,	/* FC_LONG */	/* 768 */ 0x12, 0x0, /* FC_UP */	/*
FC_POINTER */	0x36,	/* 770 */ NdrFcShort(0xfffffe8),	/* Offset= -24 (746) */
/* 706 */ 0x4c,	/* FC_EMBEDDED_COMPLEX */	/* 772 */	0x5b,
/* 708 */ NdrFcShort(0xfffffe8),	/* 0 */	0x8,	/* FC_END */
/* 710 */ 0x5c,	/* FC_PAD */	/*	0x8,
/* 712 */	0x5b,	/* 774 */ 0x8,	/* FC_LONG */
0x11, 0x0, /* FC_RP */	/* FC_END */	/* 776 */	0x5b,
/* 714 */ NdrFcShort(0xfffff0c),	/* Offset= -244 (470) */	FC_CARRAY */	/* FC_END */
/* 716 */	0x1b,	0x1b,	/*
FC_CARRAY */	/*	FC_CARRAY */	/*
0x0,	/* 0 */	0x3,	/* 3 */
/* 718 */ NdrFcShort(0x1),	/* 1 */	/* 778 */ NdrFcShort(0x4),	/* 4 */
/* 720 */ 0x19,	/* Corr desc: field pointer,	/* 780 */ 0x19,	/* Corr desc: field pointer,
FC_ULONG */	/* *	FC_ULONG */	/* *
0x0,	/*	0x0,	/*

```

/* 782 */ NdrFcShort( 0x0 ), /* 0 */
/* 784 */ 0x8, /* FC_LONG */
/* 786 */
/* 792 */
FC_NO_REPEAT */
/* 794 */ NdrFcShort( 0x4 ), /* 4 */
/* 796 */ NdrFcShort( 0x4 ), /* 4 */
/* 798 */ 0x12, 0x0, /* FC_UP */
/* 800 */ NdrFcShort( 0xfffffe8 ), /* Offset= -24 (776) */
/* 802 */
/* 804 */ 0x8, /* FC_LONG */
/* 806 */
FC_CARRAY */
/* 808 */ NdrFcShort( 0x8 ), /* 8 */
/* 810 */ 0x19, /* Corr desc: field pointer,
FC_ULONG */
/* 812 */ NdrFcShort( 0x0 ), /* 0 */
/* 814 */ 0xb, /* FC_HYPER */
/* 816 */
FC_PSTRUCT */
/* 818 */ NdrFcShort( 0x8 ), /* 8 */
/* 820 */
/* 822 */
FC_NO_REPEAT */
/* 824 */ NdrFcShort( 0x4 ), /* 4 */
/* 826 */ NdrFcShort( 0x4 ), /* 4 */
/* 828 */ 0x12, 0x0, /* FC_UP */
/* 830 */ NdrFcShort( 0xfffffe8 ), /* Offset= -24 (806) */
/* 832 */
/* 834 */ 0x8, /* FC_LONG */
/* 836 */
FC_STRUCT */
/* 838 */ NdrFcShort( 0x8 ), /* 8 */
/* 840 */ 0x8, /* FC_LONG */
/*
/* 842 */ 0x5c, /* FC_PAD */
/* 844 */
FC_CARRAY */
/* 846 */ NdrFcShort( 0x8 ), /* 8 */
/* 848 */ 0x7, /* Corr desc: FC_USHORT */
/* 850 */ NdrFcShort( 0xffd8 ), /* -40 */
/* 852 */ 0x4c, /* FC_EMBEDDED_COMPLEX */
/* 854 */ NdrFcShort( 0xfffffee ), /* Offset= -18 (836) */
/* 856 */ 0x5c, /* FC_PAD */
/* 858 */
FC_BOGUS_STRUCT */
/* 860 */ NdrFcShort( 0x28 ), /* 40 */
/* 862 */ NdrFcShort( 0xfffffee ), /* Offset= -18 (844) */
/* 864 */ NdrFcShort( 0x0 ), /* Offset= 0 (864) */
/* 866 */ 0x6, /* FC_SHORT */
/* 868 */ 0x38, /* FC_ALIGNM4 */
/* 870 */ 0x8, /* FC_LONG */
FC_EMBEDDED_COMPLEX */
/* 872 */ 0x0, /* 0 */
Offset= -521 (352) */
/* 876 */
/* 878 */ NdrFcShort( 0xfffffe6 ), /* Offset= -266 (612) */
/* 880 */
[simple_pointer] */
/* 882 */ 0x1, /* FC_BYTE */
/* 884 */
[simple_pointer] */
/* 886 */ 0x6, /* FC_SHORT */
/* 888 */
[simple_pointer] */
/* 890 */ 0x8, /* FC_LONG */
/* 892 */
[simple_pointer] */
/* 894 */ 0xa, /* FC_FLOAT */
/* 896 */
[simple_pointer] */
/* 898 */ 0xc, /* FC_DOUBLE */
/* 900 */
/* 902 */ NdrFcShort( 0xffffd90 ), /* Offset= -624 (278) */
/* 904 */
/*
0x12, 0x10, /* FC_UP */

```

```

/* 906 */ NdrFcShort( 0xffffd92 ), /* Offset= -622 (284) */
/* 908 */
                                0x12, 0x10, /* FC_UP */
/* 910 */ NdrFcShort( 0xffffda6 ), /* Offset= -602 (308) */
/* 912 */
                                0x12, 0x10, /* FC_UP */
/* 914 */ NdrFcShort( 0xffffdb4 ), /* Offset= -588 (326) */
/* 916 */
                                0x12, 0x10, /* FC_UP */
/* 918 */ NdrFcShort( 0xffffdc2 ), /* Offset= -574 (344) */
/* 920 */
                                0x12, 0x10, /* FC_UP */
/* 922 */ NdrFcShort( 0x2 ), /* Offset= 2 (924) */
/* 924 */
                                0x12, 0x0, /* FC_UP */
/* 926 */ NdrFcShort( 0xffffc62 ), /* Offset= -926 (0) */
/* 928 */
                                0x15, /*
FC_STRUCT */
                                0x7, /* 7 */
/* 930 */ NdrFcShort( 0x10 ), /* 16 */
/* 932 */ 0x6, /* FC_SHORT */
                                0x1, /* FC_BYTE
*/
/* 934 */ 0x1, /* FC_BYTE */
FC_ALIGNM4 */
/* 936 */ 0x8, /* FC_LONG */
                                0x39, /*
FC_ALIGNM8 */
/* 938 */ 0xb, /* FC_HYPER */
                                0x5b, /* FC_END */
/* 940 */
                                0x12, 0x0, /* FC_UP */
/* 942 */ NdrFcShort( 0xfffff2 ), /* Offset= -14 (928) */
/* 944 */
                                0x12, 0x8, /* FC_UP
[simple_pointer] */
/* 946 */ 0x2, /* FC_CHAR */
                                0x5c, /* FC_PAD */
/* 948 */
                                0x1a, /*
FC_BOGUS_STRUCT */
                                0x7, /* 7 */
/* 950 */ NdrFcShort( 0x20 ), /* 32 */
/* 952 */ NdrFcShort( 0x0 ), /* 0 */
/* 954 */ NdrFcShort( 0x0 ), /* Offset= 0 (954) */
/* 956 */ 0x8, /* FC_LONG */
                                0x8, /* FC_LONG
*/
/* 958 */ 0x6, /* FC_SHORT */
                                0x6, /* FC_SHORT
*/
/* 960 */ 0x6, /* FC_SHORT */
                                0x6, /* FC_SHORT
*/
/* 962 */ 0x4c, /* FC_EMBEDDED_COMPLEX */
                                0x0, /* 0 */
/* 964 */ NdrFcShort( 0xffffc42 ), /* Offset= -958 (6) */
/* 966 */ 0x5c, /* FC_PAD */
                                0x5b, /* FC_END */
/* 968 */ 0xb4, /* FC_USER_MARSHAL */
                                0x83, /* 131 */
/* 970 */ NdrFcShort( 0x0 ), /* 0 */
/* 972 */ NdrFcShort( 0x10 ), /* 16 */
/* 974 */ NdrFcShort( 0x0 ), /* 0 */
/* 976 */ NdrFcShort( 0xffffc32 ), /* Offset= -974 (2) */
/* 978 */
                                0x11, 0x4, /* FC_RP
[allocated_on_stack] */
/* 980 */ NdrFcShort( 0x6 ), /* Offset= 6 (986) */
/* 982 */
                                0x13, 0x0, /* FC_OP */
/* 984 */ NdrFcShort( 0xfffff4c ), /* Offset= -36 (948) */
/* 986 */ 0xb4, /* FC_USER_MARSHAL */
                                0x83, /* 131 */
/* 988 */ NdrFcShort( 0x0 ), /* 0 */
/* 990 */ NdrFcShort( 0x10 ), /* 16 */
/* 992 */ NdrFcShort( 0x0 ), /* 0 */
/* 994 */ NdrFcShort( 0xfffff4 ), /* Offset= -12 (982) */
                                0x0
}
};

const CInterfaceProxyVtbl * _tpcc_com_ps_ProxyVtblList[] =
{
( CInterfaceProxyVtbl *) &_ITPCCProxyVtbl,
0
};

const CInterfaceStubVtbl * _tpcc_com_ps_StubVtblList[] =
{
( CInterfaceStubVtbl *) &_ITPCCStubVtbl,
0
};

PCInterfaceName const _tpcc_com_ps_InterfaceNamesList[] =
{
"ITPCC",
0
};

#define _tpcc_com_ps_CHECK_IID(n)
IID_GENERIC_CHECK_IID(_tpcc_com_ps, pIID, n)

int __stdcall _tpcc_com_ps_IID_Lookup( const IID * pIID, int *
pIndex )
{
if(!_tpcc_com_ps_CHECK_IID(0))
{
*pIndex = 0;
return 1;
}

return 0;
}

const ExtendedProxyFileInfo tpcc_com_ps_ProxyFileInfo =
{
(PCInterfaceProxyVtblList *) &_tpcc_com_ps_ProxyVtblList,
(PCInterfaceStubVtblList *) &_tpcc_com_ps_StubVtblList,
(const PCInterfaceName *) &_tpcc_com_ps_InterfaceNamesList,
0, // no delegation
&_tpcc_com_ps_IID_Lookup,
1,
2,
0, /* table of [async_uuid] interfaces */
0, /* Filler1 */
0, /* Filler2 */
0 /* Filler3 */
};

```

tpcc_com_rem.cpp

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

// needed for CoInitializeEx
#define _WIN32_WINNT 0x0400

#include <windows.h>

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

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

#include "..\..\tpcc_com_remote_ps\src\tpcc_com_remote_ps_i.c"
#include "..\..\tpcc_com_remote\src\tpcc_com_remote_i.c"

// wrapper routine for class constructor
__declspec(dllexport) CTPCC_COM_REM*
CTPCC_COM_REM_new(BOOL bSinglePool)
{
    return new CTPCC_COM_REM(bSinglePool);
}

CTPCC_COM_REM::CTPCC_COM_REM(BOOL bSinglePool)
{
    HRESULT hr = NULL;
    long lRet = 0;
    ULONG ulTmpSize = 0;

    m_bSinglePool = bSinglePool;

    m_pNewOrder = NULL;
    m_pPayment = NULL;
    m_pStockLevel = NULL;
    m_pOrderStatus = NULL;
}

/*
(COM_DATA*)CoTaskMemAlloc(sizeof(COM_DATA));
if (!m_pTxn)
    throw new CCOMERR( E_FAIL );
*/

ulTmpSize = (ULONG) sizeof(COM_DATA);
VariantInit(&m_vTxn);
m_vTxn.vt = VT_SAFEARRAY;

m_vTxn.parray = SafeArrayCreateVector(
```

```
VT_UI1,
ulTmpSize,
ulTmpSize);
if (!m_vTxn.parray)
    throw new CCOMERR( E_FAIL );

memset((void*)m_vTxn.parray->pvData,0,ulTmpSize);

m_pTxn = (COM_DATA*)m_vTxn.parray->pvData;

hr = CoInitializeEx(NULL, COINIT_MULTITHREADED);
if (FAILED(hr))
{
    throw new CCOMERR( hr );
}

// create components
if (m_bSinglePool)
{
    hr = CoCreateInstance(CLSID_TPCCrem,
NULL, CLSCTX_SERVER, IID_ITPCCrem, (void
***)&m_pNewOrder);
if (FAILED(hr))
    throw new CCOMERR(hr);

// all txns will use same component
m_pPayment = m_pNewOrder;
m_pStockLevel = m_pNewOrder;
m_pOrderStatus = m_pNewOrder;
}
else
{
    // use different components for each txn

    hr = CoCreateInstance(CLSID_NewOrder,
NULL, CLSCTX_SERVER, IID_ITPCCrem, (void
***)&m_pNewOrder);
if (FAILED(hr))
    throw new CCOMERR(hr);

    hr = CoCreateInstance(CLSID_Payment, NULL,
CLSCTX_SERVER, IID_ITPCCrem, (void ***)&m_pPayment);
if (FAILED(hr))
    throw new CCOMERR(hr);

    hr = CoCreateInstance(CLSID_StockLevel,
NULL, CLSCTX_SERVER, IID_ITPCCrem, (void
***)&m_pStockLevel);
if (FAILED(hr))
    throw new CCOMERR(hr);

    hr = CoCreateInstance(CLSID_OrderStatus,
NULL, CLSCTX_SERVER, IID_ITPCCrem, (void
***)&m_pOrderStatus);
if (FAILED(hr))
    throw new CCOMERR(hr);
}

// call setcomplete to release each component back into pool
hr = m_pNewOrder->CallSetComplete();
if (FAILED(hr))
    throw new CCOMERR(hr);

if (!m_bSinglePool)
{
```

```

        hr = m_pPayment->CallSetComplete();
        if (FAILED(hr))
            throw new CCOMERR(hr);

        hr = m_pStockLevel->CallSetComplete();
        if (FAILED(hr))
            throw new CCOMERR(hr);

        hr = m_pOrderStatus->CallSetComplete();
        if (FAILED(hr))
            throw new CCOMERR(hr);
    }
}

CTPCC_COM_REM::~~CTPCC_COM_REM()
{
    if (m_pTxn)
        SafeArrayDestroy(m_vTxn.parray);

    ReleaseInterface(m_pNewOrder);
    if (!m_bSinglePool)
    {
        ReleaseInterface(m_pPayment);
        ReleaseInterface(m_pStockLevel);
        ReleaseInterface(m_pOrderStatus);
    }
    CoUninitialize();
}

void CTPCC_COM_REM::NewOrder()
{
    VARIANT vTxn_out;

    HRESULT hr = m_pNewOrder->NewOrder(m_vTxn,
&vTxn_out);
    if (FAILED(hr))
        throw new CCOMERR( hr );
    memcpy(m_pTxn, (void
*)vTxn_out.parray->pvData,vTxn_out.parray->rgsabound[0].cElement
s);
    SafeArrayDestroy(vTxn_out.parray);

    if ( m_pTxn->ErrorType != ERR_SUCCESS )
        throw new CCOMERR( m_pTxn->ErrorType,
m_pTxn->error );
}

void CTPCC_COM_REM::Payment()
{
    VARIANT vTxn_out;

    HRESULT hr = m_pPayment->Payment(m_vTxn,
&vTxn_out);
    if (FAILED(hr))
        throw new CCOMERR( hr );
    memcpy(m_pTxn, (void
*)vTxn_out.parray->pvData,vTxn_out.parray->rgsabound[0].cElement
s);
    SafeArrayDestroy(vTxn_out.parray);

    if ( m_pTxn->ErrorType != ERR_SUCCESS )
        throw new CCOMERR( m_pTxn->ErrorType,
m_pTxn->error );
}

void CTPCC_COM_REM::StockLevel()
{

```

```

    VARIANT vTxn_out;

    HRESULT hr = m_pStockLevel->StockLevel(m_vTxn,
&vTxn_out);
    if (FAILED(hr))
        throw new CCOMERR( hr );
    memcpy(m_pTxn, (void
*)vTxn_out.parray->pvData,vTxn_out.parray->rgsabound[0].cElement
s);
    SafeArrayDestroy(vTxn_out.parray);

    if ( m_pTxn->ErrorType != ERR_SUCCESS )
        throw new CCOMERR( m_pTxn->ErrorType,
m_pTxn->error );
}

void CTPCC_COM_REM::OrderStatus()
{
    VARIANT vTxn_out;

    HRESULT hr = m_pOrderStatus->OrderStatus(m_vTxn,
&vTxn_out);
    if (FAILED(hr))
        throw new CCOMERR( hr );
    memcpy(m_pTxn, (void
*)vTxn_out.parray->pvData,vTxn_out.parray->rgsabound[0].cElement
s);
    SafeArrayDestroy(vTxn_out.parray);

    if ( m_pTxn->ErrorType != ERR_SUCCESS )
        throw new CCOMERR( m_pTxn->ErrorType,
m_pTxn->error );
}

```

tpcc_com_rem.h

```

/*      FILE:          TPCC_COM.H
*                               Microsoft TPC-C Kit Ver.
4.20.000
*                               Copyright Microsoft,
1999
*                               All Rights Reserved
*                               not yet audited
*
*      PURPOSE:       Header file for TPC-C COM+ class
implementation.
*
*      Change history:
*                               4.20.000 - first version
*/

#pragma once

#include <stdio.h>
#include "..\..\tpcc_com_ps\src\tpcc_com_ps.h"
#include "..\..\tpcc_com_remote_ps\src\tpcc_com_remote_ps.h"

// need to declare functions for import, unless define has already been
created
// by the DLL's .cpp module for export.
#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 OrderStatus;
        } u;
    } *m_pTxn;

    VARIANT m_vTxn;

public:
    CTPCC_COM(BOOL bSinglePool);
    ~CTPCC_COM(void);

    inline PNEW_ORDER_DATA
    BuffAddr_NewOrder() { return
&m_pTxn->u.NewOrder; };
    inline PPAYMENT_DATA
    BuffAddr_Payment() { return &m_pTxn->u.Payment; };
    inline PDELIVERY_DATA
    BuffAddr_Delivery() { return &m_pTxn->u.Delivery; };
    inline PSTOCK_LEVEL_DATA
    BuffAddr_StockLevel() { return &m_pTxn->u.StockLevel; };
    inline PORDER_STATUS_DATA
    BuffAddr_OrderStatus() { return &m_pTxn->u.OrderStatus;
};

    void NewOrder ();
    void Payment ();
    void StockLevel ();
    void OrderStatus ();
    void Delivery () { throw new
CCOMERR(E_NOTIMPL); } // not supported
};

inline void ReleaseInterface(IUnknown *pUnk)
{
    if (pUnk)
    {
        pUnk->Release();
        pUnk = NULL;
    }
}

// wrapper routine for class constructor
extern "C" __declspec(dllexport) CTPCC_COM*
CTPCC_COM_new(BOOL);

typedef CTPCC_COM* (TYPE_CTPCC_COM)(BOOL);

class DllDecl CTPCC_COM_REM : public CTPCC_BASE
{
private:
    BOOL m_bSinglePool;

    // COM Interface pointers
    ITPCCrem* m_pNewOrder;

```



```

m_pPayment;           ITPCCrem*
m_pStockLevel;       ITPCCrem*
m_pOrderStatus;      ITPCCrem*

struct COM_DATA
{
    int ErrorType;
    int error;
    union
    {
        NEW_ORDER_DATA
            NewOrder;
        PAYMENT_DATA
            Payment;
        DELIVERY_DATA
            Delivery;
        STOCK_LEVEL_DATA
            StockLevel;
    } u;
    ORDER_STATUS_DATA OrderStatus;
    *m_pTxn;
    VARIANT m_vTxn;
public:
    CTPCC_COM_REM(BOOL bSinglePool);
    ~CTPCC_COM_REM(void);

    inline PNEW_ORDER_DATA
    BuffAddr_NewOrder() { return
    &m_pTxn->u.NewOrder; };
    inline PPAYMENT_DATA
    BuffAddr_Payment() { return &m_pTxn->u.Payment; };
    inline PDELIVERY_DATA
    BuffAddr_Delivery() { return &m_pTxn->u.Delivery; };
    inline PSTOCK_LEVEL_DATA
    BuffAddr_StockLevel() { return &m_pTxn->u.StockLevel; };
    inline PORDER_STATUS_DATA
    BuffAddr_OrderStatus() { return &m_pTxn->u.OrderStatus;
};

    void NewOrder          ();
    void Payment           ();
    void StockLevel        ();
    void OrderStatus       ();
    void Delivery          () { throw new
CCOMERR(E_NOTIMPL); } // not supported
};

// wrapper routine for class constructor
extern "C" __declspec(dllexport) CTPCC_COM_REM*
CTPCC_COM_REMOTE_new(BOOL);

typedef CTPCC_COM_REM* (TYPE_CTPCC_COM_REM)(BOOL);

tpcc_com_remote.cpp

/* FILE: TPCC_COM_REMOTE.CPP
* 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 <atbase.h>
//You may derive a class from CComModule and use it if you want to
override
//something, but do not change the name of _Module
extern CComModule _Module;

#include <atlcom.h>
#include <initguid.h>
#include <transact.h>
#include <atimpl.cpp>
#include <comsvcs.h>

#include <sqltypes.h>
#include <sql.h>
#include <sqlext.h>

#include "tpcc_com_remote_ps.h"
#include "..\..\common\src\trans.h"
//tpckit transaction header contains definations of structures
specific to TPC-C
#include "..\..\common\src\txn_base.h"
#include "..\..\common\src\error.h"
#include "..\..\common\src\ReadRegistry.h"
#include "..\..\db_dblib_dll\src\tpcc_dblib.h" //
DBLIB implementation of TPC-C txns
#include "..\..\db_odbc_dll\src\tpcc_odbc.h" // ODBC
implementation of TPC-C txns

#include "resource.h"
#include "tpcc_com_remote.h"
#include "tpcc_com_remote_i.c"
#include "Methods.h"
#include "..\..\tpcc_com_remote_ps\src\tpcc_com_remote_ps_i.c"
#include "..\..\common\src\ReadRegistry.cpp"

CComModule _Module;

BEGIN_OBJECT_MAP(ObjectMap)
OBJECT_ENTRY(CLSID_TPCCrem, CTPCCrem)
END_OBJECT_MAP()

// configuration settings from registry
TPCCREGISTRYDATA Reg;
char
szMyComputerName[MAX_COMPUTERNAME_LENGTH+1];

```

```

static HINSTANCE hLibInstanceDb = NULL;

TYPE_CTPCC_DBLIB      *pCTPCC_DBLIB_new;
TYPE_CTPCC_ODBC       *pCTPCC_ODBC_new;

////////////////////////////////////
// DLL Entry Point

extern "C"
BOOL WINAPI DllMain(HINSTANCE hInstance, DWORD
dwReason, LPVOID /**lpReserved*/)
{
    char szDllName[128];

    try
    {
        if (dwReason == DLL_PROCESS_ATTACH)
        {
            _Module.Init(ObjectMap, hInstance);

DisableThreadLibraryCalls(hInstance);

            DWORD dwSize =
MAX_COMPUTERNAME_LENGTH+1;

GetComputerName(szMyComputerName, &dwSize);
            szMyComputerName[dwSize] = 0;

            if ( ReadTPCCRegistrySettings(
&Reg ) )
                throw new
CCOMPONENT_ERR( ERR_MISSING_REGISTRY_ENTRIES);

            if (Reg.eDB_Protocol == DBLIB)
            {
                strcpy( szDllName,
Reg.szPath );
                strcat( szDllName,
"tpcc_dblib.dll");
                hLibInstanceDb =
LoadLibrary( szDllName );
                if (hLibInstanceDb ==
NULL)
                    throw new
CCOMPONENT_ERR( ERR_LOADDLL_FAILED,szDllName,
GetLastError());

                // get function pointer to
wrapper for class constructor
                pCTPCC_DBLIB_new =
(TYPE_CTPCC_DBLIB*)
GetProcAddress(hLibInstanceDb,"CTPCC_DBLIB_new");
                if (pCTPCC_DBLIB_new
== NULL)
                    throw new
CCOMPONENT_ERR( ERR_GETPROCADDR_FAILED,
szDllName, GetLastError());

                ODBC)
            else if (Reg.eDB_Protocol ==
{
                strcpy( szDllName,
Reg.szPath );
                strcat( szDllName,
"tpcc_odbc.dll");

```

```

                hLibInstanceDb =
                if (hLibInstanceDb ==
                throw new
CCOMPONENT_ERR( ERR_LOADDLL_FAILED,szDllName,
GetLastError());

                // get function pointer to
wrapper for class constructor
                pCTPCC_ODBC_new =
(TYPE_CTPCC_ODBC*)
GetProcAddress(hLibInstanceDb,"CTPCC_ODBC_new");
                if (pCTPCC_ODBC_new
== NULL)
                    throw new
CCOMPONENT_ERR( ERR_GETPROCADDR_FAILED,
szDllName, GetLastError());
            }
            else
                throw new
CCOMPONENT_ERR( ERR_UNKNOWN_DB_PROTOCOL);
        }
        else if (dwReason ==
DLL_PROCESS_DETACH)
            _Module.Term();
    }
    catch (CBaseErr *e)
    {
        WriteMessageToEventLog(e->ErrorText());
        delete e;
        return FALSE;
    }
    catch (...)
    {
        WriteMessageToEventLog(TEXT("Unhandled
exception in object DllMain"));
        return FALSE;
    }
    return TRUE;    // OK
}

////////////////////////////////////
// Used to determine whether the DLL can be unloaded by OLE

STDAPI DllCanUnloadNow(void)
{
    return (_Module.GetLockCount()==0) ? S_OK : S_FALSE;
}

////////////////////////////////////
// Returns a class factory to create an object of the requested type

STDAPI DllGetClassObject(REFCLSID rclsid, REFIID riid, LPVOID*
ppv)
{
    return _Module.GetClassObject(rclsid, riid, ppv);
}

////////////////////////////////////
// DllRegisterServer - Adds entries to the system registry

STDAPI DllRegisterServer(void)
{

```

```

// registers object, typelib and all interfaces in typelib
return _Module.RegisterServer(TRUE);
}

////////////////////////////////////
// DllUnregisterServer - Removes entries from the system registry

STDAPI DllUnregisterServer(void)
{
    _Module.UnregisterServer();
    return S_OK;
}

static void WriteMessageToEventLog(LPTSTR lpszMsg)
{
    TCHAR szMsg[256];
    HANDLE hEventSource;
    LPTSTR lpszStrings[2];

    // Use event logging to log the error.
    //
    hEventSource = RegisterEventSource(NULL,
    TEXT("tpcc_com_remote.dll"));

    _stprintf(szMsg, TEXT("Error in COM+ TPC-C Component: "));
    lpszStrings[0] = szMsg;
    lpszStrings[1] = lpszMsg;

    if (hEventSource != NULL)
    {
        ReportEvent(hEventSource, // handle of event source
        EVENTLOG_ERROR_TYPE, // event type
        0, // event category
        0, // event ID
        NULL, // current user's SID
        2, // strings in lpszStrings
        0, // no bytes of raw data
        (LPCTSTR *)lpszStrings, // array of error strings
        NULL); // no raw data

        (VOID) DeregisterEventSource(hEventSource);
    }
}

inline void ReleaseInterface(IUnknown *pUnk)
{
    if (pUnk)
    {
        pUnk->Release();
        pUnk = NULL;
    }
}

/* FUNCTION: CCOMPONENT_ERR::ErrorText
*
*/

char* CCOMPONENT_ERR::ErrorText(void)
{
    static SERRORMSG errorMsgs[] =
    {
        { ERR_MISSING_REGISTRY_ENTRIES,
        "Required entries missing from registry."
        },
        { ERR_LOADDLL_FAILED,
        "Load of DLL failed. DLL="
        },
    },

```

```

        { ERR_GETPROCADDR_FAILED,
        "Could not map proc in DLL. GetProcAddress error. DLL="
        },
        { ERR_UNKNOWN_DB_PROTOCOL,
        "Unknown database protocol specified in registry."
        },
        { 0,
        ""
        }
    };

    char szTmp[256];
    int i = 0;
    while (TRUE)
    {
        if (errorMsgs[i].szMsg[0] == 0)
        {
            strcpy( szTmp, "Unknown error
            number.");
            break;
        }
        if (m_Error == errorMsgs[i].iError)
        {
            strcpy( szTmp, errorMsgs[i].szMsg );
            break;
        }
        i++;
    }

    if (m_szTextDetail)
        strcat( szTmp, m_szTextDetail );
    if (m_SystemErr)
        wsprintf( szTmp+strlen(szTmp), " Error=%d",
        m_SystemErr );

    m_szErrorText = new char[strlen(szTmp)+1];
    strcpy( m_szErrorText, szTmp );
    return m_szErrorText;
}

CTPCC_Common_Remote::CTPCC_Common_Remote()
{
    m_pTxn = NULL;
    m_bCanBePooled = TRUE;
}

CTPCC_Common_Remote::~CTPCC_Common_Remote()
{
    if (m_pTxn)
        delete m_pTxn;
}

HRESULT CTPCC_Common_Remote::CallSetComplete()
{
    IObjectContext* pObjectContext = NULL;

    // get our object context
    HRESULT hr = CoGetObjectContext( IID_IObjectContext,
    (void **)&pObjectContext );
    pObjectContext->SetComplete();
    ReleaseInterface(pObjectContext);
    return hr;
}

//
// called by the ctor activator
//

```

```

STDMETHODIMP CTPCC_Common_Remote::Construct(IDispatch *
pUnk)
{
    // Code to access construction string, if needed later...
    // if (!pUnk)
    //     return E_UNEXPECTED;
    // IObjectConstructString * pString = NULL;
    // HRESULT hr =
pUnk->QueryInterface(IID_IObjectConstructString, (void
***)&pString);
    // pString->Release();
    try
    {
        if (Reg.eDB_Protocol == ODBC)
            m_pTxn = pCTPCC_ODBC_new(
Reg.szDbServer, Reg.szDbUser, Reg.szDbPassword,
szMyComputerName, Reg.szDbName);
        else if (Reg.eDB_Protocol == DBLIB)
            m_pTxn = pCTPCC_DBLIB_new(
Reg.szDbServer, Reg.szDbUser, Reg.szDbPassword,
szMyComputerName, Reg.szDbName);
    }
    catch (CBaseErr *e)
    {
        WriteMessageToEventLog(e->ErrorText());
        delete e;
        return E_FAIL;
    }
    catch (...)
    {
        WriteMessageToEventLog(TEXT("Unhandled
exception in object ::Construct"));
        return E_FAIL;
    }
    return S_OK;
}

HRESULT CTPCC_Common_Remote::NewOrder(VARIANT txn_in,
VARIANT* txn_out)
{
    PNEW_ORDER_DATA    pNewOrder;
    COM_DATA            *pData;
    try
    {
        pData = (COM_DATA*)txn_in.parray->pvData;
        pNewOrder = m_pTxn->BuffAddr_NewOrder();

        memcpy(pNewOrder, &pData->u.NewOrder,
sizeof(NEW_ORDER_DATA));

        VariantInit(txn_out);
        txn_out->vt = VT_SAFEARRAY;
        txn_out->parray = SafeArrayCreateVector(
VT_UI1,
txn_in.parray->rgsabound->cElements,
txn_in.parray->rgsabound->cElements);
        pData = (COM_DATA*)
txn_out->parray->pvData;

        m_pTxn->Payment(FALSE);

        memcpy( &pData->u.Payment, pNewOrder,
sizeof(PAYMENT_DATA));

        pData->retval = ERR_SUCCESS;
        pData->error = 0;
        return S_OK;
    }
    catch (CBaseErr *e)
    {
        // check for lost database connection; if yes,
component is toast
    }
}

pData->retval = ERR_SUCCESS;
pData->error = 0;
return S_OK;
}
catch (CBaseErr *e)
{
    // check for lost database connection; if yes,
component is toast
    if ( ((e->ErrorType() == ERR_TYPE_DBLIB)
&& (e->ErrorNum() == 10005)) ||
        ((e->ErrorType() ==
ERR_TYPE_ODBC) && (e->ErrorNum() == 10054)) )
        m_bCanBePooled = FALSE;

    pData->retval = e->ErrorType();
    pData->error = e->ErrorNum();
    delete e;
    return E_FAIL;
}
catch (...)
{
    WriteMessageToEventLog(TEXT("Unhandled
exception.));
    pData->retval = ERR_TYPE_LOGIC;
    pData->error = 0;
    m_bCanBePooled = FALSE;
    return E_FAIL;
}
}

HRESULT CTPCC_Common_Remote::Payment(VARIANT txn_in,
VARIANT* txn_out)
{
    PPAYMENT_DATA pPayment;
    COM_DATA      *pData;
    try
    {
        pData = (COM_DATA*)txn_in.parray->pvData;
        pPayment = m_pTxn->BuffAddr_Payment();

        memcpy(pPayment, &pData->u.Payment,
sizeof(PAYMENT_DATA));

        VariantInit(txn_out);
        txn_out->vt = VT_SAFEARRAY;
        txn_out->parray = SafeArrayCreateVector(
VT_UI1,
txn_in.parray->rgsabound->cElements,
txn_in.parray->rgsabound->cElements);
        pData = (COM_DATA*)
txn_out->parray->pvData;

        m_pTxn->Payment(FALSE);

        memcpy( &pData->u.Payment, pPayment,
sizeof(PAYMENT_DATA));

        pData->retval = ERR_SUCCESS;
        pData->error = 0;
        return S_OK;
    }
    catch (CBaseErr *e)
    {
        // check for lost database connection; if yes,
component is toast
    }
}

```

```

        if ( ((e->ErrorType() == ERR_TYPE_DBLIB)
&& (e->ErrorNum() == 10005)) ||
        ((e->ErrorType() ==
ERR_TYPE_ODBC) && (e->ErrorNum() == 10054)) )
            m_bCanBePooled = FALSE;

        pData->retval = e->ErrorType();
        pData->error = e->ErrorNum();
        delete e;
        return E_FAIL;
    }
    catch (...)
    {
        WriteMessageToEventLog(TEXT("Unhandled
exception.));

        pData->retval = ERR_TYPE_LOGIC;
        pData->error = 0;
        m_bCanBePooled = FALSE;
        return E_FAIL;
    }
}

HRESULT CTPCC_Common_Remote::StockLevel(VARIANT txn_in,
VARIANT* txn_out)
{
    PSTOCK_LEVEL_DATA    pStockLevel;
    COM_DATA              *pData;

    try
    {
        pData = (COM_DATA*)txn_in.parray->pvData;
        pStockLevel =
m_pTxn->BuffAddr_StockLevel();

        memcpy(pStockLevel, &pData->u.StockLevel,
sizeof(STOCK_LEVEL_DATA));

        VariantInit(txn_out);
        txn_out->vt = VT_SAFEARRAY;
        txn_out->parray = SafeArrayCreateVector(
VT_UI1,
txn_in.parray->rgsabound->cElements,
txn_in.parray->rgsabound->cElements);
        pData =
(COM_DATA*)txn_out->parray->pvData;

        m_pTxn->StockLevel();

        memcpy( &pData->u.StockLevel, pStockLevel,
sizeof(STOCK_LEVEL_DATA));

        pData->retval = ERR_SUCCESS;
        pData->error = 0;
        return S_OK;
    }
    catch (CBaseErr *e)
    {
        // check for lost database connection; if yes,
component is toast
        if ( ((e->ErrorType() == ERR_TYPE_DBLIB)
&& (e->ErrorNum() == 10005)) ||
        ((e->ErrorType() ==
ERR_TYPE_ODBC) && (e->ErrorNum() == 10054)) )
            m_bCanBePooled = FALSE;
    }
}

```

```

        pData->retval = e->ErrorType();
        pData->error = e->ErrorNum();
        delete e;
        return E_FAIL;
    }
    catch (...)
    {
        WriteMessageToEventLog(TEXT("Unhandled
exception.));

        pData->retval = ERR_TYPE_LOGIC;
        pData->error = 0;
        m_bCanBePooled = FALSE;
        return E_FAIL;
    }
}

HRESULT CTPCC_Common_Remote::OrderStatus(VARIANT
txn_in, VARIANT* txn_out)
{
    PORDER_STATUS_DATA    pOrderStatus;
    COM_DATA              *pData;
    try
    {
        pData = (COM_DATA*)txn_in.parray->pvData;
        pOrderStatus =
m_pTxn->BuffAddr_OrderStatus();

        memcpy(pOrderStatus, &pData->u.OrderStatus,
sizeof(ORDER_STATUS_DATA));

        VariantInit(txn_out);
        txn_out->vt = VT_SAFEARRAY;
        txn_out->parray = SafeArrayCreateVector(
VT_UI1,
txn_in.parray->rgsabound->cElements,
txn_in.parray->rgsabound->cElements);
        pData =
(COM_DATA*)txn_out->parray->pvData;

        m_pTxn->OrderStatus();

        memcpy( &pData->u.OrderStatus, pOrderStatus,
sizeof(ORDER_STATUS_DATA));

        pData->retval = ERR_SUCCESS;
        pData->error = 0;
        return S_OK;
    }
    catch (CBaseErr *e)
    {
        // check for lost database connection; if yes,
component is toast
        if ( ((e->ErrorType() == ERR_TYPE_DBLIB)
&& (e->ErrorNum() == 10005)) ||
        ((e->ErrorType() ==
ERR_TYPE_ODBC) && (e->ErrorNum() == 10054)) )
            m_bCanBePooled = FALSE;
    }
}

```

```

        {
            WriteMessageToEventLog(TEXT("Unhandled
exception.));
            pData->retval = ERR_TYPE_LOGIC;
            pData->error = 0;
            m_bCanBePooled = FALSE;
            return E_FAIL;
        }
    }
}

```

tpcc_com_remote.def

: tpcc_com_remote.def : Declares the module parameters.

```
LIBRARY "tpcc_com_remote.dll"
```

```
EXPORTS
```

```

    DllCanUnloadNow @1 PRIVATE
    DllGetClassObject @2 PRIVATE
    DllRegisterServer @3 PRIVATE
    DllUnregisterServer @4 PRIVATE

```

tpcc_com_remote.h

/* this ALWAYS GENERATED file contains the definitions for the interfaces */

/* File created by MIDL compiler version 5.01.0164 */

/* at Fri Jan 19 11:15:15 2001

*/

/* Compiler settings for .\src\tpcc_com_remote.idl:
 Oicf (OptLev=i2), W1, Zp8, env=Win32, ms_ext, c_ext
 error checks: allocation ref bounds_check enum stub_data
 */

//@@MIDL_FILE_HEADING()

/* verify that the <rpcndr.h> version is high enough to compile this file*/

```

#ifndef __REQUIRED_RPCNDR_H_VERSION__
#define __REQUIRED_RPCNDR_H_VERSION__ 440
#endif

```

```

#include "rpc.h"
#include "rpcndr.h"

```

```

#ifndef __tpcc_com_remote_h__
#define __tpcc_com_remote_h__

```

```

#ifdef __cplusplus
extern "C"{
#endif

```

/* Forward Declarations */

```

#ifndef __TPCCrem_FWD_DEFINED__
#define __TPCCrem_FWD_DEFINED__

```

```

#ifdef __cplusplus
typedef class TPCCrem TPCCrem;
#else
typedef struct TPCCrem TPCCrem;
#endif /* __cplusplus */

```

```
#endif /* __TPCCrem_FWD_DEFINED__ */
```

/* header files for imported files */

```

#include "oidl.h"
#include "ocidl.h"
#include "tpcc_com_remote_ps.h"

```

```

void __RPC_FAR * __RPC_USER MIDL_user_allocate(size_t);
void __RPC_USER MIDL_user_free( void __RPC_FAR * );

```

```

/* interface __MIDL_itf_tpcc_com_remote_0000 */
/* [local] */

```

```

extern RPC_IF_HANDLE
__MIDL_itf_tpcc_com_remote_0000_v0_0_c_ifspec;
extern RPC_IF_HANDLE
__MIDL_itf_tpcc_com_remote_0000_v0_0_s_ifspec;

```

```

#ifndef __TPCCremLib_LIBRARY_DEFINED__
#define __TPCCremLib_LIBRARY_DEFINED__

```

```

/* library TPCCremLib */
/* [helpstring][version][uuid] */

```

```
EXTERN_C const IID LIBID_TPCCremLib;
```

```
EXTERN_C const CLSID CLSID_TPCCrem;
```

```
#ifdef __cplusplus
```

```

class
DECLSPEC_UUID("63EC25AB-828A-4ed7-8C6C-B46D29889594")
TPCCrem;
#endif
#endif /* __TPCCremLib_LIBRARY_DEFINED__ */

```

/* Additional Prototypes for ALL interfaces */

/* end of Additional Prototypes */

```

#ifdef __cplusplus
}
#endif

```

```
#endif
```

tpcc_com_remote.idl

```

/* FILE: TPCC.IDL
 * Microsoft TPC-C Kit Ver.
4.20.000
 * Copyright Microsoft,
1999
 * All Rights Reserved
 * not yet audited
 *
 * PURPOSE: IDL source for TPCC.dll. This file is
processed by the MIDL tool to
 * produce the type library
(TPCC.tlb) and marshalling code.

```

```

*
* Change history:
*           4.20.000 - first version
*/

interface TPCCrem;

import "oidl.idl";
import "ocidl.idl";
import "..\tpcc_com_remote_ps\src\tpcc_com_remote_ps.idl";

[
    uuid(B207575F-7A88-489a-9383-859E2D771537),
    version(1.0),
    helpstring("Remote TPC-C 1.0 Type Library")
]
library TPCCremLib
{
    importlib("stdole32.tlb");
    importlib("stdole2.tlb");

    [
        uuid(63EC25AB-828A-4ed7-8C6C-B46D29889594),
        helpstring("Remote All Txns Class")
    ]
    coclass TPCCrem
    {
        [default] interface ITPCCrem;
    };
};

tpcc_com_remote.rc

//Microsoft Developer Studio generated resource script.
//
#include "resource.h"

#define APSTUDIO_READONLY_SYMBOLS
//
// Generated from the TEXTINCLUDE2 resource.
//
#include "winres.h"

//
// English (U.S.) resources

#if !defined(AFX_RESOURCE_DLL) || defined(AFX_TARG_ENU)
#ifndef _WIN32
LANGUAGE LANG_ENGLISH, SUBLANG_ENGLISH_US
#pragma code_page(1252)
#endif
#endif

#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_remote.tlb""\r\n"
    "\0"
END

#endif // APSTUDIO_INVOKED

#ifdef _MAC
//
// Version
//
VS_VERSION_INFO VERSIONINFO
FILEVERSION 1,0,0,1
PRODUCTVERSION 1,0,0,1
FILEFLAGSMASK 0x3fL
#ifdef _DEBUG
FILEFLAGS 0x1L
#else
FILEFLAGS 0x0L
#endif
FILEOS 0x4L
FILETYPE 0x2L
FILESUBTYPE 0x0L
BEGIN
    BLOCK "StringFileInfo"
    BEGIN
        BLOCK "040904B0"
        BEGIN
            VALUE "CompanyName", "\0"
            VALUE "FileDescription", "tpcc_com_remote Module\0"
            VALUE "FileVersion", "1, 0, 0, 1\0"
            VALUE "InternalName", "TPCCNEWORDER\0"
            VALUE "LegalCopyright", "Copyright 1997\0"
            VALUE "OriginalFilename", "tpcc_com_remote.DLL\0"
            VALUE "ProductName", "tpcc_com_remote Module\0"
            VALUE "ProductVersion", "1, 0, 0, 1\0"
            VALUE "OLESelfRegister", "\0"
        END
    END
    BLOCK "VarFileInfo"
    BEGIN
        VALUE "Translation", 0x409, 1200
    END
END

#endif // !_MAC

//
// REGISTRY
//

```

```
IDR_TPCCrem          REGISTRY DISCARDABLE
"tpcc_com_remote.rgs"
```

```
////////////////////////////////////
//
// String Table
//
```

```
STRINGTABLEDISCARDABLE
BEGIN
    IDS_PROJNAME          "tpcc_com_remote"
END
```

```
#endif // English (U.S.) resources
////////////////////////////////////
```

```
#ifndef APSTUDIO_INVOKED
////////////////////////////////////
//
// Generated from the TEXTINCLUDE3 resource.
//
1 TYPELIB "tpcc_com_remote.tlb"
```

```
////////////////////////////////////
#endif // not APSTUDIO_INVOKED
```

tpcc_com_remote.rgs

```
HKCR
{
    TPCC.AllRemoteTxns.1 = s 'All RemoteTxns Class'
    {
        CLSID = s
        '{63EC25AB-828A-4ed7-8C6C-B46D29889594}'
    }
    TPCC.AllRemoteTxns = s 'TPCC Remote Class'
    {
        CurVer = s 'TPCC.AllRemoteTxns.1'
    }
    NoRemove CLSID
    {
        ForceRemove
        {63EC25AB-828A-4ed7-8C6C-B46D29889594} = s 'TPCC Remote
        Class'
    }
    ProgID = s 'TPCC.AllRemoteTxns.1'
    VersionIndependentProgID = s
    'TPCC.AllRemoteTxns'
    InprocServer32 = s '%MODULE%'
    {
        val ThreadingModel = s
        'Both'
    }
}
}
```

tpcc_com_remote.i.c

```
/* this file contains the actual definitions of */
/* the IIDs and CLSIDs */
```

```
/* link this file in with the server and any clients */
```

```
/* File created by MIDL compiler version 5.01.0164 */
/* at Fri Jan 19 11:15:15 2001
*/
/* Compiler settings for .\src\tpcc_com_remote.idl:
Oicf (OptLev=i2), W1, Zp8, env=Win32, ms_ext, c_ext
error checks: allocation ref bounds_check enum stub_data
*/
```

```
//@@MIDL_FILE_HEADING( )
#ifdef __cplusplus
extern "C"{
#endif
```

```
#ifndef __IID_DEFINED__
#define __IID_DEFINED__
```

```
typedef struct _IID
{
    unsigned long x;
    unsigned short s1;
    unsigned short s2;
    unsigned char c[8];
} IID;
```

```
#endif // __IID_DEFINED__
```

```
#ifndef CLSID_DEFINED
#define CLSID_DEFINED
typedef IID CLSID;
#endif // CLSID_DEFINED
```

```
const IID LIBID_TPCCremLib =
{0xB207575F,0x7A88,0x489a,{0x93,0x83,0x85,0x9E,0x2D,0x77,0x1
5,0x37}};
```

```
const CLSID CLSID_TPCCrem =
{0x63EC25AB,0x828A,0x4ed7,{0x8C,0x6C,0xB4,0x6D,0x29,0x88,0
x95,0x94}};
```

```
#ifdef __cplusplus
}
#endif
```

tpcc_com_remote.Methods.h

```
/* FILE: METHODS.H
* Microsoft TPC-C Kit Ver.
4.20.000
* Copyright Microsoft,
1999
* All Rights Reserved
* not yet audited
* PURPOSE: Header file for COM components.
* Change history:
* 4.20.000 - first version
*/
```

```
enum COMPONENT_ERROR
```



```

{
    ERR_MISSING_REGISTRY_ENTRIES = 1,
    ERR_LOADDLL_FAILED,
    ERR_GETPROCADDR_FAILED,
    ERR_UNKNOWN_DB_PROTOCOL
};

class CCOMPONENT_ERR : public CBaseErr
{
public:
    CCOMPONENT_ERR(COMPONENT_ERROR
Err)
    {
        m_Error = Err;
        m_szTextDetail = NULL;
        m_SystemErr = 0;
        m_szErrorText = NULL;
    };

    CCOMPONENT_ERR(COMPONENT_ERROR
Err, char *szTextDetail, DWORD dwSystemErr)
    {
        m_Error = Err;
        m_szTextDetail = new
char[strlen(szTextDetail)+1];
strcpy( m_szTextDetail, szTextDetail
);

        m_SystemErr = dwSystemErr;
        m_szErrorText = NULL;
    };

    ~CCOMPONENT_ERR()
    {
        if (m_szTextDetail != NULL)
            delete [] m_szTextDetail;
        if (m_szErrorText != NULL)
            delete [] m_szErrorText;
    };

    COMPONENT_ERROR      m_Error;
    char                  *m_szTextDetail;
    char                  *m_szErrorText;
    DWORD                 m_SystemErr;

    int ErrorType() {return
ERR_TYPE_COMPONENT;};
    int ErrorNum() {return m_Error;};
    char *ErrorText();
};

static void WriteMessageToEventLog(LPTSTR lpszMsg);

////////////////////////////////////
// CTPCC_Common
class CTPCC_Common_Remote :
public ITPCCrem,
public IObjectControl,
public IObjectConstruct,
public CComObjectRootEx<CComSingleThreadModel>
{
public:
    BEGIN_COM_MAP(CTPCC_Common_Remote)
        COM_INTERFACE_ENTRY(ITPCCrem)
        COM_INTERFACE_ENTRY(IObjectControl)

```

```

        COM_INTERFACE_ENTRY(IObjectConstruct)
    END_COM_MAP()

    CTPCC_Common_Remote();
    ~CTPCC_Common_Remote();

// ITPCCrem
public:
    HRESULT __stdcall NewOrder(          VARIANT
    txn_in, VARIANT* txn_out);
    HRESULT __stdcall Payment(          VARIANT
    txn_in, VARIANT* txn_out);
    HRESULT __stdcall Delivery(         VARIANT
    txn_in) {return E_NOTIMPL;};
    HRESULT __stdcall StockLevel(VARIANT txn_in,
    VARIANT* txn_out);
    HRESULT __stdcall OrderStatus(      VARIANT
    txn_in, VARIANT* txn_out);

    HRESULT __stdcall CallSetComplete();

// IObjectControl
    STDMETHODIMP_(BOOL) CanBePooled() { return
m_bCanBePooled; }
    STDMETHODIMP Activate() { return S_OK; } //
we don't support COM Services transactions (no enlistment)
    STDMETHODIMP_(void) Deactivate() { /* nothing to do
*/ }

// IObjectConstruct
    STDMETHODIMP Construct(IDispatch * pUnk);

// helper methods
private:
    BOOL      m_bCanBePooled;
    CTPCC_BASE *m_pTxn;

    struct COM_DATA
    {
        int retVal;
        int error;
        union
        {
            NEW_ORDER_DATA
            PAYMENT_DATA
            DELIVERY_DATA
            STOCK_LEVEL_DATA
            ORDER_STATUS_DATA
        } u;
    };
};

////////////////////////////////////
// CTPCC
class CTPCCrem :
public CTPCC_Common_Remote,
public CComCoClass<CTPCCrem, &CLSID_TPCCrem>
{
public:
    DECLARE_REGISTRY_RESOURCEID(IDR_TPCCrem)
    BEGIN_COM_MAP(CTPCCrem)

```

```

        COM_INTERFACE_ENTRY2(IUnknown,
        CComObjectRootEx)

COM_INTERFACE_ENTRY_CHAIN(CTPCC_Common_Remote)
END_COM_MAP()

};

```

tpcc_com_remote_ps.def

```
LIBRARY "tpcc_com_remote_ps"
```

```
DESCRIPTION 'Proxy/Stub DLL'
```

```
EXPORTS
```

```

    DllGetClassObject    @1    PRIVATE
    DllCanUnloadNow     @2    PRIVATE
    GetProxyDllInfo     @3    PRIVATE
    DllRegisterServer   @4    PRIVATE
    DllUnregisterServer @5    PRIVATE

```

tpcc_com_remote_ps.h

```
/* this ALWAYS GENERATED file contains the definitions for the
interfaces */
```

```

/* File created by MIDL compiler version 5.01.0164 */
/* at Fri Jan 19 11:15:07 2001
*/
/* Compiler settings for .\src\tpcc_com_remote_ps.idl:
    Oicf (OptLev=i2), W1, Zp8, env=Win32, ms_ext, c_ext
    error checks: allocation ref bounds_check enum stub_data
*/
//@@@MIDL_FILE_HEADING( )

```

```
/* verify that the <rpcndr.h> version is high enough to compile this
file*/
```

```

#ifndef __REQUIRED_RPCNDR_H_VERSION__
#define __REQUIRED_RPCNDR_H_VERSION__ 440
#endif

```

```

#include "rpc.h"
#include "rpcndr.h"

```

```

#ifndef __RPCNDR_H_VERSION__
#error this stub requires an updated version of <rpcndr.h>
#endif // __RPCNDR_H_VERSION__

```

```

#ifndef COM_NO_WINDOWS_H
#include "windows.h"
#include "ole2.h"
#endif /*COM_NO_WINDOWS_H*/

```

```

#ifndef __tpcc_com_remote_ps_h__
#define __tpcc_com_remote_ps_h__

```

```

#ifdef __cplusplus
extern "C"{
#endif

```

```
/* Forward Declarations */
```

```

#ifndef __ITPCCrem_FWD_DEFINED__
#define __ITPCCrem_FWD_DEFINED__

```

```

typedef interface ITPCCrem ITPCCrem;
#endif /* __ITPCCrem_FWD_DEFINED__ */

```

```
/* header files for imported files */
```

```

#include "oaidl.h"
#include "ocidl.h"

```

```

void __RPC_FAR * __RPC_USER MIDL_user_allocate(size_t);
void __RPC_USER MIDL_user_free( void __RPC_FAR * );

```

```

/* interface __MIDL_itf_tpcc_com_remote_ps_0000 */
/* [local] */

```

```

extern RPC_IF_HANDLE
__MIDL_itf_tpcc_com_remote_ps_0000_v0_0_c_ifspec;
extern RPC_IF_HANDLE
__MIDL_itf_tpcc_com_remote_ps_0000_v0_0_s_ifspec;

```

```

#ifndef __ITPCCrem_INTERFACE_DEFINED__
#define __ITPCCrem_INTERFACE_DEFINED__

```

```

/* interface ITPCCrem */
/* [unique][helpstring][uuid][oleautomation][object] */

```

```
EXTERN_C const IID IID_ITPCCrem;
```

```
#if defined(__cplusplus) && !defined(CINTERFACE)
```

```
MIDL_INTERFACE("4899AD2E-D521-4d59-824A-4381438FD840")
```

```

ITPCCrem : public IUnknown
{
public:
    virtual HRESULT __stdcall NewOrder(
        /* [in] */ VARIANT txn_in,
        /* [out] */ VARIANT __RPC_FAR *txn_out) = 0;

    virtual HRESULT __stdcall Payment(
        /* [in] */ VARIANT txn_in,
        /* [out] */ VARIANT __RPC_FAR *txn_out) = 0;

    virtual HRESULT __stdcall Delivery(
        /* [in] */ VARIANT txn_in) = 0;

    virtual HRESULT __stdcall StockLevel(
        /* [in] */ VARIANT txn_in,
        /* [out] */ VARIANT __RPC_FAR *txn_out) = 0;

    virtual HRESULT __stdcall OrderStatus(
        /* [in] */ VARIANT txn_in,
        /* [out] */ VARIANT __RPC_FAR *txn_out) = 0;

    virtual HRESULT __stdcall CallSetComplete( void) = 0;
};

```

```
#else /* C style interface */
```

```

typedef struct ITPCCremVtbl
{
    BEGIN_INTERFACE

```

```

HRESULT ( STDMETHODCALLTYPE __RPC_FAR
*QueryInterface)(
    ITPCCrem __RPC_FAR * This,
    /* [in] */ REFIID riid,
    /* [iid_is][out] */ void __RPC_FAR * __RPC_FAR
*ppvObject);

ULONG ( STDMETHODCALLTYPE __RPC_FAR *AddRef)(
    ITPCCrem __RPC_FAR * This);

ULONG ( STDMETHODCALLTYPE __RPC_FAR *Release)(
    ITPCCrem __RPC_FAR * This);

HRESULT ( _stdcall __RPC_FAR *NewOrder)(
    ITPCCrem __RPC_FAR * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT __RPC_FAR *txn_out);

HRESULT ( _stdcall __RPC_FAR *Payment)(
    ITPCCrem __RPC_FAR * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT __RPC_FAR *txn_out);

HRESULT ( _stdcall __RPC_FAR *Delivery)(
    ITPCCrem __RPC_FAR * This,
    /* [in] */ VARIANT txn_in);

HRESULT ( _stdcall __RPC_FAR *StockLevel)(
    ITPCCrem __RPC_FAR * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT __RPC_FAR *txn_out);

HRESULT ( _stdcall __RPC_FAR *OrderStatus)(
    ITPCCrem __RPC_FAR * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT __RPC_FAR *txn_out);

HRESULT ( _stdcall __RPC_FAR *CallSetComplete)(
    ITPCCrem __RPC_FAR * This);

END_INTERFACE
} ITPCCremVtbl;

interface ITPCCrem
{
    CONST_VTBL struct ITPCCremVtbl __RPC_FAR *IpVtbl;
};

#ifdef COBJMACROS

#define ITPCCrem_QueryInterface(This,riid,ppvObject) \
    (This)->IpVtbl -> QueryInterface(This,riid,ppvObject)

#define ITPCCrem_AddRef(This) \
    (This)->IpVtbl -> AddRef(This)

#define ITPCCrem_Release(This) \
    (This)->IpVtbl -> Release(This)

#define ITPCCrem_NewOrder(This,txn_in,txn_out) \
    (This)->IpVtbl -> NewOrder(This,txn_in,txn_out)

#define ITPCCrem_Payment(This,txn_in,txn_out) \
    (This)->IpVtbl -> Payment(This,txn_in,txn_out)

```

```

#define ITPCCrem_Delivery(This,txn_in) \
    (This)->IpVtbl -> Delivery(This,txn_in)

#define ITPCCrem_StockLevel(This,txn_in,txn_out) \
    (This)->IpVtbl -> StockLevel(This,txn_in,txn_out)

#define ITPCCrem_OrderStatus(This,txn_in,txn_out) \
    (This)->IpVtbl -> OrderStatus(This,txn_in,txn_out)

#define ITPCCrem_CallSetComplete(This) \
    (This)->IpVtbl -> CallSetComplete(This)

#endif /* COBJMACROS */

#ifdef /* C style interface */

HRESULT _stdcall ITPCCrem_NewOrder_Proxy(
    ITPCCrem __RPC_FAR * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT __RPC_FAR *txn_out);

void __RPC_STUB ITPCCrem_NewOrder_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer * _pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,
    DWORD *_pdwStubPhase);

HRESULT _stdcall ITPCCrem_Payment_Proxy(
    ITPCCrem __RPC_FAR * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT __RPC_FAR *txn_out);

void __RPC_STUB ITPCCrem_Payment_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer * _pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,
    DWORD *_pdwStubPhase);

HRESULT _stdcall ITPCCrem_Delivery_Proxy(
    ITPCCrem __RPC_FAR * This,
    /* [in] */ VARIANT txn_in);

void __RPC_STUB ITPCCrem_Delivery_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer * _pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,
    DWORD *_pdwStubPhase);

HRESULT _stdcall ITPCCrem_StockLevel_Proxy(
    ITPCCrem __RPC_FAR * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT __RPC_FAR *txn_out);

void __RPC_STUB ITPCCrem_StockLevel_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer * _pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,

```

```

DWORD *_pdwStubPhase);

HRESULT __stdcall ITPCCrem_OrderStatus_Proxy(
    ITPCCrem __RPC_FAR * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT __RPC_FAR *txn_out);

void __RPC_STUB ITPCCrem_OrderStatus_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *_pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,
    DWORD *_pdwStubPhase);

HRESULT __stdcall ITPCCrem_CallSetComplete_Proxy(
    ITPCCrem __RPC_FAR * This);

void __RPC_STUB ITPCCrem_CallSetComplete_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *_pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,
    DWORD *_pdwStubPhase);

#endif /* __ITPCCrem_INTERFACE_DEFINED__ */

/* Additional Prototypes for ALL interfaces */

unsigned long __RPC_USER VARIANT_UserSize(
    unsigned long __RPC_FAR *, unsigned long , VARIANT
    __RPC_FAR * );
unsigned char __RPC_FAR * __RPC_USER
VARIANT_UserMarshal( unsigned long __RPC_FAR *, unsigned
char __RPC_FAR *, VARIANT __RPC_FAR * );
unsigned char __RPC_FAR * __RPC_USER
VARIANT_UserUnmarshal(unsigned long __RPC_FAR *, unsigned
char __RPC_FAR *, VARIANT __RPC_FAR * );
void __RPC_USER VARIANT_UserFree( unsigned
long __RPC_FAR *, VARIANT __RPC_FAR * );

/* end of Additional Prototypes */

#ifdef __cplusplus
}
#endif

#endif

tpcc_com_remote_ps.idl

/* FILE: ITPCC.IDL
 * Microsoft TPC-C Kit Ver.
4.20.000
 * Copyright Microsoft,
1999
 * All Rights Reserved
 *
 * not yet audited
 *
 * PURPOSE: Defines the interface used by TPCC.
This interface can be implemented by C++ components.

```

```

*
* Change history:
* 4.20.000 - first version
*/

// Forward declare all types defined
interface ITPCCrem;
import "oidl.idl";
import "ocidl.idl";

[
    object,
    oleautomation,
    uuid(4899AD2E-D521-4d59-824A-4381438FD840),
    helpstring("ITPCC Interface"),
    pointer_default(unique)
]
interface ITPCCrem : IUnknown
{
    HRESULT __stdcall NewOrder
    (
    [in]
    VARIANT txn_in,
    [out] VARIANT *txn_out
    );

    HRESULT __stdcall Payment
    (
    [in]
    VARIANT txn_in,
    [out] VARIANT *txn_out
    );

    HRESULT __stdcall Delivery
    (
    [in]
    VARIANT txn_in
    );

    HRESULT __stdcall StockLevel
    (
    [in]
    VARIANT txn_in,
    [out] VARIANT *txn_out
    );

    HRESULT __stdcall OrderStatus
    (
    [in]
    VARIANT txn_in,
    [out] VARIANT *txn_out
    );

    HRESULT __stdcall CallSetComplete
    (
    );

}; // interface ITPCC

tpcc_com_remote_ps_i.c

```

```

/* this file contains the actual definitions of */
/* the IIDs and CLSIDs */

/* link this file in with the server and any clients */

/* File created by MIDL compiler version 5.01.0164 */
/* at Fri Jan 19 11:15:07 2001
*/
/* Compiler settings for .\src\tpcc_com_remote_ps.idl:
Oicf (OptLev=i2), W1, Zp8, env=Win32, ms_ext, c_ext
error checks: allocation ref bounds_check enum stub_data
*/
//@@@MIDL_FILE_HEADING( )
#ifdef __cplusplus
extern "C"{
#endif

#ifdef __IID_DEFINED__
#define __IID_DEFINED__

typedef struct _IID
{
    unsigned long x;
    unsigned short s1;
    unsigned short s2;
    unsigned char c[8];
} IID;

#endif // __IID_DEFINED__

#ifdef CLSID_DEFINED
#define CLSID_DEFINED
typedef IID CLSID;
#endif // CLSID_DEFINED

const IID IID_ITPCCrem =
{0x4899AD2E,0xD521,0x4d59,{0x82,0x4A,0x43,0x81,0x43,0x8F,0x
D8,0x40}};

#ifdef __cplusplus
}
#endif

tpcc_com_remote_ps_p.c

/* this ALWAYS GENERATED file contains the proxy stub code */

/* File created by MIDL compiler version 5.01.0164 */
/* at Fri Jan 19 11:15:07 2001
*/
/* Compiler settings for .\src\tpcc_com_remote_ps.idl:
Oicf (OptLev=i2), W1, Zp8, env=Win32, ms_ext, c_ext
error checks: allocation ref bounds_check enum stub_data
*/
//@@@MIDL_FILE_HEADING( )

#define USE_STUBLESS_PROXY

/* verify that the <rpcproxy.h> version is high enough to compile this
file*/
#endif __REDQ_RPCPROXY_H_VERSION__

```

```

#define __REQUIRED_RPCPROXY_H_VERSION__ 440
#endif

#include "rpcproxy.h"
#ifdef __RPCPROXY_H_VERSION__
#error this stub requires an updated version of <rpcproxy.h>
#endif // __RPCPROXY_H_VERSION__

#include "tpcc_com_remote_ps.h"

#define TYPE_FORMAT_STRING_SIZE 997
#define PROC_FORMAT_STRING_SIZE 187

typedef struct _MIDL_TYPE_FORMAT_STRING
{
    short Pad;
    unsigned char Format[ TYPE_FORMAT_STRING_SIZE];
} MIDL_TYPE_FORMAT_STRING;

typedef struct _MIDL_PROC_FORMAT_STRING
{
    short Pad;
    unsigned char Format[ PROC_FORMAT_STRING_SIZE];
} MIDL_PROC_FORMAT_STRING;

extern const MIDL_TYPE_FORMAT_STRING
__MIDL_TypeFormatString;
extern const MIDL_PROC_FORMAT_STRING
__MIDL_ProcFormatString;

/* Standard interface: __MIDL_itf_tpcc_com_remote_ps_0000, ver.
0.0,
GUID={0x00000000,0x0000,0x0000,{0x00,0x00,0x00,0x00,0x00,0x
00,0x00,0x00}} */

/* Object interface: IUnknown, ver. 0.0,
GUID={0x00000000,0x0000,0x0000,{0xC0,0x00,0x00,0x00,0x00,0x
00,0x00,0x46}} */

/* Object interface: ITPCCrem, ver. 0.0,
GUID={0x4899AD2E,0xD521,0x4d59,{0x82,0x4A,0x43,0x81,0x43,0
x8F,0xD8,0x40}} */

extern const MIDL_STUB_DESC Object_StubDesc;

extern const MIDL_SERVER_INFO ITPCCrem_ServerInfo;

#pragma code_seg(".orpc")
extern const USER_MARSHAL_ROUTINE_QUADRUPLE
UserMarshalRoutines[1];

static const MIDL_STUB_DESC Object_StubDesc =
{
    0,
    NdrOleAllocate,
    NdrOleFree,
    0,

```

```

0,
0,
0,
0,
__MIDL_TypeFormatString.Format,
1, /* -error bounds_check flag */
0x20000, /* Ndr library version */
0,
0x50100a4, /* MIDL Version 5.1.164 */
0,
UserMarshalRoutines,
0, /* notify & notify_flag routine table */
1, /* Flags */
0, /* Reserved3 */
0, /* Reserved4 */
0 /* Reserved5 */
};

static const unsigned short ITPCCrem_FormatStringOffsetTable[] =
{
0,
34,
68,
96,
130,
164
};

static const MIDL_SERVER_INFO ITPCCrem_ServerInfo =
{
&Object_StubDesc,
0,
__MIDL_ProcFormatString.Format,
&ITPCCrem_FormatStringOffsetTable[-3],
0,
0,
0,
0
};

static const MIDL_STUBLESS_PROXY_INFO ITPCCrem_ProxyInfo =
{
&Object_StubDesc,
__MIDL_ProcFormatString.Format,
&ITPCCrem_FormatStringOffsetTable[-3],
0,
0,
0
};

CINTERFACE_PROXY_VTABLE(9)_ITPCCremProxyVtbl =
{
&ITPCCrem_ProxyInfo,
&IID_ITPCCrem,
IUnknown_QueryInterface_Proxy,
IUnknown_AddRef_Proxy,
IUnknown_Release_Proxy,
(void *)-1 /* ITPCCrem::NewOrder */,
(void *)-1 /* ITPCCrem::Payment */,
(void *)-1 /* ITPCCrem::Delivery */,
(void *)-1 /* ITPCCrem::StockLevel */,
(void *)-1 /* ITPCCrem::OrderStatus */,
(void *)-1 /* ITPCCrem::CallSetComplete */
};

const CInterfaceStubVtbl _ITPCCremStubVtbl =
{
&IID_ITPCCrem,
&ITPCCrem_ServerInfo,
9,
0, /* pure interpreted */
CStdStubBuffer_METHODS
};

#pragma data_seg("rdata")

static const USER_MARSHAL_ROUTINE_QUADRUPLE
UserMarshalRoutines[1] =
{
{
VARIANT_UserSize,
VARIANT_UserMarshal,
VARIANT_UserUnmarshal,
VARIANT_UserFree
}
};

#if !defined(__RPC_WIN32__)
#error Invalid build platform for this stub.
#endif

#if !(TARGET_IS_NT40_OR_LATER)
#error You need a Windows NT 4.0 or later to run this stub because it
uses these features:
#error -Oif or -Oicf, [wire_marshall] or [user_marshall] attribute, more
than 32 methods in the interface.
#error However, your C/C++ compilation flags indicate you intend to
run this app on earlier systems.
#error This app will die there with the
RPC_X_WRONG_STUB_VERSION error.
#endif

static const MIDL_PROC_FORMAT_STRING
__MIDL_ProcFormatString =
{
0,
{
/* Procedure NewOrder */
FC_AUTO_HANDLE* 0x33, /*
object, Oi2 */
/* 2 */ NdrFcLong( 0x0 ), /* 0 */
/* 6 */ NdrFcShort( 0x3 ), /* 3 */
#ifdef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 8 */ NdrFcShort( 0x1c ), /* x86 Stack size/offset = 28 */
#else
NdrFcShort( 0x20 ), /* MIPS &
PPC Stack size/offset = 32 */
#endif
#else
NdrFcShort( 0x28 ), /* Alpha Stack
size/offset = 40 */
#endif
/* 10 */ NdrFcShort( 0x0 ), /* 0 */
/* 12 */ NdrFcShort( 0x8 ), /* 8 */
/* 14 */ 0x7, /* Oi2 Flags: srv must size, clt must
size, has return, */
}
}
}

```

```

                                0x3,                /* 3 */
                                /* Parameter txn_in */
/* 16 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by
val, */
#ifndef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 18 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
#else
                                NdrFcShort( 0x8 ), /* MIPS &
PPC Stack size/offset = 8 */
#endif
#else
                                NdrFcShort( 0x8 ), /* Alpha Stack
size/offset = 8 */
#endif
/* 20 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */

                                /* Parameter txn_out */

/* 22 */ NdrFcShort( 0x4113 ), /* Flags: must size, must
free, out, simple ref, srv alloc size=16 */
#ifndef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 24 */ NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
#else
                                NdrFcShort( 0x18 ), /* MIPS &
PPC Stack size/offset = 24 */
#endif
#else
                                NdrFcShort( 0x18 ), /* Alpha Stack
size/offset = 24 */
#endif
/* 26 */ NdrFcShort( 0x3da ), /* Type Offset=986 */

                                /* Return value */

/* 28 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifndef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 30 */ NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
#else
                                NdrFcShort( 0x1c ), /* MIPS &
PPC Stack size/offset = 28 */
#endif
#else
                                NdrFcShort( 0x20 ), /* Alpha Stack
size/offset = 32 */
#endif
/* 32 */ 0x8, /* FC_LONG */
                                0x0, /* 0 */

                                /* Procedure Payment */

/* 34 */ 0x33, /* FC_AUTO_HANDLE */
                                0x6c, /* Old Flags:
object, Oi2 */
/* 36 */ NdrFcLong( 0x0 ), /* 0 */
/* 40 */ NdrFcShort( 0x4 ), /* 4 */
#ifndef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 42 */ NdrFcShort( 0x1c ), /* x86 Stack size/offset = 28 */
#else
                                NdrFcShort( 0x20 ), /* MIPS &
PPC Stack size/offset = 32 */
#endif
#else
                                NdrFcShort( 0x20 ), /* Alpha Stack
size/offset = 40 */
#endif
/* 44 */ NdrFcShort( 0x0 ), /* 0 */
/* 46 */ NdrFcShort( 0x8 ), /* 8 */
/* 48 */ 0x7, /* Oi2 Flags: srv must size, clt must
size, has return, */
                                0x3, /* 3 */

                                /* Parameter txn_in */

/* 50 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by
val, */
#ifndef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 52 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
#else
                                NdrFcShort( 0x8 ), /* MIPS &
PPC Stack size/offset = 8 */
#endif
#endif
/* 54 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */

                                /* Parameter txn_out */

/* 56 */ NdrFcShort( 0x4113 ), /* Flags: must size, must
free, out, simple ref, srv alloc size=16 */
#ifndef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 58 */ NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
#else
                                NdrFcShort( 0x18 ), /* MIPS &
PPC Stack size/offset = 24 */
#endif
#endif
/* 60 */ NdrFcShort( 0x3da ), /* Type Offset=986 */

                                /* Return value */

/* 62 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifndef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 64 */ NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
#else
                                NdrFcShort( 0x1c ), /* MIPS &
PPC Stack size/offset = 28 */
#endif
#endif
/* 66 */ 0x8, /* FC_LONG */
                                0x0, /* 0 */

                                /* Procedure Delivery */

/* 68 */ 0x33, /* FC_AUTO_HANDLE */
                                0x6c, /* Old Flags:
object, Oi2 */
/* 70 */ NdrFcLong( 0x0 ), /* 0 */
/* 74 */ NdrFcShort( 0x5 ), /* 5 */
#ifndef _ALPHA_

```

```

#if !defined(_MIPS_) && !defined(_PPC_)
/* 76 */ NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
#else
NdrFcShort( 0x1c ), /* MIPS &
PPC Stack size/offset = 28 */
#endif
#else
NdrFcShort( 0x20 ), /* Alpha Stack
size/offset = 32 */
#endif
/* 78 */ NdrFcShort( 0x0 ), /* 0 */
/* 80 */ NdrFcShort( 0x8 ), /* 8 */
/* 82 */ 0x6, /* Oi2 Flags: clt must size, has
return, */
0x2, /* 2 */

/* Parameter txn_in */

/* 84 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by
val, */
#ifdef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 86 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
#else
NdrFcShort( 0x8 ), /* MIPS &
PPC Stack size/offset = 8 */
#endif
#else
NdrFcShort( 0x8 ), /* Alpha Stack
size/offset = 8 */
#endif
/* 88 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */

/* Return value */

/* 90 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifdef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 92 */ NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
#else
NdrFcShort( 0x18 ), /* MIPS &
PPC Stack size/offset = 24 */
#endif
#else
NdrFcShort( 0x18 ), /* Alpha Stack
size/offset = 24 */
#endif
/* 94 */ 0x8, /* FC_LONG */
0x0, /* 0 */

/* Procedure StockLevel */

/* 96 */ 0x33, /* FC_AUTO_HANDLE */
0x6c, /* Old Flags:
object, Oi2 */
/* 98 */ NdrFcLong( 0x0 ), /* 0 */
/* 102 */ NdrFcShort( 0x6 ), /* 6 */
#ifdef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 104 */ NdrFcShort( 0x1c ), /* x86 Stack size/offset = 28 */
#else
NdrFcShort( 0x20 ), /* MIPS &
PPC Stack size/offset = 32 */
#endif
#else
NdrFcShort( 0x28 ), /* Alpha Stack
size/offset = 40 */
#endif
/* 106 */ NdrFcShort( 0x0 ), /* 0 */
/* 108 */ NdrFcShort( 0x8 ), /* 8 */
/* 110 */ 0x7, /* Oi2 Flags: srv must size, clt must
size, has return, */
0x3, /* 3 */

/* Parameter txn_in */

/* 112 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by
val, */
#ifdef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 114 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
#else
NdrFcShort( 0x8 ), /* MIPS &
PPC Stack size/offset = 8 */
#endif
#else
NdrFcShort( 0x8 ), /* Alpha Stack
size/offset = 8 */
#endif
/* 116 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */

/* Parameter txn_out */

/* 118 */ NdrFcShort( 0x4113 ), /* Flags: must size, must
free, out, simple ref, srv alloc size=16 */
#ifdef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 120 */ NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
#else
NdrFcShort( 0x18 ), /* MIPS &
PPC Stack size/offset = 24 */
#endif
#else
NdrFcShort( 0x18 ), /* Alpha Stack
size/offset = 24 */
#endif
/* 122 */ NdrFcShort( 0x3da ), /* Type Offset=986 */

/* Return value */

/* 124 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifdef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 126 */ NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
#else
NdrFcShort( 0x1c ), /* MIPS &
PPC Stack size/offset = 28 */
#endif
#else
NdrFcShort( 0x20 ), /* Alpha Stack
size/offset = 32 */
#endif
/* 128 */ 0x8, /* FC_LONG */
0x0, /* 0 */

/* Procedure OrderStatus */

/* 130 */ 0x33, /* FC_AUTO_HANDLE */
0x6c, /* Old Flags:
object, Oi2 */
/* 132 */ NdrFcLong( 0x0 ), /* 0 */
/* 136 */ NdrFcShort( 0x7 ), /* 7 */
#ifdef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 138 */ NdrFcShort( 0x1c ), /* x86 Stack size/offset = 28 */
#endif
#endif

```



```

NdrFcShort( 0x20 ), /* MIPS &
PPC Stack size/offset = 32 */
#endif
#else
NdrFcShort( 0x28 ), /* Alpha Stack
size/offset = 40 */
#endif
/* 140 */ NdrFcShort( 0x0 ), /* 0 */
/* 142 */ NdrFcShort( 0x8 ), /* 8 */
/* 144 */ 0x7, /* Oi2 Flags: srv must size, clt must
size, has return, */
0x3, /* 3 */
/* Parameter txn_in */
/* 146 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by
val, */
#ifndef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 148 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
#else
NdrFcShort( 0x8 ), /* MIPS &
PPC Stack size/offset = 8 */
#endif
#else
NdrFcShort( 0x8 ), /* Alpha Stack
size/offset = 8 */
#endif
/* 150 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */
/* Parameter txn_out */
/* 152 */ NdrFcShort( 0x4113 ), /* Flags: must size, must
free, out, simple ref, srv alloc size=16 */
#ifndef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 154 */ NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
#else
NdrFcShort( 0x18 ), /* MIPS &
PPC Stack size/offset = 24 */
#endif
#else
NdrFcShort( 0x18 ), /* Alpha Stack
size/offset = 24 */
#endif
/* 156 */ NdrFcShort( 0x3da ), /* Type Offset=986 */
/* Return value */
/* 158 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifndef _ALPHA_
#if !defined(_MIPS_) && !defined(_PPC_)
/* 160 */ NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
#else
NdrFcShort( 0x1c ), /* MIPS &
PPC Stack size/offset = 28 */
#endif
#else
NdrFcShort( 0x20 ), /* Alpha Stack
size/offset = 32 */
#endif
/* 162 */ 0x8, /* FC_LONG */
0x0, /* 0 */
/* Procedure CallSetComplete */
/* 164 */ 0x33, /* FC_AUTO_HANDLE */
object, Oi2 */
/* 166 */ NdrFcLong( 0x0 ), /* 0 */
/* 170 */ NdrFcShort( 0x8 ), /* 8 */
#ifndef _ALPHA_
/* 172 */ NdrFcShort( 0x8 ), /* x86, MIPS, PPC Stack size/offset
= 8 */
#else
NdrFcShort( 0x10 ), /* Alpha Stack
size/offset = 16 */
#endif
/* 174 */ NdrFcShort( 0x0 ), /* 0 */
/* 176 */ NdrFcShort( 0x8 ), /* 8 */
/* 178 */ 0x4, /* Oi2 Flags: has return, */
0x1, /* 1 */
/* Return value */
/* 180 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifndef _ALPHA_
/* 182 */ NdrFcShort( 0x4 ), /* x86, MIPS, PPC Stack size/offset
= 4 */
#else
NdrFcShort( 0x8 ), /* Alpha Stack
size/offset = 8 */
#endif
/* 184 */ 0x8, /* FC_LONG */
0x0, /* 0 */
0x0
}
};
static const MIDL_TYPE_FORMAT_STRING
_MIDL_TypeFormatString =
{
0,
{
NdrFcShort( 0x0 ), /* 0 */
/* 2 */
0x12, 0x0, /* FC_UP */
/* 4 */ NdrFcShort( 0x3b0 ), /* Offset= 944 (948) */
/* 6 */
0x2b, /*
FC_NON_ENCAPSULATED_UNION */
0x9, /* FC_ULONG
*/
/* 8 */ 0x7, /* Corr desc: FC_USHORT */
0x0, /*
*/
/* 10 */ NdrFcShort( 0xffff8 ), /* -8 */
/* 12 */ NdrFcShort( 0x2 ), /* Offset= 2 (14) */
/* 14 */ NdrFcShort( 0x10 ), /* 16 */
/* 16 */ NdrFcShort( 0x2b ), /* 43 */
/* 18 */ NdrFcLong( 0x3 ), /* 3 */
/* 22 */ NdrFcShort( 0x8008 ), /* Simple arm type:
FC_LONG */
/* 24 */ NdrFcLong( 0x11 ), /* 17 */
/* 28 */ NdrFcShort( 0x8001 ), /* Simple arm type:
FC_BYTE */
/* 30 */ NdrFcLong( 0x2 ), /* 2 */
/* 34 */ NdrFcShort( 0x8006 ), /* Simple arm type:
FC_SHORT */
/* 36 */ NdrFcLong( 0x4 ), /* 4 */
/* 40 */ NdrFcShort( 0x800a ), /* Simple arm type:
FC_FLOAT */
/* 42 */ NdrFcLong( 0x5 ), /* 5 */
/* 46 */ NdrFcShort( 0x800c ), /* Simple arm type:
FC_DOUBLE */
}
}

```

```

/* 48 */ NdrFcLong( 0xb ), /* 11 */
/* 52 */ NdrFcShort( 0x8006 ), /* Simple arm type:
FC_SHORT */
/* 54 */ NdrFcLong( 0xa ), /* 10 */
/* 58 */ NdrFcShort( 0x8008 ), /* Simple arm type:
FC_LONG */
/* 60 */ NdrFcLong( 0x6 ), /* 6 */
/* 64 */ NdrFcShort( 0xd6 ), /* Offset= 214 (278) */
/* 66 */ NdrFcLong( 0x7 ), /* 7 */
/* 70 */ NdrFcShort( 0x800c ), /* Simple arm type:
FC_DOUBLE */
/* 72 */ NdrFcLong( 0x8 ), /* 8 */
/* 76 */ NdrFcShort( 0xd0 ), /* Offset= 208 (284) */
/* 78 */ NdrFcLong( 0xd ), /* 13 */
/* 82 */ NdrFcShort( 0xe2 ), /* Offset= 226 (308) */
/* 84 */ NdrFcLong( 0x9 ), /* 9 */
/* 88 */ NdrFcShort( 0xee ), /* Offset= 238 (326) */
/* 90 */ NdrFcLong( 0x2000 ), /* 8192 */
/* 94 */ NdrFcShort( 0xfa ), /* Offset= 250 (344) */
/* 96 */ NdrFcLong( 0x24 ), /* 36 */
/* 100 */ NdrFcShort( 0x308 ), /* Offset= 776 (876) */
/* 102 */ NdrFcLong( 0x4024 ), /* 16420 */
/* 106 */ NdrFcShort( 0x302 ), /* Offset= 770 (876) */
/* 108 */ NdrFcLong( 0x4011 ), /* 16401 */
/* 112 */ NdrFcShort( 0x300 ), /* Offset= 768 (880) */
/* 114 */ NdrFcLong( 0x4002 ), /* 16386 */
/* 118 */ NdrFcShort( 0x2fe ), /* Offset= 766 (884) */
/* 120 */ NdrFcLong( 0x4003 ), /* 16387 */
/* 124 */ NdrFcShort( 0x2fc ), /* Offset= 764 (888) */
/* 126 */ NdrFcLong( 0x4004 ), /* 16388 */
/* 130 */ NdrFcShort( 0x2fa ), /* Offset= 762 (892) */
/* 132 */ NdrFcLong( 0x4005 ), /* 16389 */
/* 136 */ NdrFcShort( 0x2f8 ), /* Offset= 760 (896) */
/* 138 */ NdrFcLong( 0x400b ), /* 16395 */
/* 142 */ NdrFcShort( 0x2e6 ), /* Offset= 742 (884) */
/* 144 */ NdrFcLong( 0x400a ), /* 16394 */
/* 148 */ NdrFcShort( 0x2e4 ), /* Offset= 740 (888) */
/* 150 */ NdrFcLong( 0x4006 ), /* 16390 */
/* 154 */ NdrFcShort( 0x2ea ), /* Offset= 746 (900) */
/* 156 */ NdrFcLong( 0x4007 ), /* 16391 */
/* 160 */ NdrFcShort( 0x2e0 ), /* Offset= 736 (896) */
/* 162 */ NdrFcLong( 0x4008 ), /* 16392 */
/* 166 */ NdrFcShort( 0x2e2 ), /* Offset= 738 (904) */
/* 168 */ NdrFcLong( 0x400d ), /* 16397 */
/* 172 */ NdrFcShort( 0x2e0 ), /* Offset= 736 (908) */
/* 174 */ NdrFcLong( 0x4009 ), /* 16393 */
/* 178 */ NdrFcShort( 0x2de ), /* Offset= 734 (912) */
/* 180 */ NdrFcLong( 0x6000 ), /* 24576 */
/* 184 */ NdrFcShort( 0x2dc ), /* Offset= 732 (916) */
/* 186 */ NdrFcLong( 0x400c ), /* 16396 */
/* 190 */ NdrFcShort( 0x2da ), /* Offset= 730 (920) */
/* 192 */ NdrFcLong( 0x10 ), /* 16 */
/* 196 */ NdrFcShort( 0x8002 ), /* Simple arm type:
FC_CHAR */
/* 198 */ NdrFcLong( 0x12 ), /* 18 */
/* 202 */ NdrFcShort( 0x8006 ), /* Simple arm type:
FC_SHORT */
/* 204 */ NdrFcLong( 0x13 ), /* 19 */
/* 208 */ NdrFcShort( 0x8008 ), /* Simple arm type:
FC_LONG */
/* 210 */ NdrFcLong( 0x16 ), /* 22 */
/* 214 */ NdrFcShort( 0x8008 ), /* Simple arm type:
FC_LONG */
/* 216 */ NdrFcLong( 0x17 ), /* 23 */
/* 220 */ NdrFcShort( 0x8008 ), /* Simple arm type:
FC_LONG */
/* 222 */ NdrFcLong( 0xe ), /* 14 */
/* 226 */ NdrFcShort( 0x2be ), /* Offset= 702 (928) */
/* 228 */ NdrFcLong( 0x400e ), /* 16398 */
/* 232 */ NdrFcShort( 0x2c4 ), /* Offset= 708 (940) */
/* 234 */ NdrFcLong( 0x4010 ), /* 16400 */
/* 238 */ NdrFcShort( 0x2c2 ), /* Offset= 706 (944) */
/* 240 */ NdrFcLong( 0x4012 ), /* 16402 */
/* 244 */ NdrFcShort( 0x280 ), /* Offset= 640 (884) */
/* 246 */ NdrFcLong( 0x4013 ), /* 16403 */
/* 250 */ NdrFcShort( 0x27e ), /* Offset= 638 (888) */
/* 252 */ NdrFcLong( 0x4016 ), /* 16406 */
/* 256 */ NdrFcShort( 0x278 ), /* Offset= 632 (888) */
/* 258 */ NdrFcLong( 0x4017 ), /* 16407 */
/* 262 */ NdrFcShort( 0x272 ), /* Offset= 626 (888) */
/* 264 */ NdrFcLong( 0x0 ), /* 0 */
/* 268 */ NdrFcShort( 0x0 ), /* Offset= 0 (268) */
/* 270 */ NdrFcLong( 0x1 ), /* 1 */
/* 274 */ NdrFcShort( 0x0 ), /* Offset= 0 (274) */
/* 276 */ NdrFcShort( 0xffffffff ), /* Offset= -1 (275) */
/* 278 */
0x15, /*
FC_STRUCT */
0x7, /* 7 */
/* 280 */ NdrFcShort( 0x8 ), /* 8 */
/* 282 */ 0xb, /* FC_HYPER */
0x5b, /* FC_END */
/* 284 */
0x12, 0x0, /* FC_UP */
/* 286 */ NdrFcShort( 0xc ), /* Offset= 12 (298) */
/* 288 */
0x1b, /*
FC_CARRAY */
0x1, /* 1 */
/* 290 */ NdrFcShort( 0x2 ), /* 2 */
/* 292 */ 0x9, /* Corr desc: FC_ULONG */
0x0, /*
/* 294 */ NdrFcShort( 0xfffc ), /* -4 */
/* 296 */ 0x6, /* FC_SHORT */
0x5b, /* FC_END */
/* 298 */
0x17, /*
FC_CSTRUCT */
0x3, /* 3 */
/* 300 */ NdrFcShort( 0x8 ), /* 8 */
/* 302 */ NdrFcShort( 0xffffffff2 ), /* Offset= -14 (288) */
/* 304 */ 0x8, /* FC_LONG */
0x8, /* FC_LONG
*/
/* 306 */ 0x5c, /* FC_PAD */
0x5b, /* FC_END */
/* 308 */
0x2f, /* FC_IP */
0x5a, /*
FC_CONSTANT_IID */
/* 310 */ NdrFcLong( 0x0 ), /* 0 */
/* 314 */ NdrFcShort( 0x0 ), /* 0 */
/* 316 */ NdrFcShort( 0x0 ), /* 0 */
/* 318 */ 0xc0, /* 192 */
0x0, /* 0 */
/* 320 */ 0x0, /* 0 */
0x0, /* 0 */
/* 322 */ 0x0, /* 0 */
0x0, /* 0 */
/* 324 */ 0x0, /* 0 */
0x46, /* 70 */
/* 326 */
0x2f, /* FC_IP */
0x5a, /*
FC_CONSTANT_IID */
/* 328 */ NdrFcLong( 0x20400 ), /* 132096 */

```

```

/* 332 */ NdrFcShort( 0x0 ), /* 0 */
/* 334 */ NdrFcShort( 0x0 ), /* 0 */
/* 336 */ 0xc0, /* 192 */
/* 338 */ 0x0, /* 0 */
/* 340 */ 0x0, /* 0 */
/* 342 */ 0x0, /* 0 */
/* 344 */ /* 70 */
/* 346 */ NdrFcShort( 0x2 ), /* Offset= 2 (348) */
/* 348 */ /* FC_UP */
/* 350 */ NdrFcShort( 0x1fc ), /* Offset= 508 (858) */
/* 352 */ /* FC_UP */
FC_ENCAPSULATED_UNION /*
/* 354 */ NdrFcShort( 0x18 ), /* 24 */
/* 356 */ NdrFcShort( 0xa ), /* 10 */
/* 358 */ NdrFcLong( 0x8 ), /* 8 */
/* 362 */ NdrFcShort( 0x58 ), /* Offset= 88 (450) */
/* 364 */ NdrFcLong( 0xd ), /* 13 */
/* 368 */ NdrFcShort( 0x78 ), /* Offset= 120 (488) */
/* 370 */ NdrFcLong( 0x9 ), /* 9 */
/* 374 */ NdrFcShort( 0x94 ), /* Offset= 148 (522) */
/* 376 */ NdrFcLong( 0xc ), /* 12 */
/* 380 */ NdrFcShort( 0xbc ), /* Offset= 188 (568) */
/* 382 */ NdrFcLong( 0x24 ), /* 36 */
/* 386 */ NdrFcShort( 0x114 ), /* Offset= 276 (662) */
/* 388 */ NdrFcLong( 0x800d ), /* 32781 */
/* 392 */ NdrFcShort( 0x130 ), /* Offset= 304 (696) */
/* 394 */ NdrFcLong( 0x10 ), /* 16 */
/* 398 */ NdrFcShort( 0x148 ), /* Offset= 328 (726) */
/* 400 */ NdrFcLong( 0x2 ), /* 2 */
/* 404 */ NdrFcShort( 0x160 ), /* Offset= 352 (756) */
/* 406 */ NdrFcLong( 0x3 ), /* 3 */
/* 410 */ NdrFcShort( 0x178 ), /* Offset= 376 (786) */
/* 412 */ NdrFcLong( 0x14 ), /* 20 */
/* 416 */ NdrFcShort( 0x190 ), /* Offset= 400 (816) */
/* 418 */ NdrFcShort( 0xffffffff ), /* Offset= -1 (417) */
/* 420 */ /* FC_UP */
FC_CARRAY /*
/* 422 */ NdrFcShort( 0x4 ), /* 4 */
/* 424 */ 0x19, /* Corr desc: field pointer,
FC_ULONG */
/* 426 */ NdrFcShort( 0x0 ), /* 0 */
/* 428 */ /* FC_PP */
/* 430 */ /* FC_PAD */
/* 432 */ /* FC_PP */
FC_VARIABLE_REPEAT /*
/* 434 */ NdrFcShort( 0x4 ), /* 4 */
/* 436 */ NdrFcShort( 0x0 ), /* 0 */
/* 438 */ NdrFcShort( 0x1 ), /* 1 */
/* 440 */ NdrFcShort( 0x0 ), /* 0 */
/* 442 */ 0x12, 0x0, /* FC_UP */
/* 444 */ NdrFcShort( 0xffffffff6e ), /* Offset= -146 (298) */
/* 446 */
0x5b, /* FC_END */
0x8, /* FC_LONG
/* 448 */ 0x5c, /* FC_PAD */
0x5b, /* FC_END */
/* 450 */ /* FC_END */
0x16, /* FC_END */
FC_PSTRUCT /*
/* 452 */ NdrFcShort( 0x8 ), /* 8 */
/* 454 */ /* FC_PP */
/* 456 */ /* FC_PAD */
0x4b, /* FC_PP */
0x5c, /* FC_PAD */
0x46, /* FC_END */
FC_NO_REPEAT /*
/* 458 */ NdrFcShort( 0x4 ), /* 4 */
/* 460 */ NdrFcShort( 0x4 ), /* 4 */
/* 462 */ 0x11, 0x0, /* FC_RP */
/* 464 */ NdrFcShort( 0xffffffffd4 ), /* Offset= -44 (420) */
/* 466 */ /* FC_END */
0x5b, /* FC_END */
0x8, /* FC_LONG
/* 468 */ 0x8, /* FC_LONG */
0x5b, /* FC_END */
/* 470 */ /* FC_END */
0x21, /* FC_END */
FC_BOGUS_ARRAY /*
/* 472 */ NdrFcShort( 0x0 ), /* 0 */
/* 474 */ 0x19, /* Corr desc: field pointer,
FC_ULONG */
/* 476 */ NdrFcShort( 0x0 ), /* 0 */
/* 478 */ NdrFcLong( 0xffffffff ), /* -1 */
/* 482 */ 0x4c, /* FC_EMBEDDED_COMPLEX */
/* 484 */ NdrFcShort( 0xffffffff50 ), /* Offset= -176 (308) */
/* 486 */ 0x5c, /* FC_PAD */
/* 488 */ 0x5b, /* FC_END */
0x1a, /* FC_END */
FC_BOGUS_STRUCT /*
/* 490 */ NdrFcShort( 0x8 ), /* 8 */
/* 492 */ NdrFcShort( 0x0 ), /* 0 */
/* 494 */ NdrFcShort( 0x6 ), /* Offset= 6 (500) */
/* 496 */ 0x8, /* FC_LONG */
0x36, /* FC_END */
FC_POINTER /*
/* 498 */ 0x5c, /* FC_PAD */
0x5b, /* FC_END */
/* 500 */ /* FC_RP */
/* 502 */ NdrFcShort( 0xffffffffe0 ), /* Offset= -32 (470) */
/* 504 */ /* FC_END */
0x21, /* FC_END */
FC_BOGUS_ARRAY /*
/* 506 */ NdrFcShort( 0x0 ), /* 0 */
/* 508 */ 0x19, /* Corr desc: field pointer,
FC_ULONG */
/* 510 */ NdrFcShort( 0x0 ), /* 0 */

```

/* 512 */ NdrFcLong(0xffffffff), /* -1 */			0x2f,	/* FC_IP */
/* 516 */ 0x4c, /* FC_EMBEDDED_COMPLEX */			0x5a,	/*
	0x0, /* 0 */	FC_CONSTANT_IID */		
/* 518 */ NdrFcShort(0xffffffff40), /* Offset= -192 (326) */		/* 586 */ NdrFcLong(0x2f), /* 47 */		
/* 520 */ 0x5c, /* FC_PAD */		/* 590 */ NdrFcShort(0x0), /* 0 */		
	0x5b, /* FC_END */	/* 592 */ NdrFcShort(0x0), /* 0 */		
/* 522 */		/* 594 */ 0xc0, /* 192 */		
	0x1a, /*		0x0,	/* 0 */
FC_BOGUS_STRUCT */		/* 596 */ 0x0, /* 0 */		
	0x3, /* 3 */	/* 598 */ 0x0, /* 0 */	0x0,	/* 0 */
/* 524 */ NdrFcShort(0x8), /* 8 */			0x0,	/* 0 */
/* 526 */ NdrFcShort(0x0), /* 0 */		/* 600 */ 0x0, /* 0 */		
/* 528 */ NdrFcShort(0x6), /* Offset= 6 (534) */		/* 602 */	0x46,	/* 70 */
/* 530 */ 0x8, /* FC_LONG */				
	0x36, /*		0x1b,	/*
FC_POINTER */		FC_CARRAY */		
/* 532 */ 0x5c, /* FC_PAD */		/* 604 */ NdrFcShort(0x1), /* 1 */	0x0,	/* 0 */
	0x5b, /* FC_END */	/* 606 */ 0x19, /* Corr desc: field pointer,		
/* 534 */		FC_ULONG */		
	0x11, 0x0, /* FC_RP */		0x0,	/* */
/* 536 */ NdrFcShort(0xffffffffe0), /* Offset= -32 (504) */		/* 608 */ NdrFcShort(0x4), /* 4 */		
/* 538 */		/* 610 */ 0x1, /* FC_BYTE */	0x5b,	/* FC_END */
FC_CARRAY */		/* 612 */		
	0x3, /* 3 */	FC_BOGUS_STRUCT */	0x1a,	/*
/* 540 */ NdrFcShort(0x4), /* 4 */				
/* 542 */ 0x19, /* Corr desc: field pointer,		/* 614 */ NdrFcShort(0x10), /* 16 */	0x3,	/* 3 */
FC_ULONG */		/* 616 */ NdrFcShort(0x0), /* 0 */		
	0x0, /* */	/* 618 */ NdrFcShort(0xa), /* Offset= 10 (628) */		
/* 544 */ NdrFcShort(0x0), /* 0 */		/* 620 */ 0x8, /* FC_LONG */	0x8,	/* FC_LONG
/* 546 */				
	0x4b, /* FC_PP */	/* 622 */ 0x4c, /* FC_EMBEDDED_COMPLEX */		
/* 548 */	0x5c, /* FC_PAD */	0x0, /* 0 */		
		/* 624 */ NdrFcShort(0xffffffffd8), /* Offset= -40 (584) */		
FC_VARIABLE_REPEAT */		/* 626 */ 0x36, /* FC_POINTER */		
	0x48, /*	/* 628 */	0x5b,	/* FC_END */
	0x49, /*			
FC_FIXED_OFFSET */		0x12, 0x0, /* FC_UP */		
/* 550 */ NdrFcShort(0x4), /* 4 */		/* 630 */ NdrFcShort(0xffffffffe4), /* Offset= -28 (602) */		
/* 552 */ NdrFcShort(0x0), /* 0 */		/* 632 */	0x1b,	/*
/* 554 */ NdrFcShort(0x1), /* 1 */		FC_CARRAY */		
/* 556 */ NdrFcShort(0x0), /* 0 */		/* 634 */ NdrFcShort(0x4), /* 4 */	0x3,	/* 3 */
/* 558 */ NdrFcShort(0x0), /* 0 */		/* 636 */ 0x19, /* Corr desc: field pointer,		
/* 560 */ 0x12, 0x0, /* FC_UP */		FC_ULONG */		
/* 562 */ NdrFcShort(0xffffffffdce), /* Offset= -562 (0) */		/* 638 */ NdrFcShort(0x0), /* 0 */	0x0,	/* */
/* 564 */		/* 640 */		
	0x5b, /* FC_END */		0x4b,	/* FC_PP */
			0x5c,	/* FC_PAD */
	0x8, /* FC_LONG	/* 642 */		
/*		FC_VARIABLE_REPEAT */		
/* 566 */ 0x5c, /* FC_PAD */			0x48,	/*
/* 568 */	0x5b, /* FC_END */	FC_FIXED_OFFSET */		
		/* 644 */ NdrFcShort(0x4), /* 4 */		
FC_BOGUS_STRUCT */		/* 646 */ NdrFcShort(0x0), /* 0 */		
	0x1a, /*	/* 648 */ NdrFcShort(0x1), /* 1 */		
	0x3, /* 3 */	/* 650 */ NdrFcShort(0x0), /* 0 */		
/* 570 */ NdrFcShort(0x8), /* 8 */		/* 652 */ NdrFcShort(0x0), /* 0 */		
/* 572 */ NdrFcShort(0x0), /* 0 */		/* 654 */ 0x12, 0x0, /* FC_UP */		
/* 574 */ NdrFcShort(0x6), /* Offset= 6 (580) */		/* 656 */ NdrFcShort(0xffffffffd4), /* Offset= -44 (612) */		
/* 576 */ 0x8, /* FC_LONG */		/* 658 */		
	0x36, /*			
FC_POINTER */				
/* 578 */ 0x5c, /* FC_PAD */				
	0x5b, /* FC_END */			
/* 580 */				
	0x11, 0x0, /* FC_RP */			
/* 582 */ NdrFcShort(0xffffffffd4), /* Offset= -44 (538) */				
/* 584 */				

	0x5b,	/* FC_END */		0x0,	/* */
	0x8,	/* FC_LONG	/* 722 */ NdrFcShort(0x0),	/* 0 */	
/			/ 724 */ 0x1,	/* FC_BYTE */	
/* 660 */ 0x5c,	/* FC_PAD */		/* 726 */	0x5b,	/* FC_END */
/* 662 */	0x5b,	/* FC_END */		0x16,	/*
FC_BOGUS_STRUCT */	0x1a,	/*	FC_PSTRUCT */	0x3,	/* 3 */
	0x3,	/* 3 */	/* 728 */ NdrFcShort(0x8),	/* 8 */	
/* 664 */ NdrFcShort(0x8),	/* 8 */		/* 730 */	0x4b,	/* FC_PP */
/* 666 */ NdrFcShort(0x0),	/* 0 */			0x5c,	/* FC_PAD */
/* 668 */ NdrFcShort(0x6),	/* Offset= 6 (674) */		/* 732 */		
/* 670 */ 0x8,	/* FC_LONG */			0x46,	/*
FC_POINTER */	0x36,	/*	FC_NO_REPEAT */	0x5c,	/* FC_PAD */
/* 672 */ 0x5c,	/* FC_PAD */		/* 734 */ NdrFcShort(0x4),	/* 4 */	
	0x5b,	/* FC_END */	/* 736 */ NdrFcShort(0x4),	/* 4 */	
/* 674 */			/* 738 */ 0x12, 0x0, /* FC_UP */		
	0x11, 0x0, /* FC_RP */		/* 740 */ NdrFcShort(0xfffffe8),	/* Offset= -24 (716) */	
/* 676 */ NdrFcShort(0xfffffd4),	/* Offset= -44 (632) */		/* 742 */		
/* 678 */				0x5b,	/* FC_END */
FC_SMFARRAY */	0x1d,	/*		0x8,	/* FC_LONG
	0x0,	/* 0 */	*/		
/* 680 */ NdrFcShort(0x8),	/* 8 */		/* 744 */ 0x8,	/* FC_LONG */	
/* 682 */ 0x2,	/* FC_CHAR */			0x5b,	/* FC_END */
	0x5b,	/* FC_END */	/* 746 */		
/* 684 */				0x1b,	/*
FC_STRUCT */	0x15,	/*	FC_CARRAY */	0x1,	/* 1 */
	0x3,	/* 3 */	/* 748 */ NdrFcShort(0x2),	/* 2 */	
/* 686 */ NdrFcShort(0x10),	/* 16 */		/* 750 */ 0x19,	/* Corr desc: field pointer,	
/* 688 */ 0x8,	/* FC_LONG */		FC_ULONG */		
/	0x6,	/ FC_SHORT		0x0,	/* */
/* 690 */ 0x6,	/* FC_SHORT */		/* 752 */ NdrFcShort(0x0),	/* 0 */	
FC_EMBEDDED_COMPLEX */	0x4c,	/*	/* 754 */ 0x6,	/* FC_SHORT */	
/* 692 */ 0x0,	/* 0 */		/* 756 */	0x5b,	/* FC_END */
Offset= -15 (678) */	NdrFcShort(0xfffff1),	/*	FC_PSTRUCT */	0x16,	/*
	0x5b,	/* FC_END */	/* 758 */ NdrFcShort(0x8),	/* 8 */	
/* 696 */			/* 760 */		
FC_BOGUS_STRUCT */	0x1a,	/*		0x4b,	/* FC_PP */
	0x3,	/* 3 */	/* 762 */	0x5c,	/* FC_PAD */
/* 698 */ NdrFcShort(0x18),	/* 24 */			0x46,	/*
/* 700 */ NdrFcShort(0x0),	/* 0 */		FC_NO_REPEAT */		
/* 702 */ NdrFcShort(0xa),	/* Offset= 10 (712) */		/* 764 */ NdrFcShort(0x4),	/* 4 */	
/* 704 */ 0x8,	/* FC_LONG */		/* 766 */ NdrFcShort(0x4),	/* 4 */	
	0x36,	/*	/* 768 */ 0x12, 0x0, /* FC_UP */		
FC_POINTER */			/* 770 */ NdrFcShort(0xfffffe8),	/* Offset= -24 (746) */	
/* 706 */ 0x4c,	/* FC_EMBEDDED_COMPLEX */		/* 772 */		
	0x0,	/* 0 */		0x5b,	/* FC_END */
/* 708 */ NdrFcShort(0xfffffe8),	/* Offset= -24 (684) */			0x8,	/* FC_LONG
/* 710 */ 0x5c,	/* FC_PAD */		*/		
/* 712 */	0x5b,	/* FC_END */	/* 774 */ 0x8,	/* FC_LONG */	
	0x11, 0x0, /* FC_RP */			0x5b,	/* FC_END */
/* 714 */ NdrFcShort(0xfffff0c),	/* Offset= -244 (470) */		/* 776 */		
/* 716 */				0x1b,	/*
FC_CARRAY */	0x0,	/* 0 */	FC_CARRAY */	0x3,	/* 3 */
/* 718 */ NdrFcShort(0x1),	/* 1 */		/* 778 */ NdrFcShort(0x4),	/* 4 */	
/* 720 */ 0x19,	/* Corr desc: field pointer,		/* 780 */ 0x19,	/* Corr desc: field pointer,	
FC_ULONG */			FC_ULONG */		

/* 782 */ NdrFcShort(0x0),	0x0,	/* */	/*	0x8,	/* FC_LONG
/* 784 */ 0x8,	/* FC_LONG */		/* 842 */ 0x5c,	/* FC_PAD */	
/* 786 */	0x5b,	/* FC_END */	/* 844 */	0x5b,	/* FC_END */
FC_PSTRUCT */	0x16,	/*	FC_CARRAY */	0x1b,	/*
/* 788 */ NdrFcShort(0x8),	0x3,	/* 3 */	/* 846 */ NdrFcShort(0x8),	0x3,	/* 3 */
/* 790 */	/* 8 */		/* 848 */ 0x7,	/* 8 */	
/* 792 */	0x4b,	/* FC_PP */	/* 850 */ NdrFcShort(0xffd8),	0x0,	/*
	0x5c,	/* FC_PAD */	/* 852 */ 0x4c,	/* FC_EMBEDDED_COMPLEX */	
FC_NO_REPEAT */	0x46,	/*	/* 854 */ NdrFcShort(0xfffffee),	/* 0 */	
/* 794 */ NdrFcShort(0x4),	0x5c,	/* FC_PAD */	/* 856 */ 0x5c,	/* Offset= -18 (836) */	
/* 796 */ NdrFcShort(0x4),	/* 4 */		/* 858 */	/* FC_PAD */	
/* 798 */ 0x12, 0x0, /* FC_UP */	/* 4 */		0x5b,	/* FC_END */	
/* 800 */ NdrFcShort(0xfffffe8),	/* Offset= -24 (776) */		0x1a,	/*	
/* 802 */			FC_BOGUS_STRUCT */		
	0x5b,	/* FC_END */	/* 860 */ NdrFcShort(0x28),	0x3,	/* 3 */
	0x8,	/* FC_LONG	/* 862 */ NdrFcShort(0xfffffee),	/* 40 */	
/*	/* FC_LONG */		/* 864 */ NdrFcShort(0x0),	/* Offset= -18 (844) */	
/* 804 */ 0x8,	/* FC_LONG */		/* 866 */ 0x6,	/* Offset= 0 (864) */	
/* 806 */	0x5b,	/* FC_END */	/* FC_SHORT */	0x6,	/* FC_SHORT
	0x1b,	/*	/*		
FC_CARRAY */	0x7,	/* 7 */	/* 868 */ 0x38,	/* FC_ALIGNM4 */	
/* 808 */ NdrFcShort(0x8),	/* 8 */		0x8,	/* FC_LONG */	
/* 810 */ 0x19,	/* Corr desc: field pointer,		/* 870 */ 0x8,	0x4c,	/*
FC_ULONG */	0x0,	/* */	FC_EMBEDDED_COMPLEX */	/* 0 */	
/* 812 */ NdrFcShort(0x0),	/* 0 */		/* 872 */ 0x0,	NdrFcShort(0xfffffd7),	/*
/* 814 */ 0xb,	/* FC_HYPER */		Offset= -521 (352) */		
/* 816 */	0x5b,	/* FC_END */	/* 876 */	0x5b,	/* FC_END */
FC_PSTRUCT */	0x16,	/*	/* 878 */ NdrFcShort(0xfffffef6),	/* FC_UP */	
/* 818 */ NdrFcShort(0x8),	0x3,	/* 3 */	/* 880 */	0x12, 0x8, /* FC_UP	
/* 820 */	/* 8 */		[simple_pointer] */		
/* 822 */	0x4b,	/* FC_PP */	/* 882 */ 0x1,	/* FC_BYTE */	
	0x5c,	/* FC_PAD */	/* 884 */	0x5c,	/* FC_PAD */
FC_NO_REPEAT */	0x46,	/*	[simple_pointer] */	0x12, 0x8, /* FC_UP	
/* 824 */ NdrFcShort(0x4),	0x5c,	/* FC_PAD */	/* 886 */ 0x6,	/* FC_SHORT */	
/* 826 */ NdrFcShort(0x4),	/* 4 */		/* 888 */	0x5c,	/* FC_PAD */
/* 828 */ 0x12, 0x0, /* FC_UP */	/* 4 */		0x12, 0x8, /* FC_UP		
/* 830 */ NdrFcShort(0xfffffe8),	/* Offset= -24 (806) */		[simple_pointer] */		
/* 832 */	0x5b,	/* FC_END */	/* 890 */ 0x8,	/* FC_LONG */	
	0x8,	/* FC_LONG	/* 892 */	0x5c,	/* FC_PAD */
/*	/* FC_LONG */		[simple_pointer] */	0x12, 0x8, /* FC_UP	
/* 834 */ 0x8,	/* FC_LONG */		/* 894 */ 0xa,	/* FC_FLOAT */	
/* 836 */	0x5b,	/* FC_END */	/* 896 */	0x5c,	/* FC_PAD */
FC_STRUCT */	0x15,	/*	[simple_pointer] */	0x12, 0x8, /* FC_UP	
/* 838 */ NdrFcShort(0x8),	0x3,	/* 3 */	/* 898 */ 0xc,	/* FC_DOUBLE */	
/* 840 */ 0x8,	/* 8 */		/* 900 */	0x5c,	/* FC_PAD */
	/* FC_LONG */		/* 902 */ NdrFcShort(0xfffffd90),	/* FC_UP */	
			/* Offset= -624 (278) */		

```

/* 904 */
                                0x12, 0x10, /* FC_UP */
/* 906 */ NdrFcShort( 0xffffd92 ), /* Offset= -622 (284) */
/* 908 */
                                0x12, 0x10, /* FC_UP */
/* 910 */ NdrFcShort( 0xffffda6 ), /* Offset= -602 (308) */
/* 912 */
                                0x12, 0x10, /* FC_UP */
/* 914 */ NdrFcShort( 0xffffdb4 ), /* Offset= -588 (326) */
/* 916 */
                                0x12, 0x10, /* FC_UP */
/* 918 */ NdrFcShort( 0xffffdc2 ), /* Offset= -574 (344) */
/* 920 */
                                0x12, 0x10, /* FC_UP */
/* 922 */ NdrFcShort( 0x2 ), /* Offset= 2 (924) */
/* 924 */
                                0x12, 0x0, /* FC_UP */
/* 926 */ NdrFcShort( 0xffffc62 ), /* Offset= -926 (0) */
/* 928 */
                                0x15, /*
FC_STRUCT */
                                0x7, /* 7 */
/* 930 */ NdrFcShort( 0x10 ), /* 16 */
/* 932 */ 0x6, /* FC_SHORT */
                                0x1, /* FC_BYTE */
/*
/* 934 */ 0x1, /* FC_BYTE */
                                0x38, /*
FC_ALIGNM4 */
/* 936 */ 0x8, /* FC_LONG */
                                0x39, /*
FC_ALIGNM8 */
/* 938 */ 0xb, /* FC_HYPER */
                                0x5b, /* FC_END */
/* 940 */
                                0x12, 0x0, /* FC_UP */
/* 942 */ NdrFcShort( 0xfffff2 ), /* Offset= -14 (928) */
/* 944 */
                                0x12, 0x8, /* FC_UP */
[simple_pointer] */
/* 946 */ 0x2, /* FC_CHAR */
                                0x5c, /* FC_PAD */
/* 948 */
                                0x1a, /*
FC_BOGUS_STRUCT */
                                0x7, /* 7 */
/* 950 */ NdrFcShort( 0x20 ), /* 32 */
/* 952 */ NdrFcShort( 0x0 ), /* 0 */
/* 954 */ NdrFcShort( 0x0 ), /* Offset= 0 (954) */
/* 956 */ 0x8, /* FC_LONG */
                                0x8, /* FC_LONG */
/*
/* 958 */ 0x6, /* FC_SHORT */
                                0x6, /* FC_SHORT */
/*
/* 960 */ 0x6, /* FC_SHORT */
                                0x6, /* FC_SHORT */
/*
/* 962 */ 0x4c, /* FC_EMBEDDED_COMPLEX */
                                0x0, /* 0 */
/* 964 */ NdrFcShort( 0xffffc42 ), /* Offset= -958 (6) */
/* 966 */ 0x5c, /* FC_PAD */
                                0x5b, /* FC_END */
/* 968 */ 0xb4, /* FC_USER_MARSHAL */
                                0x83, /* 131 */
/* 970 */ NdrFcShort( 0x0 ), /* 0 */
/* 972 */ NdrFcShort( 0x10 ), /* 16 */
/* 974 */ NdrFcShort( 0x0 ), /* 0 */
/* 976 */ NdrFcShort( 0xffffc32 ), /* Offset= -974 (2) */
/* 978 */
                                0x11, 0x4, /* FC_RP */
[allocated_on_stack] */
/* 980 */ NdrFcShort( 0x6 ), /* Offset= 6 (986) */
/* 982 */
                                0x13, 0x0, /* FC_OP */
/* 984 */ NdrFcShort( 0xfffff4c ), /* Offset= -36 (948) */
/* 986 */ 0xb4, /* FC_USER_MARSHAL */
                                0x83, /* 131 */
/* 988 */ NdrFcShort( 0x0 ), /* 0 */
/* 990 */ NdrFcShort( 0x10 ), /* 16 */
/* 992 */ NdrFcShort( 0x0 ), /* 0 */
/* 994 */ NdrFcShort( 0xfffff4 ), /* Offset= -12 (982) */
                                0x0
                                }
                                };
const CInterfaceProxyVtbl* _tpcc_com_remote_ps_ProxyVtblList[] =
{
( CInterfaceProxyVtbl* ) &_ITPCCremProxyVtbl,
0
};
const CInterfaceStubVtbl* _tpcc_com_remote_ps_StubVtblList[] =
{
( CInterfaceStubVtbl* ) &_ITPCCremStubVtbl,
0
};
PCInterfaceName const _tpcc_com_remote_ps_InterfaceNamesList[] =
{
"ITPCCrem",
0
};
#define _tpcc_com_remote_ps_CHECK_IID(n)
IID_GENERIC_CHECK_IID(_tpcc_com_remote_ps, pIID, n)
int __stdcall _tpcc_com_remote_ps_IID_Lookup( const IID* pIID, int
*pIndex )
{
if(!_tpcc_com_remote_ps_CHECK_IID(0))
{
*pIndex = 0;
return 1;
}
return 0;
}
const ExtendedProxyFileInfo tpcc_com_remote_ps_ProxyFileInfo =
{
(PCInterfaceProxyVtblList*) &
_tpcc_com_remote_ps_ProxyVtblList,
(PCInterfaceStubVtblList*) &_tpcc_com_remote_ps_StubVtblList,
(const PCInterfaceName*) &
_tpcc_com_remote_ps_InterfaceNamesList,
0, // no delegation
&_tpcc_com_remote_ps_IID_Lookup,
1,
2,
0, /* table of [async_uuid] interfaces */
}

```

```

0, /* Filler1 */
0, /* Filler2 */
0 /* Filler3 */
};

```

tpcc_com_remote_resource.h

```

///<{NO_DEPENDENCIES}
// Microsoft Developer Studio generated include file.
// Used by tpcc_com_remote.rc
//
#define IDS_PROJNAME            100
#define IDR_TPCCrem            101

// Next default values for new objects
//
#ifdef APSTUDIO_INVOKED
#ifndef APSTUDIO_READONLY_SYMBOLS
#define _APS_NEXT_RESOURCE_VALUE    202
#define _APS_NEXT_COMMAND_VALUE    32768
#define _APS_NEXT_CONTROL_VALUE    201
#define _APS_NEXT_SYMED_VALUE    102
#endif
#endif

```

tpcc_com_sl.rgs

```

HKCR
{
    TPCC.StockLevel.1 = s 'StockLevel Class'
    {
        CLSID = s
        '{2668369E-A50D-11D2-BA4E-00C04FBFE08B}'
    }
    TPCC.StockLevel = s 'StockLevel Class'
    {
        CurVer = s 'TPCC.StockLevel.1'
    }
    NoRemove CLSID
    {
        ForceRemove
        {2668369E-A50D-11D2-BA4E-00C04FBFE08B} = s 'StockLevel
        Class'
    }
    ProgID = s 'TPCC.StockLevel.1'
    VersionIndependentProgID = s
    'TPCC.StockLevel'
    InprocServer32 = s '%MODULE%'
    {
        val ThreadingModel = s
    }
}
}

```

tpcc_odbc.cpp

```

/* FILE: TPCC_ODBC.CPP
 * Microsoft TPC-C Kit Ver.
4.20.000
 * Copyright Microsoft,
1999
 * All Rights Reserved

```

```

 *
 * Version 4.10.000 audited
 * by Richard Gimarc, Performance Metrics, 3/17/99
 *
 * PURPOSE: Implements ODBC calls for TPC-C
 * txns.
 * Contact: Charles Levine (clevine@microsoft.com)
 *
 * Change history:
 * 4.30.000 - changes for w_id > 32767 and
 * multinode options.
 * 4.20.000 - updated rev number to match kit
 * 4.10.001 - not deleting error class in catch
 * handler on deadlock retry;
 * not a functional bug, but
 * a memory leak
 */
#include <windows.h>
#include <stdio.h>
#include <assert.h>

#define DBNTWIN32
#include <sqltypes.h>
#include <sql.h>
#include <sqllex.h>
#include <odbcss.h>

#ifdef ICECAP
#include <icapexp.h>
#endif

// need to declare functions for export
#define DllDecl __declspec( dllexport )

#include "..\..\common\src\error.h"
#include "..\..\common\src\trans.h"
#include "..\..\common\src\txn_base.h"
#include "tpcc_odbc.h"

// version string; must match return value from tpcc_version stored
// proc
const char sVersion[] = "4.10.000";

const iMaxRetries = 10; // how many retries on
// deadlock

const int iErrOleDbProvider = 7312;
const char sErrTimeoutExpired[] = "Timeout expired";

#define ErrTrace 1

#ifdef ErrTrace
FILE *errFile;
#endif

static SQLHENV henv = SQL_NULL_HENV;
// ODBC environment handle

BOOL APIENTRY DllMain(HMODULE hModule, DWORD
ul_reason_for_call, LPVOID lpReserved)
{
    switch( ul_reason_for_call )
    {
        case DLL_PROCESS_ATTACH:
            DisableThreadLibraryCalls(hModule);
    }
}

```



```

        if (
SQLAllocHandleStd(SQL_HANDLE_ENV,SQL_NULL_HANDLE,
&henv) != SQL_SUCCESS )
        return FALSE;
#ifdef ErrTrace
        errFile =
fopen("c:\\odbcerrfile.txt","a");
#endif
        break;
        case DLL_PROCESS_DETACH:
        if (henv != NULL)
            SQLFreeEnv(henv);
        break;
        default:
            /* nothing */;
    }
    return TRUE;
}

/* FUNCTION: CTPCC_ODBC_ERR::ErrorText
*
*/

char* CTPCC_ODBC_ERR::ErrorText(void)
{
    int i;

    static SERRORMSG errorMsgs[] =
    {
        { ERR_WRONG_SP_VERSION,
"Wrong version of stored procs on database server" },
        { ERR_INVALID_CUST,
"Invalid Customer id,name." },
        { ERR_NO_SUCH_ORDER,
orders found for customer." },
        { ERR_RETRIED_TRANS,
"Retries before transaction succeeded." },
        { 0,
"" }
    };

    static char szNotFound[] = "Unknown error number.";

    for(i=0; errorMsgs[i].szMsg[0]; i++)
    {
        if ( m_errno == errorMsgs[i].iError )
            break;
    }
    if ( !errorMsgs[i].szMsg[0] )
        return szNotFound;
    else
        return errorMsgs[i].szMsg;
}

// wrapper routine for class constructor
__declspec(dllexport) CTPCC_ODBC* CTPCC_ODBC_new(
    LPCSTR szServer, // name of SQL server
    LPCSTR szUser, // user name
    LPCSTR szPassword, // password for login
    LPCSTR szHost, // not used
    LPCSTR szDatabase ) // name of database to use
{
    return new CTPCC_ODBC( szServer, szUser, szPassword,
szHost, szDatabase );
}

CTPCC_ODBC::CTPCC_ODBC (
    LPCSTR szServer, //
name of SQL server
    LPCSTR szUser, //
user name for login
    LPCSTR szPassword, //
password for login
    LPCSTR szHost, //
not used
    LPCSTR szDatabase //
name of database to use
)
{
    RETCODE rc;

    // initialization
    m_hdbc = SQL_NULL_HDBC;
    m_hstmt = SQL_NULL_HSTMT;

    m_hstmtNewOrder = SQL_NULL_HSTMT;
    m_hstmtPayment = SQL_NULL_HSTMT;
    m_hstmtDelivery = SQL_NULL_HSTMT;
    m_hstmtOrderStatus = SQL_NULL_HSTMT;
    m_hstmtStockLevel = SQL_NULL_HSTMT;

    m_descNewOrderCols1 = SQL_NULL_HDESC;
    m_descNewOrderCols2 = SQL_NULL_HDESC;
    m_descOrderStatusCols1 = SQL_NULL_HDESC;
    m_descOrderStatusCols2 = SQL_NULL_HDESC;

    if ( SQLAllocHandle(SQL_HANDLE_DBC,henv,
&m_hdbc) != SQL_SUCCESS )
        ThrowError(CODBCERR::eAllocHandle);

    if ( SQLSetConnectOption(m_hdbc, SQL_PACKET_SIZE,
4096) != SQL_SUCCESS )
        ThrowError(CODBCERR::eConnOption);

    {
        char
szConnectStr[256];
        char
szOutStr[1024];
        SQLSMALLINT iOutStrLen;

        sprintf( szConnectStr, "DRIVER=SQL
Server;SERVER=%s;UID=%s;PWD=%s;DATABASE=%s",
szServer, szUser, szPassword,
szDatabase );

        rc = SQLDriverConnect(m_hdbc, NULL,
(SQLCHAR*)szConnectStr, sizeof(szConnectStr),
(SQLCHAR*)szOutStr,
sizeof(szOutStr), &iOutStrLen, SQL_DRIVER_NOPROMPT);

        if (rc != SQL_SUCCESS && rc !=
SQL_SUCCESS_WITH_INFO)
            ThrowError(CODBCERR::eConnect);
    }
}

```

```

        if (SQLAllocHandle(SQL_HANDLE_STMT,m_hdbc,
&m_hstmt) != SQL_SUCCESS)
            ThrowError(CODBCERR::eAllocHandle);

        {

            // set some options affecting connection behavior
            char            buffer[128];

            // set some options affecting connection behavior
            strcpy(buffer, "set nocount on ");
            strcat(buffer, "set XACT_ABORT ON ");

            // for coyote
            strcat(buffer, "set ansi_warnings on ");
            strcat(buffer, "set ansi_nulls on ");
            rc = SQLExecDirect(m_hstmt, (unsigned char
*)buffer, SQL_NTS);
            if (rc != SQL_SUCCESS && rc !=
SQL_SUCCESS_WITH_INFO)
                ThrowError(CODBCERR::eExecDirect);

            // verify that version of stored procs on server is
correct
            char db_sp_version[10];
            strcpy(buffer, "{call tpcc_version}");
            rc = SQLExecDirect(m_hstmt, (unsigned char
*)buffer, SQL_NTS);
            if (rc != SQL_SUCCESS && rc !=
SQL_SUCCESS_WITH_INFO)
                ThrowError(CODBCERR::eExecDirect);
            if ( SQLBindCol(m_hstmt, 1, SQL_C_CHAR,
&db_sp_version, sizeof(db_sp_version), NULL) != SQL_SUCCESS )
                ThrowError(CODBCERR::eBindCol);
            if ( SQLFetch(m_hstmt) == SQL_ERROR )
                ThrowError(CODBCERR::eFetch);
            if (strcmp(db_sp_version,sVersion)
                throw new CTPCC_ODBC_ERR(
CTPCC_ODBC_ERR::ERR_WRONG_SP_VERSION);

            SQLFreeHandle(SQL_HANDLE_STMT,
m_hstmt);
        }

        // Bind parameters for each of the transactions
        InitNewOrderParams();
        InitPaymentParams();
        InitOrderStatusParams();
        InitDeliveryParams();
        InitStockLevelParams();
    }

CTPCC_ODBC::~~CTPCC_ODBC( void )
{
    // note: descriptors are automatically released when the
connection is dropped
    SQLFreeHandle(SQL_HANDLE_STMT,
m_hstmtNewOrder);
    SQLFreeHandle(SQL_HANDLE_STMT,
m_hstmtPayment);
    SQLFreeHandle(SQL_HANDLE_STMT,
m_hstmtDelivery);
    SQLFreeHandle(SQL_HANDLE_STMT,
m_hstmtOrderStatus);

```

```

        SQLFreeHandle(SQL_HANDLE_STMT,
m_hstmtStockLevel);

        SQLDisconnect(m_hdbc);
        SQLFreeHandle(SQL_HANDLE_DBC,m_hdbc);
    }

void CTPCC_ODBC::ThrowError( CODBCERR::ACTION eAction )
{
    RETCODE            rc;
    SDWORD            INativeError;
    char            szState[6];
    char
    szMsg[SQL_MAX_MESSAGE_LENGTH];
    char
    szTmp[6*SQL_MAX_MESSAGE_LENGTH];
    CODBCERR            *pODBCerr;
    //
    not allocated until needed (maybe never)

    pODBCerr = new CODBCERR();

    pODBCerr->m_NativeError = 0;
    pODBCerr->m_eAction = eAction;
    pODBCerr->m_bDeadLock = FALSE;

    szTmp[0] = 0;
    while (TRUE)
    {
        rc = SQLError(henv, m_hdbc, m_hstmt, (BYTE
*)&szState, &INativeError,
                                (BYTE
*)&szMsg, sizeof(szMsg), NULL);
        if (rc == SQL_NO_DATA)
            break;

        // check for deadlock
        if (INativeError == 1205 || (INativeError ==
iErrOleDbProvider &&
                                strstr(szMsg, sErrTimeoutExpired) !=
NULL))
            pODBCerr->m_bDeadLock =
TRUE;

        // capture the (first) database error
        if (pODBCerr->m_NativeError == 0 &&
            pODBCerr->m_NativeError =

        // quit if there isn't enough room to concatenate
error text
        if ( ( strlen(szMsg) + 2) > (sizeof(szTmp) -
strlen(szTmp)) )
        {
#ifdef ErrTrace
            fprintf(errFile, "%s\n",szMsg);
            fflush(errFile);

            break;
        }

        // include line break after first error msg
        if (szTmp[0] != 0)
            strcat( szTmp, "\n");
        strcat( szTmp, szMsg );
    }
}

```

```

        if (pODBCErr->m_odbcerrstr != NULL)
        {
            delete [] pODBCErr->m_odbcerrstr;
            pODBCErr->m_odbcerrstr = NULL;
        }

        if (strlen(szTmp) > 0)
        {
            pODBCErr->m_odbcerrstr = new char[
strlen(szTmp)+1 ];
            strcpy( pODBCErr->m_odbcerrstr, szTmp );
#ifdef ErrTrace
            fprintf(errFile,"%s\n",szTmp);
            fflush(errFile);
#endif
        }

        SQLFreeStmt(m_hstmt, SQL_CLOSE);
        throw pODBCErr;
    }

void CTPCC_ODBC::InitStockLevelParams()
{
    if ( SQLAllocHandle(SQL_HANDLE_STMT,m_hdbc,
&m_hstmtStockLevel) != SQL_SUCCESS )
        ThrowError(CODBCERR::eAllocHandle);

    m_hstmt = m_hstmtStockLevel;

    int i = 0;
    if ( SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT,SQL_C_SLONG,SQL_INTEGER,0,0,
&m_txn.StockLevel.w_id,0, NULL) != SQL_SUCCESS
        || SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT,SQL_C_UTINYINT,SQL_TINYINT,0,0,
&m_txn.StockLevel.d_id,0, NULL) != SQL_SUCCESS
        || SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT,SQL_C_SSHORT,SQL_SMALLINT,0,0,
&m_txn.StockLevel.threshold,0, NULL) != SQL_SUCCESS
        )
        ThrowError(CODBCERR::eBindParam);

    if ( SQLBindCol(m_hstmt, 1, SQL_C_SLONG,
&m_txn.StockLevel.low_stock,0, NULL) != SQL_SUCCESS )
        ThrowError(CODBCERR::eBindCol);
}

void CTPCC_ODBC::StockLevel()
{
    RETCODE          rc;
    int              iTryCount = 0;

    m_hstmt = m_hstmtStockLevel;

    while (TRUE)
    {
        try
        {
            rc = SQLExecDirectW(m_hstmt,
(SQLWCHAR*)L"{call tpcc_stocklevel(?,?,?)", SQL_NTS);
            if (rc != SQL_SUCCESS && rc !=
SQL_SUCCESS_WITH_INFO)
                ThrowError(CODBCERR::eExecDirect);

            if ( SQLFetch(m_hstmt) ==
SQL_ERROR )

```

```

ThrowError(CODBCERR::eFetch);

                SQLFreeStmt(m_hstmt,
SQL_CLOSE);

                m_txn.StockLevel.exec_status_code
= eOK;
                break;
            }
            catch (CODBCERR *e)
            {
                if ((!e->m_bDeadLock) ||
(++iTryCount > iMaxRetries))
                    throw;

                // hit deadlock; backoff for
                // increasingly longer period
                delete e;
                Sleep(10 * iTryCount);
            }
        }
    }

    // if (iTryCount)
    // throw new

    CTPCC_ODBC_ERR(CTPCC_ODBC_ERR::ERR_RETRIED_TRAN
S, iTryCount);
}

void CTPCC_ODBC::InitNewOrderParams()
{
    if ( SQLAllocHandle(SQL_HANDLE_STMT,m_hdbc,
&m_hstmtNewOrder) != SQL_SUCCESS
        || SQLAllocHandle(SQL_HANDLE_DESC,
m_hdbc, &m_descNewOrderCols1) != SQL_SUCCESS
        || SQLAllocHandle(SQL_HANDLE_DESC,
m_hdbc, &m_descNewOrderCols2) != SQL_SUCCESS
        )
        ThrowError(CODBCERR::eAllocHandle);

    m_hstmt = m_hstmtNewOrder;

    if ( SQLSetStmtAttrW( m_hstmt,
SQL_ATTR_APP_ROW_DESC, m_descNewOrderCols1,
SQL_IS_POINTER) != SQL_SUCCESS )
        ThrowError(CODBCERR::eSetStmtAttr);

    int i = 0;
    if ( SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT,SQL_C_SLONG,SQL_INTEGER,0,0,
&m_txn.NewOrder.w_id,0, NULL) != SQL_SUCCESS
        || SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT,SQL_C_UTINYINT,SQL_TINYINT,0,0,
&m_txn.NewOrder.d_id,0, NULL) != SQL_SUCCESS
        || SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT,SQL_C_SLONG,SQL_INTEGER,0,0,
&m_txn.NewOrder.c_id,0, NULL) != SQL_SUCCESS
        || SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT,SQL_C_UTINYINT,SQL_TINYINT,0,0,
&m_txn.NewOrder.o_ol_cnt,0, NULL) != SQL_SUCCESS
        || SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT,SQL_C_UTINYINT,SQL_TINYINT,0,0,
&m_txn.NewOrder.o_all_local,0, NULL) != SQL_SUCCESS
        )
        ThrowError(CODBCERR::eBindParam);

    for (int j=0; j<MAX_OL_NEW_ORDER_ITEMS;j++)

```

```

        {
            if ( SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT, SQL_C_SLONG, SQL_INTEGER, 0, 0,
&m_txn.NewOrder.OL[j].ol_i_id, 0, NULL) != SQL_SUCCESS
                || SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT, SQL_C_SLONG, SQL_INTEGER, 0, 0,
&m_txn.NewOrder.OL[j].ol_supply_w_id, 0, NULL) !=
SQL_SUCCESS
                || SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT, SQL_C_SSHORT, SQL_SMALLINT, 0, 0,
&m_txn.NewOrder.OL[j].ol_quantity, 0, NULL) != SQL_SUCCESS
            )
                ThrowError(CODBCERR::eBindParam);
        }
#ifdef new_order_strstr
        // set the bind offset pointer
        if ( SQLSetStmtAttrW( m_hstmt,
SQL_ATTR_ROW_BIND_OFFSET_PTR, &m_BindOffset,
SQL_IS_POINTER) != SQL_SUCCESS )
            ThrowError(CODBCERR::eSetStmtAttr);
        i = 0;
        if ( SQLBindCol(m_hstmt, ++i, SQL_C_CHAR,
&m_txn.NewOrder.OL[0].ol_i_name,
sizeof(m_txn.NewOrder.OL[0].ol_i_name), NULL) !=
SQL_SUCCESS
            || SQLBindCol(m_hstmt, ++i,
SQL_C_SSHORT, &m_txn.NewOrder.OL[0].ol_stock, 0, NULL) !=
SQL_SUCCESS
            || SQLBindCol(m_hstmt, ++i, SQL_C_CHAR,
&m_txn.NewOrder.OL[0].ol_brand_generic,
sizeof(m_txn.NewOrder.OL[0].ol_brand_generic), NULL) !=
SQL_SUCCESS
            || SQLBindCol(m_hstmt, ++i,
SQL_C_DOUBLE, &m_txn.NewOrder.OL[0].ol_i_price, 0, NULL) !=
SQL_SUCCESS
            || SQLBindCol(m_hstmt, ++i,
SQL_C_DOUBLE, &m_txn.NewOrder.OL[0].ol_amount, 0, NULL) !=
SQL_SUCCESS
        )
            ThrowError(CODBCERR::eBindCol);
#else
        // prototype to eliminate patindex in server; shift work to
client
        i = 0;
        if ( SQLBindCol(m_hstmt, ++i, SQL_C_CHAR,
&m_ol_i_name, sizeof(m_ol_i_name), NULL) != SQL_SUCCESS
            || SQLBindCol(m_hstmt, ++i,
SQL_C_SSHORT, &m_ol_stock, 0, NULL) != SQL_SUCCESS
            || SQLBindCol(m_hstmt, ++i, SQL_C_CHAR,
&m_i_data, sizeof(m_i_data), NULL) != SQL_SUCCESS
            || SQLBindCol(m_hstmt, ++i, SQL_C_CHAR,
&m_s_data, sizeof(m_s_data), NULL) != SQL_SUCCESS
            || SQLBindCol(m_hstmt, ++i,
SQL_C_DOUBLE, &m_ol_i_price, 0, NULL) != SQL_SUCCESS
            || SQLBindCol(m_hstmt, ++i,
SQL_C_DOUBLE, &m_ol_amount, 0, NULL) != SQL_SUCCESS
        )
            ThrowError(CODBCERR::eBindCol);
#endif

        // associate the column bindings for the second result set
        if ( SQLSetStmtAttrW( m_hstmt,
SQL_ATTR_APP_ROW_DESC, m_descNewOrderCols2,
SQL_IS_POINTER) != SQL_SUCCESS )
            ThrowError(CODBCERR::eSetStmtAttr);

```

```

        i = 0;
        if ( SQLBindCol(m_hstmt, ++i, SQL_C_DOUBLE,
&m_txn.NewOrder.w_tax, 0, NULL) != SQL_SUCCESS
            || SQLBindCol(m_hstmt, ++i,
SQL_C_DOUBLE, &m_txn.NewOrder.d_tax, 0, NULL) !=
SQL_SUCCESS
            || SQLBindCol(m_hstmt, ++i, SQL_C_SLONG,
&m_txn.NewOrder.o_id, 0, NULL) != SQL_SUCCESS
            || SQLBindCol(m_hstmt, ++i, SQL_C_CHAR,
&m_txn.NewOrder.c_last, sizeof(m_txn.NewOrder.c_last), NULL) !=
SQL_SUCCESS
            || SQLBindCol(m_hstmt, ++i,
SQL_C_DOUBLE, &m_txn.NewOrder.c_discount, 0, NULL) !=
SQL_SUCCESS
            || SQLBindCol(m_hstmt, ++i, SQL_C_CHAR,
&m_txn.NewOrder.c_credit, sizeof(m_txn.NewOrder.c_credit),
NULL) != SQL_SUCCESS
            || SQLBindCol(m_hstmt, ++i,
SQL_C_TYPE_TIMESTAMP, &m_txn.NewOrder.o_entry_d, 0,
NULL) != SQL_SUCCESS
            || SQLBindCol(m_hstmt, ++i, SQL_C_SLONG,
&m_no_commit_flag, 0, NULL) != SQL_SUCCESS
        )
            ThrowError(CODBCERR::eBindCol);
    }

void CTPCC_ODBC::NewOrder(BOOL fLocalFlag)
{
    int i;
    RETCODE rc;
    int iTryCount = 0;

    // 0 1 2
    //
    012345678901234567890123456789
    wchar_t
    szSqlTemplate[] = L" { call tpcc_neworder(?,?,?,?,?,"
        L"?,?,?,?,?,?,?,?,?,?,?,?,?,"
        L"?,?,?,?,?,?,?,?,?,?,?,?,?,"
        L"?,?,?,?,?,?,?,?,?,?,?,?,?)} ";

    m_hstmt = m_hstmtNewOrder;

    // associate the parameter and column bindings for this
transaction
    if ( SQLSetStmtAttrW( m_hstmt,
SQL_ATTR_APP_ROW_DESC, m_descNewOrderCols1,
SQL_IS_POINTER) != SQL_SUCCESS )
        ThrowError(CODBCERR::eSetStmtAttr);

    // clip statement buffer based on number of parameters
    // fixed part is 29 chars and variable part is 6 chars per line
item
    i = 29 + m_txn.NewOrder.o_ol_cnt*6;
    wcsncpy( &szSqlTemplate[i], L" )" );

    // check whether any order lines are for a remote warehouse
    m_txn.NewOrder.o_all_local = 1;
    for ( i = 0; i < m_txn.NewOrder.o_ol_cnt; i++)
    {
        if ( m_txn.NewOrder.OL[i].ol_supply_w_id !=
m_txn.NewOrder.w_id)
            {

```

```

        m_txn.NewOrder.o_all_local = 0; //
at least one remote warehouse
        break;
    }
}

while (TRUE)
{
    try
    {
        m_BindOffset = 0;
        rc = SQLExecDirectW(m_hstmt,
(SQLWCHAR*)szSqlTemplate,SQL_NTS);
        if (rc != SQL_SUCCESS && rc !=
SQL_SUCCESS_WITH_INFO)
        {
#ifdef ErrTrace
            fprintf(errFile,"NewOrder
Error\nw_id: %d\nd_id: %d\nc_id: %d\nError:\n\n",
m_txn.NewOrder.w_id,
m_txn.NewOrder.d_id,
m_txn.NewOrder.c_id);
            fflush(errFile);
#endif

            ThrowError(CODBCERR::eExecDirect);
        }
        // Get order line results
        m_txn.NewOrder.total_amount = 0;
        for (i = 0;
i < m_txn.NewOrder.o_ol_cnt; i++)
        {
#ifdef new_order_strstr
            // set the bind offset
            value...
            m_BindOffset = i *
sizeof(m_txn.NewOrder.OL[0]);

            if ( SQLFetch(m_hstmt)
== SQL_ERROR)
                ThrowError(CODBCERR::eFetch);
#else
            if ( SQLFetch(m_hstmt)
== SQL_ERROR)
                ThrowError(CODBCERR::eFetch);

            strcpy(
m_txn.NewOrder.OL[i].ol_i_name, m_ol_i_name );

            if ( strstr(m_i_data,
"ORIGINAL") != NULL &&strstr(m_s_data, "ORIGINAL") != NULL
)
                m_txn.NewOrder.OL[i].ol_brand_generic[0] = 'B';
            else
                m_txn.NewOrder.OL[i].ol_brand_generic[0] = 'G';
            m_txn.NewOrder.OL[i].ol_brand_generic[1] = 0;

```

```

        m_txn.NewOrder.OL[i].ol_stock
            = m_ol_stock;
        m_txn.NewOrder.OL[i].ol_i_price
            =
m_ol_i_price;
        m_txn.NewOrder.OL[i].ol_amount
            =
m_ol_amount;
#endif
        // move to the next
        resultset
            if (
SQLMoreResults(m_hstmt) == SQL_ERROR )
                ThrowError(CODBCERR::eMoreResults);

        m_txn.NewOrder.total_amount += m_txn.NewOrder.OL[i].ol_amount;
        }
        // associate the column bindings for
the second result set
        if ( SQLSetStmtAttrW( m_hstmt,
SQL_ATTR_APP_ROW_DESC, m_descNewOrderCols2,
SQL_IS_POINTER ) != SQL_SUCCESS )
            ThrowError(CODBCERR::eSetStmtAttr);

        if ( SQLFetch(m_hstmt) ==
SQL_ERROR)
            ThrowError(CODBCERR::eFetch);
        SQLFreeStmt(m_hstmt,
SQL_CLOSE);

        if (m_no_commit_flag == 1)
        {
            m_txn.NewOrder.total_amount *= ((1 + m_txn.NewOrder.w_tax +
m_txn.NewOrder.d_tax) * (1 - m_txn.NewOrder.c_discount));
            m_txn.NewOrder.exec_status_code = eOK;
        }
        else
            m_txn.NewOrder.exec_status_code = eInvalidItem;

        break;
    }
    catch (CODBCERR *e)
    {
        if ((!e->m_bDeadLock) ||
(++iTryCount > iMaxRetries))
            throw;

        // hit deadlock; backoff for
increasingly longer period
        delete e;
        Sleep(10 * iTryCount);
    }
}

// if (iTryCount)
// throw new

```

```

CTPCC_ODBC_ERR(CTPCC_ODBC_ERR::ERR_RETRIED_TRAN
S, iTryCount);
}

void CTPCC_ODBC::InitPaymentParams()
{
    if ( SQLAllocHandle(SQL_HANDLE_STMT,m_hdbc,
&m_hstmtPayment) != SQL_SUCCESS )
        ThrowError(CODBCERR::eAllocHandle);

    m_hstmt = m_hstmtPayment;

    int i = 0;
    if ( SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT,SQL_C_SLONG,SQL_INTEGER,0,0,
&m_txn.Payment.w_id,0,NULL) != SQL_SUCCESS
        || SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT,SQL_C_SLONG,SQL_INTEGER,0,0,
&m_txn.Payment.c_w_id,0,NULL) != SQL_SUCCESS
        || SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT,SQL_C_DOUBLE,SQL_NUMERIC,6,2,
&m_txn.Payment.h_amount,0,NULL) != SQL_SUCCESS
        || SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT,SQL_C_UTINYINT,SQL_TINYINT,0,0,
&m_txn.Payment.d_id,0,NULL) != SQL_SUCCESS
        || SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT,SQL_C_UTINYINT,SQL_TINYINT,0,0,
&m_txn.Payment.c_d_id,0,NULL) != SQL_SUCCESS
        || SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT,SQL_C_SLONG,SQL_INTEGER,0,0,
&m_txn.Payment.c_id,0,NULL) != SQL_SUCCESS
        || SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT,SQL_C_CHAR,SQL_CHAR,
sizeof(m_txn.Payment.c_last),0,&m_txn.Payment.c_last,
sizeof(m_txn.Payment.c_last),NULL) != SQL_SUCCESS
    )
        ThrowError(CODBCERR::eBindParam);

    i = 0;
    if ( SQLBindCol(m_hstmt, ++i,SQL_C_SLONG,
&m_txn.Payment.c_id,0,NULL) !=
SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.c_last,
sizeof(m_txn.Payment.c_last),NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,
SQL_C_TYPE_TIMESTAMP,&m_txn.Payment.h_date,
0,NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.w_street_1, sizeof(m_txn.Payment.w_street_1),
NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.w_street_2, sizeof(m_txn.Payment.w_street_2),
NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.w_city,
sizeof(m_txn.Payment.w_city),NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.w_state,
sizeof(m_txn.Payment.w_state),NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.w_zip,
sizeof(m_txn.Payment.w_zip),NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.d_street_1, sizeof(m_txn.Payment.d_street_1),
NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.d_street_2, sizeof(m_txn.Payment.d_street_2),
NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.d_city,
sizeof(m_txn.Payment.d_city),NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.d_state,
sizeof(m_txn.Payment.d_state),NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.d_zip,
sizeof(m_txn.Payment.d_zip),NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.c_first,
sizeof(m_txn.Payment.c_first),NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.c_middle, sizeof(m_txn.Payment.c_middle),
NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.c_street_1, sizeof(m_txn.Payment.c_street_1),
NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.c_street_2, sizeof(m_txn.Payment.c_street_2),
NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.c_city,
sizeof(m_txn.Payment.c_city),NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.c_state,
sizeof(m_txn.Payment.c_state),NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.c_zip,
sizeof(m_txn.Payment.c_zip),NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.c_phone,
sizeof(m_txn.Payment.c_phone),NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,
SQL_C_TYPE_TIMESTAMP,&m_txn.Payment.c_since,
0,NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.c_credit, sizeof(m_txn.Payment.c_credit),
NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,
SQL_C_DOUBLE, &m_txn.Payment.c_credit_lim,0,NULL) !=
SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,
SQL_C_DOUBLE, &m_txn.Payment.c_discount, 0,NULL) !=
SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,
SQL_C_DOUBLE, &m_txn.Payment.c_balance, 0,NULL) !=
SQL_SUCCESS
        || SQLBindCol(m_hstmt, ++i,SQL_C_CHAR,
&m_txn.Payment.c_data,
sizeof(m_txn.Payment.c_data),NULL) != SQL_SUCCESS
    )
        ThrowError(CODBCERR::eBindCol);
}

void CTPCC_ODBC::Payment(BOOL fLocalFlag)
{
    RETCODE rc;
    int iTryCount = 0;

    m_hstmt = m_hstmtPayment;

    if (m_txn.Payment.c_id != 0)
        m_txn.Payment.c_last[0] = 0;
}

```

```

        while (TRUE)
        {
            try
            {
                rc = SQLExecDirectW(m_hstmt,
(SQLWCHAR*)"L"{call tpcc_payment(?,?,?,?,?)", SQL_NTS);
                if (rc != SQL_SUCCESS && rc !=
SQL_SUCCESS_WITH_INFO)
                {
#ifdef ErrTrace
                    fprintf(errFile,"Payment
Error\n w_id: %d\nc_w_id: %d\nc_id: %d\nc_last: %s\nd_id:
%d\nc_d_id: %d\nError:\n",
m_txn.Payment.w_id,
m_txn.Payment.c_w_id,
m_txn.Payment.c_id,
m_txn.Payment.c_last,
m_txn.Payment.d_id,
m_txn.Payment.c_d_id);
                    fflush(errFile);
#endif

                ThrowError(CODBCERR::eExecDirect);
            }
            if ( SQLFetch(m_hstmt) ==
SQL_ERROR)
                ThrowError(CODBCERR::eFetch);

                SQLFreeStmt(m_hstmt,
SQL_CLOSE);

                if (m_txn.Payment.c_id == 0)
                    throw new
CTPCC_ODBC_ERR( CTPCC_ODBC_ERR::ERR_INVALID_CUST
);
                else
                    m_txn.Payment.exec_status_code = eOK;

                    break;
                }
            catch (CODBCERR *e)
            {
                if (!(e->m_bDeadLock) ||
(++iTryCount > iMaxRetries))
                    throw;

                    // hit deadlock; backoff for
increasingly longer period

                    delete e;
                    Sleep(10 * iTryCount);
                }
            }

            // if (iTryCount)
            // throw new

CTPCC_ODBC_ERR(CTPCC_ODBC_ERR::ERR_RETRIED_TRAN
S, iTryCount);

```

```

    }

void CTPCC_ODBC::InitOrderStatusParams()
{
    if ( SQLAllocHandle(SQL_HANDLE_STMT,m_hdbc,
&m_hstmtOrderStatus) != SQL_SUCCESS
        || SQLAllocHandle(SQL_HANDLE_DESC,
m_hdbc, &m_descOrderStatusCols1) != SQL_SUCCESS
        || SQLAllocHandle(SQL_HANDLE_DESC,
m_hdbc, &m_descOrderStatusCols2) != SQL_SUCCESS
    )
        ThrowError(CODBCERR::eAllocHandle);

        m_hstmt = m_hstmtOrderStatus;

        if ( SQLSetStmtAttrW(m_hstmt,
SQL_ATTR_APP_ROW_DESC, m_descOrderStatusCols1,
SQL_IS_POINTER) != SQL_SUCCESS )
            ThrowError(CODBCERR::eSetStmtAttr);

            int i = 0;
            if ( SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT,SQL_C_SLONG,SQL_INTEGER,0,0,
&m_txn.OrderStatus.w_id, 0, NULL) != SQL_SUCCESS
                || SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT,SQL_C_UTINYINT,SQL_TINYINT,0,0,
&m_txn.OrderStatus.d_id, 0, NULL) != SQL_SUCCESS
                || SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT,SQL_C_SLONG,SQL_INTEGER,0,0,
&m_txn.OrderStatus.c_id, 0, NULL) != SQL_SUCCESS
                || SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT,SQL_C_CHAR,SQL_CHAR,
sizeof(m_txn.OrderStatus.c_last), 0, &m_txn.OrderStatus.c_last,
sizeof(m_txn.OrderStatus.c_last), NULL) != SQL_SUCCESS
            )
                ThrowError(CODBCERR::eBindParam);

                // configure block cursor
                if ( SQLSetStmtAttrW(m_hstmt,
SQL_ATTR_ROW_BIND_TYPE,
(SQLPOINTER)sizeof(m_txn.OrderStatus.OL[0]),0) !=
SQL_SUCCESS
                    || SQLSetStmtAttrW(m_hstmt,
SQL_ATTR_ROWS_FETCHED_PTR, &m_RowsFetched, 0) !=
SQL_SUCCESS
                )
                    ThrowError(CODBCERR::eSetStmtAttr);

                    i = 0;
                    if ( SQLBindCol(m_hstmt, ++i, SQL_C_SLONG,
&m_txn.OrderStatus.OL[0].ol_supply_w_id, 0, NULL) !=
SQL_SUCCESS
                        || SQLBindCol(m_hstmt, ++i, SQL_C_SLONG,
&m_txn.OrderStatus.OL[0].ol_i_id, 0, NULL) != SQL_SUCCESS
                        || SQLBindCol(m_hstmt, ++i,
SQL_C_SSHORT, &m_txn.OrderStatus.OL[0].ol_quantity, 0, NULL)
!= SQL_SUCCESS
                        || SQLBindCol(m_hstmt, ++i,
SQL_C_DOUBLE, &m_txn.OrderStatus.OL[0].ol_amount, 0, NULL)
!= SQL_SUCCESS
                    )
                        || SQLBindCol(m_hstmt, ++i,
SQL_C_TYPE_TIMESTAMP,
&m_txn.OrderStatus.OL[0].ol_delivery_d, 0, NULL) !=
SQL_SUCCESS
                )
                    ThrowError(CODBCERR::eBindCol);

```

```

        if ( SQLSetStmtAttrW(m_hstmt,
SQL_ATTR_APP_ROW_DESC, m_descOrderStatusCols2,
SQL_IS_POINTER) != SQL_SUCCESS )
            ThrowError(CODBCERR::eSetStmtAttr);

        i = 0;
        if ( SQLBindCol(m_hstmt, ++i, SQL_C_SLONG,
&m_txn.OrderStatus.c_id, 0, NULL) != SQL_SUCCESS
            || SQLBindCol(m_hstmt, ++i, SQL_C_CHAR,
&m_txn.OrderStatus.c_last, sizeof(m_txn.OrderStatus.c_last), NULL)
!= SQL_SUCCESS
            || SQLBindCol(m_hstmt, ++i, SQL_C_CHAR,
&m_txn.OrderStatus.c_first, sizeof(m_txn.OrderStatus.c_first), NULL)
!= SQL_SUCCESS
            || SQLBindCol(m_hstmt, ++i, SQL_C_CHAR,
&m_txn.OrderStatus.c_middle, sizeof(m_txn.OrderStatus.c_middle),
NULL) != SQL_SUCCESS
            || SQLBindCol(m_hstmt, ++i,
SQL_C_TYPE_TIMESTAMP, &m_txn.OrderStatus.o_entry_d, 0,
NULL) != SQL_SUCCESS
            || SQLBindCol(m_hstmt, ++i,
SQL_C_SSHORT, &m_txn.OrderStatus.o_carrier_id, 0, NULL) !=
SQL_SUCCESS
            || SQLBindCol(m_hstmt, ++i,
SQL_C_DOUBLE, &m_txn.OrderStatus.c_balance, 0, NULL) !=
SQL_SUCCESS
            || SQLBindCol(m_hstmt, ++i, SQL_C_SLONG,
&m_txn.OrderStatus.o_id, 0, NULL) != SQL_SUCCESS
        )
            ThrowError(CODBCERR::eBindCol);
    }

void CTPCC_ODBC::OrderStatus()
{
    int
iTryCount = 0;
    RETCODE
rc;

    m_hstmt = m_hstmtOrderStatus;

    if ( SQLSetStmtAttrW(m_hstmt,
SQL_ATTR_APP_ROW_DESC, m_descOrderStatusCols1,
SQL_IS_POINTER) != SQL_SUCCESS )
        ThrowError(CODBCERR::eSetStmtAttr);

    if (m_txn.OrderStatus.c_id != 0)
        m_txn.OrderStatus.c_last[0] = 0;

    while (TRUE)
    {
        try
        {
            // configure block cursor
            if ( SQLSetStmtAttrW(m_hstmt,
SQL_ATTR_ROW_ARRAY_SIZE, (SQLPOINTER)1, 0) !=
SQL_SUCCESS )

                ThrowError(CODBCERR::eSetStmtAttr);

            rc = SQLExecDirectW(m_hstmt,
(SQLWCHAR*)" {call tpcc_orderstatus(?,?,?)}", SQL_NTS);
            if ( ((rc ==
SQL_SUCCESS_WITH_INFO) && (m_RowsFetched != 0)) || (rc ==
SQL_ERROR) )
                {
#ifdef ErrTrace

```

```

                fprintf(errFile, "Order
Status\n w_id: %d\nc_id: %d\nc_last: %s\nd_id: %d\nError:\n",
m_txn.OrderStatus.w_id,
m_txn.OrderStatus.c_id,
m_txn.OrderStatus.c_last,
m_txn.OrderStatus.d_id);

                fflush(errFile);
            #endif

            ThrowError(CODBCERR::eExecDirect);
        }

        // configure block cursor
        if ( SQLSetStmtAttrW(m_hstmt,
SQL_ATTR_ROW_ARRAY_SIZE,
(SQLPOINTER)MAX_OL_ORDER_STATUS_ITEMS, 0) !=
SQL_SUCCESS )

            ThrowError(CODBCERR::eSetStmtAttr);

            rc = SQLFetchScroll( m_hstmt,
SQL_FETCH_NEXT, 0 );
            if ( ((rc ==
SQL_SUCCESS_WITH_INFO) && (m_RowsFetched != 0)) || (rc ==
SQL_ERROR) )

                ThrowError(CODBCERR::eFetchScroll);

                m_txn.OrderStatus.o_ol_cnt =
(short)m_RowsFetched;

                if (m_txn.OrderStatus.o_ol_cnt != 0)
                {
                    if ( SQLSetStmtAttrW(
m_hstmt, SQL_ATTR_APP_ROW_DESC, m_descOrderStatusCols2,
SQL_IS_POINTER) != SQL_SUCCESS )

                        ThrowError(CODBCERR::eSetStmtAttr);

                    if (
SQLMoreResults(m_hstmt) == SQL_ERROR )

                        ThrowError(CODBCERR::eMoreResults);

                    if ( (rc =
SQLFetch(m_hstmt)) == SQL_ERROR )

                        ThrowError(CODBCERR::eFetch);
                }

                SQLFreeStmt(m_hstmt,
SQL_CLOSE);

                if (m_txn.OrderStatus.o_ol_cnt == 0)
                    throw new
CTPCC_ODBC_ERR(
CTPCC_ODBC_ERR::ERR_NO_SUCH_ORDER );
                else if (m_txn.OrderStatus.c_id == 0
&& m_txn.OrderStatus.c_last[0] == 0)
                    throw new
CTPCC_ODBC_ERR( CTPCC_ODBC_ERR::ERR_INVALID_CUST
);

```



```

                else
m_txn.OrderStatus.exec_status_code = eOK;

                break;
            }
            catch (CODBCERR *e)
            {
                if (!(e->m_bDeadLock) ||
(++iTryCount > iMaxRetries))
                    throw;

                // hit deadlock; backoff for
                // increasingly longer period
                delete e;
                Sleep(10 * iTryCount);
            }
        }

//        if (iTryCount)
//            throw new

CTPCC_ODBC_ERR(CTPCC_ODBC_ERR::ERR_RETRIED_TRAN
S, iTryCount);
}

void CTPCC_ODBC::InitDeliveryParams()
{
    if ( SQLAllocHandle(SQL_HANDLE_STMT,m_hdbc,
&m_hstmtDelivery) != SQL_SUCCESS )
        ThrowError(CODBCERR::eAllocHandle);

    m_hstmt = m_hstmtDelivery;

    int i = 0;
    if ( SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT,SQL_C_SLONG,SQL_INTEGER,0,0,
&m_txn.Delivery.w_id,0, NULL) != SQL_SUCCESS
        || SQLBindParameter(m_hstmt, ++i,
SQL_PARAM_INPUT,SQL_C_SSHORT,SQL_SMALLINT,0,0,
&m_txn.Delivery.o_carrier_id,0, NULL) != SQL_SUCCESS
        )
        ThrowError(CODBCERR::eBindParam);

    for (i=0;i<10;i++)
    {
        if ( SQLBindCol(m_hstmt, (UWORD)(i+1),
SQL_C_SLONG, &m_txn.Delivery.o_id[i], 0, NULL) !=
SQL_SUCCESS )

        ThrowError(CODBCERR::eBindCol);
    }
}

void CTPCC_ODBC::Delivery()
{
    RETCODE                rc;
    int                    iTryCount = 0;

    m_hstmt = m_hstmtDelivery;

    while (TRUE)
    {
        try
        {
            rc = SQLExecDirectW(m_hstmt,
(SQLWCHAR*)L"{call tpcc_delivery(?,?)", SQL_NTS);

```

```

                if (rc != SQL_SUCCESS && rc !=
SQL_SUCCESS_WITH_INFO)
                    ThrowError(CODBCERR::eExecDirect);

                if ( SQLFetch(m_hstmt) ==
SQL_ERROR )
                    ThrowError(CODBCERR::eFetch);

                SQLFreeStmt(m_hstmt,
SQL_CLOSE);
                m_txn.Delivery.exec_status_code =
eOK;
                break;
            }
            catch (CODBCERR *e)
            {
                if (!(e->m_bDeadLock) ||
(++iTryCount > iMaxRetries))
                    throw;

                // hit deadlock; backoff for
                // increasingly longer period
                delete e;
                Sleep(10 * iTryCount);
            }
        }

//        if (iTryCount)
//            throw new

CTPCC_ODBC_ERR(CTPCC_ODBC_ERR::ERR_RETRIED_TRAN
S, iTryCount);
}

tpcc_odbc.h

/*        FILE:                TPCC_ODBC.H
*                                Microsoft TPC-C Kit Ver.
4.20.000
*                                Copyright Microsoft,
1999
*                                All Rights Reserved
*                                Version 4.10.000 audited
by Richard Gimarc, Performance Metrics, 3/17/99
*
*        PURPOSE:                Header file for TPC-C txn class
implementation.
*
*        Change history:
*                                4.20.000 - updated rev number to match kit
*/
#pragma once

// need to declare functions for import, unless define has already been
created
// by the DLL's .cpp module for export.
#ifdef DllDecl
#define DllDecl __declspec( dllimport )
#endif

class CODBCERR : public CBaseErr
{
public:

```

```

enum ACTION
{
    eNone,
    eUnknown,
    eAllocConn,
// error from SQLAllocConnect
    eAllocHandle,
error from SQLAllocHandle
    eConnOption,
error from SQLSetConnectOption
    eConnect,
error from SQLConnect
    eAllocStmt,
// error from SQLAllocStmt
    eExecDirect,
error from SQLExecDirect
    eBindParam,
// error from SQLBindParameter
    eBindCol,
error from SQLBindCol
    eFetch,
// error from SQLFetch
    eFetchScroll,
error from SQLFetchScroll
    eMoreResults,
error from SQLMoreResults
    ePrepare,
error from SQLPrepare
    eExecute,
error from SQLExecute
    eSetEnvAttr,
error from SQLSetEnvAttr
    eSetStmtAttr,
error from SQLSetStmtAttr
};

CODBCERR(void)
{
    m_eAction = eNone;
    m_NativeError = 0;
    m_bDeadLock = FALSE;
    m_odbcerrstr = NULL;
};

~CODBCERR()
{
    if (m_odbcerrstr != NULL)
        delete [] m_odbcerrstr;
};

ACTION m_eAction;
int m_NativeError;
BOOL m_bDeadLock;
char *m_odbcerrstr;

int ErrorType() {return ERR_TYPE_ODBC;};
int ErrorNum() {return m_NativeError;};
char *ErrorText() {return m_odbcerrstr;};
};

class CTPCC_ODBC_ERR : public CBaseErr
{
public:
    enum TPCC_ODBC_ERRS
    {
        ERR_WRONG_SP_VERSION = 1,
// "Wrong version of stored procs on database server"

        ERR_INVALID_CUST,
// "Invalid Customer id,name."
        ERR_NO_SUCH_ORDER,
// "No orders found for customer."
        ERR_RETRIED_TRANS,
// "Retries before transaction succeeded."
    };

    CTPCC_ODBC_ERR( int iErr ) { m_erno =
iErr; m_iTryCount = 0; };

    CTPCC_ODBC_ERR( int iErr, int iTryCount ) {
m_erno = iErr; m_iTryCount = iTryCount; };

    int m_erno;
    int m_iTryCount;

    int ErrorType() {return
ERR_TYPE_TPCC_ODBC;};
    int ErrorNum() {return m_erno;};

    char *ErrorText();
};

class DllDecl CTPCC_ODBC : public CTPCC_BASE
{
private:
// declare variables and private functions here...
    BOOL m_bDeadlock;
// transaction was selected as deadlock victim
    int m_MaxRetries;
// retry count on deadlock

    SQLHENV m_henv;
// ODBC environment handle
    SQLHDBC m_hdbc;
    SQLHSTMT m_hstmt;
// the current hstmt

    SQLHSTMT m_hstmtNewOrder;
    SQLHSTMT m_hstmtPayment;
    SQLHSTMT m_hstmtDelivery;
    SQLHSTMT m_hstmtOrderStatus;
    SQLHSTMT m_hstmtStockLevel;

    SQLHDESC m_descNewOrderCols1;
    SQLHDESC m_descNewOrderCols2;
    SQLHDESC m_descOrderStatusCols1;
    SQLHDESC m_descOrderStatusCols2;

// new-order specific fields
    SQLINTEGER m_BindOffset;
    SQLINTEGER m_RowsFetched;
    int
m_no_commit_flag;
#ifdef new_order_strstr
// for new-order txn;
// output params
    char
m_ol_i_name[I_NAME_LEN+1];
    double m_ol_i_price;
    double m_ol_amount;
    short m_ol_stock;
// used locally, but not returned to caller
    char m_i_data[I_DATA_LEN];
    char
m_s_data[S_DATA_LEN];
};

```

```

#endif
void ThrowError( CODBCERR::ACTION
eAction );

void InitNewOrderParams();
void InitPaymentParams();
void InitDeliveryParams();
void InitStockLevelParams();
void InitOrderStatusParams();

union
{
    NEW_ORDER_DATA
NewOrder;
    PAYMENT_DATA
Payment;
    DELIVERY_DATA
Delivery;
    STOCK_LEVEL_DATA
StockLevel;
    ORDER_STATUS_DATA
OrderStatus;
} m_txn;

public:
    CTPCC_ODBC(LPCSTR szServer, LPCSTR
szUser, LPCSTR szPassword, LPCSTR szHost, LPCSTR szDatabase);
    ~CTPCC_ODBC(void);

    inline PNEW_ORDER_DATA
BuffAddr_NewOrder() { return
&m_txn.NewOrder; };
    inline PPAYMENT_DATA
BuffAddr_Payment() { return &m_txn.Payment; };
    inline PDELIVERY_DATA
BuffAddr_Delivery() { return &m_txn.Delivery; };
    inline PSTOCK_LEVEL_DATA
BuffAddr_StockLevel() { return &m_txn.StockLevel; };
    inline PORORDER_STATUS_DATA
BuffAddr_OrderStatus() { return &m_txn.OrderStatus; };

    void NewOrder (BOOL
fLocalFlag);
    void Payment (BOOL
fLocalFlag);
    void Delivery ();
    void StockLevel ();
    void OrderStatus ();

};

// wrapper routine for class constructor
extern "C" DllDecl CTPCC_ODBC* CTPCC_ODBC_new
(LPCSTR szServer, LPCSTR szUser, LPCSTR
szPassword, LPCSTR szHost, LPCSTR szDatabase );

typedef CTPCC_ODBC* (TYPE_CTPCC_ODBC)(LPCSTR,
LPCSTR, LPCSTR, LPCSTR, LPCSTR);

trans.h

/* FILE: TRANS.H
* Microsoft TPC-C Kit Ver.
4.30.002

```

```

* Copyright Microsoft,
1999
* All Rights Reserved
*
* Version 4.10.000 audited
by Richard Gimarc, Performance Metrics, 3/17/99
*
* PURPOSE: Header file for TPC-C structure
templates.
*
* Change history:
* 4.30.002 - changed w_id from short to long to
support warehouses above 32,767
* 4.20.000 - updated rev number to match kit
*/
#pragma once

// String length constants
#define SERVER_NAME_LEN 20
#define DATABASE_NAME_LEN 20
#define USER_NAME_LEN 20
#define PASSWORD_LEN 20
#define TABLE_NAME_LEN 20
#define I_DATA_LEN 50
#define I_NAME_LEN 24
#define BRAND_LEN 1
#define LAST_NAME_LEN 16
#define W_NAME_LEN 10
#define ADDRESS_LEN 20
#define STATE_LEN 2
#define ZIP_LEN 9
#define S_DIST_LEN 24
#define S_DATA_LEN 50
#define D_NAME_LEN 10
#define FIRST_NAME_LEN 16
#define MIDDLE_NAME_LEN 2
#define PHONE_LEN 16
#define DATETIME_LEN 30
#define CREDIT_LEN 2
#define C_DATA_LEN 250
#define H_DATA_LEN 24
#define DIST_INFO_LEN 24
#define MAX_OL_NEW_ORDER_ITEMS 15
#define MAX_OL_ORDER_STATUS_ITEMS 15
#define STATUS_LEN 25
#define OL_DIST_INFO_LEN 24

// TIMESTAMP_STRUCT is provided by the ODBC header file
sqltypes.h, but is not available
// when compiling with dblib, so redefined here. Note: we are using
the symbol "__SQLTYPES"
// (declared in sqltypes.h) as a way to determine if
TIMESTAMP_STRUCT has been declared.
#ifndef __SQLTYPES
typedef struct
{
    short /*
SQLSMALLINT*/ year;
    unsigned short /*
SQLUSMALLINT*/ month;
    unsigned short /*
SQLUSMALLINT*/ day;
    unsigned short /*
SQLUSMALLINT*/ hour;
    unsigned short /*
SQLUSMALLINT*/ minute;
    unsigned short /*
SQLUSMALLINT*/ second;

```

```

                unsigned long      /*
SQLINTEGER */ fraction;
    } TIMESTAMP_STRUCT;
#endif

// possible values for exec_status_code after transaction completes
enum EXEC_STATUS
{
    eOK,                // 0      "Transaction
committed."
    eInvalidItem,      // 1      "Item number is not
valid."
    eDeliveryFailed    // 2      "Delivery Post Failed."
};

// transaction structures
typedef struct
{
    // input params
    long
ol_supply_w_id;
    long                ol_i_id;
    short               ol_quantity;

    // output params
    char
ol_i_name[I_NAME_LEN+1];
    char
ol_brand_generic[BRAND_LEN+1];
    double              ol_i_price;
    double              ol_amount;
    short               ol_stock;
} OL_NEW_ORDER_DATA;

typedef struct
{
    // input params
    long                w_id;
    short               d_id;
    long                c_id;
    short               o_ol_cnt;

    // output params
    EXEC_STATUS
exec_status_code;
    char                c_last[LAST_NAME_LEN+1];
    char                c_credit[CREDIT_LEN+1];
    double              c_discount;
    double              w_tax;
    double              d_tax;
    long                o_id;
    short

o_commit_flag;
    TIMESTAMP_STRUCT   o_entry_d;
    short              o_all_local;
    double              total_amount;
    OL_NEW_ORDER_DATA
OL[MAX_OL_NEW_ORDER_ITEMS];
} NEW_ORDER_DATA, *PNEW_ORDER_DATA;

typedef struct
{
    // input params
    long                w_id;
    short               d_id;
    long                c_id;
    short               c_d_id;
    long                c_w_id;
    double              h_amount;

```

```

    char                c_last[LAST_NAME_LEN+1];

    // output params
    EXEC_STATUS
exec_status_code;
    TIMESTAMP_STRUCT   h_date;
    char
w_street_1[ADDRESS_LEN+1];
    char
w_street_2[ADDRESS_LEN+1];
    char
w_city[ADDRESS_LEN+1];
    char
w_state[STATE_LEN+1];
    char                w_zip[ZIP_LEN+1];
    char
d_street_1[ADDRESS_LEN+1];
    char
d_street_2[ADDRESS_LEN+1];
    char
d_city[ADDRESS_LEN+1];
    char                d_state[STATE_LEN+1];
    char                d_zip[ZIP_LEN+1];
    char
c_first[FIRST_NAME_LEN+1];
    char
c_middle[MIDDLE_NAME_LEN + 1];
    char
c_street_1[ADDRESS_LEN+1];
    char
c_street_2[ADDRESS_LEN+1];
    char
c_city[ADDRESS_LEN+1];
    char                c_state[STATE_LEN+1];
    char                c_zip[ZIP_LEN+1];
    char
c_phone[PHONE_LEN+1];
    TIMESTAMP_STRUCT   c_since;
    char
c_credit[CREDIT_LEN+1];
    double              c_credit_lim;
    double              c_discount;
    double              c_balance;
    char                c_data[200+1];
} PAYMENT_DATA, *PPAYMENT_DATA;

typedef struct
{
    long                ol_i_id;
    long                ol_supply_w_id;
    short               ol_quantity;
    double              ol_amount;
    TIMESTAMP_STRUCT   ol_delivery_d;
} OL_ORDER_STATUS_DATA;

typedef struct
{
    // input params
    long                w_id;
    short               d_id;
    long                c_id;
    char                c_last[LAST_NAME_LEN+1];

    // output params
    EXEC_STATUS
exec_status_code;
    char                c_first[FIRST_NAME_LEN+1];
    char                c_middle[MIDDLE_NAME_LEN+1];
    double              c_balance;

```

```

        long        o_id;
        TIMESTAMP_STRUCT    o_entry_d;
        short       o_carrier_id;
        OL_ORDER_STATUS_DATA
OL[MAX_OL_ORDER_STATUS_ITEMS];
        short       o_ol_cnt;
    } ORDER_STATUS_DATA, *PORDER_STATUS_DATA;

typedef struct
{
    // input params
    long        w_id;
    short       o_carrier_id;

    // output params
    EXEC_STATUS
exec_status_code;
    SYSTEMTIME    queue_time;
    long        o_id[10];
// id's of delivered orders for districts 1 to 10
} DELIVERY_DATA, *PDELIVERY_DATA;

//This structure is used for posting delivery transactions and for writing
them to the delivery server.
typedef struct _DELIVERY_TRANSACTION
{
    SYSTEMTIME    queue;
//time delivery transaction queued
    long        w_id;
//delivery warehouse
    short       o_carrier_id;    //carrier id
} DELIVERY_TRANSACTION;

typedef struct
{
    // input params
    long        w_id;
    short       d_id;
    short       threshold;

    // output params
    EXEC_STATUS
exec_status_code;
    long        low_stock;
} STOCK_LEVEL_DATA, *PSTOCK_LEVEL_DATA;

```

txn_base.h

```

/*      FILE:          TXN_BASE.H
*
*      Microsoft TPC-C Kit Ver.
4.20.000
*
*      Copyright Microsoft,
1999
*      All Rights Reserved
*
*      Version 4.10.000 audited
by Richard Gimarc, Performance Metrics, 3/17/99
*
*      PURPOSE:      Header file for TPC-C txn class
implementation.
*
*      Change history:
*      4.20.000 - updated rev number to match kit
*/

#pragma once

```

```

// need to declare functions for import, unless define has already been
created
// by the DLL's .cpp module for export.
#ifdef DllDecl
#define DllDecl __declspec( dllimport )
#endif

class DllDecl CTPCC_BASE
{
public:
    CTPCC_BASE(void) {} ;
    virtual ~CTPCC_BASE(void) {} ;

    virtual PNEW_ORDER_DATA
BuffAddr_NewOrder()    = 0;
    virtual PPAYMENT_DATA
BuffAddr_Payment()    = 0;
    virtual PDELIVERY_DATA
BuffAddr_Delivery()    = 0;
    virtual PSTOCK_LEVEL_DATA
BuffAddr_StockLevel()    = 0;
    virtual PORDER_STATUS_DATA
BuffAddr_OrderStatus()    = 0;
//    virtual void NewOrder    () =
0;
    virtual void NewOrder
(BOOL fLocal) = 0;
//    virtual void Payment    () = 0;
    virtual void Payment    (BOOL fLocal)
= 0;
    virtual void Delivery    () = 0;
    virtual void StockLevel    () =
0;
    virtual void OrderStatus    () = 0;
};

```

txnlog.h

```

/*      FILE:          TXNLOG.H
*
*      Microsoft TPC-C Kit Ver.
4.10.000
*
*      not yet audited
*
*      PURPOSE:      Header file for txn log class
*
*      Copyright Microsoft,
1999
*      All Rights Reserved
*
*/

#pragma once

typedef struct _TXN_NEWORDER
{
    BYTE        OL_Count;    //range 0 to 31
    BYTE        OL_Remote_Count; //range 0 to 31
    WORD        c_id;
    int         o_id;
} TXN_NEWORDER;

typedef struct _TXN_PAYMENT
{
    BYTE        CustByName;
    BYTE        IsRemote;
} TXN_PAYMENT;

```



```

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.
//
//
//
//
// timestamp of record TS;
int
iPos; // byte position in file
//
//
RecMap[RecMapSize];
// #define RecMapSize 200

} TXN_LOG_HEADER, *PTXN_LOG_HEADER;

#define READ_BUFFER_SIZE 64*1024
#define WRITE_BUFFER_SIZE 8*1024

#define NUM_READ_BUFFERS 1
#define NUM_WRITE_BUFFERS 2
#define MAX_NUM_BUFFERS 2

// flags passed in to the constructor
#define TXN_LOG_WRITE 0x01
#define TXN_LOG_READ 0x02
#define TXN_LOG_SORTED 0x04
#define TXN_LOG_CRASHOPEN 0x08 // if
set, invalid headers will be tolerated; used for recovery

#define TXN_LOG_OS_ERROR 1
#define TXN_LOG_NOT_SORTED 2

#define SKIP_CTRL_RECS 1

class CTxnLog
{
private:
DWORD iBufferSize;
//buffer allocated size
DWORD iBytesFreeInBuffer;
//total bytes available for use in buffer
int iNumBuffers;
//buffers in use
int iActiveBuffer;
//indicates which buffer is active: 0 or 1
int iIoBuffer;
//buffer for any pending IO operation
int iFilePointer;
//position in file.
LARGE_INTEGER iFilePointer;
//position in file.
int iNextRec;
//when reading, ordinal value of next record

// A "save point" is remembered each time
getNextRecord is called with a start time specified.
// The next time it is called, if start time is after
the save point, we start scanning from the
// save point. This is particularly useful in
FindBestInterval, where the log is scanned repeatedly.
JULIAN_TIME SavePtTime;
int
iSavePtFilePointer;
LARGE_INTEGER iSavePtFilePointer;
int
iSavePtNextRec;

JULIAN_TIME lastTS;
//when writing sorted output, used to verify
records are sorted
BOOL bWrite;
//writing log file
BOOL bCrashOpen;
//tolerate bad headers and consistency checks

BOOL bLogSorted;
// is log file sorted? applies to both input and
output
JULIAN_TIME BeginTxnTS;
// timestamp of first (lowest) txn start
JULIAN_TIME EndTxnTS;
// timestamp of last (highest) txn completion
time
int
iRecCount; // number of records in log
file
BYTE *pCurrent;
//ptr to current buffer

```

```

        BYTE
*pBuffer[MAX_NUM_BUFFERS];

        PTXN_RECORD_HEADER *TxnArray;
//transaction record pointer array for sort

        DWORD dwError;
HANDLE hTxnFile;
//handle to log file
HANDLE hMapFile;
//map file used when sorting the log
HANDLE hIoComplete;
//event to signify that there are no pending IOs
HANDLE hLogFileIo;
//event to signal the IO thread to write the
inactive buffer

        Spinlock Spin;
//spin lock to protect the txn log file buffers

        int Write(BYTE *ptr, DWORD Size);
static void LogFileIO(CTXNLog*);

public:
        CTXNLog::CTXNLog(LPCTSTRszFileName,
DWORD dwOpts);
        ~CTXNLog(void);

        int WriteToLog(PTXN_RECORD_TPCC
pTxnRcrd);
        int
WriteToLog(PTXN_RECORD_TPCC_DELIV_DEFpTxnRcrd);
        int WriteToLog(PTXN_RECORD_CONTROL
pCtrlRec);
        int WriteToLog(PTXN_RECORD_HEADER
pCtrlRec);

        int WriteCtrlRecToLog(BYTE SubType,
LPTSTR lpStr, DWORD dwLen);

        void CloseTransactionLogFile(void);

        PTXN_RECORD_HEADER
GetNextRecord(BOOL bSkipCtrlRecs = FALSE);
        PTXN_RECORD_HEADER
GetNextRecord(JULIAN_TIMESeekTimeT0, BOOL bSkipCtrlRecs =
FALSE);

        int Sort(void);
        PTXN_RECORD_HEADER
GetSortedRecord(int index);

        inline BOOL IsSorted(void) { return
bLogSorted; };
        inline JULIAN_TIME BeginTS(void) { return
BeginTxnTS; };
        inline JULIAN_TIME EndTS(void) { return
EndTxnTS; };
        inline int RecordCount(void) { return
iRecCount; };
};

class CTXNLOG_ERR : public CBaseErr
{
public:
        enum CTXNLOG_ERRS
{

```

```

        ERR_BAD_FILE_FORMAT,
// "File format is invalid."
        ERR_UNKNOWN_LOG_VERSION,
// "Log file version is unknown."
        ERR_BROKEN_LOG_FILE,
// "Log file is broken."
        ERR_LOG_NOT_SORTED,
// "Log file is not sorted"
        ERR_INVALID_TIME_SEQ,
// "Internal Error: Record Time Sequence invalid."
};

        CTXNLOG_ERR(int iErr) : CBaseErr(iErr) {};

        int ErrorType() {return
ERR_TYPE_TXNLOG;};

        char *ErrorText()
{
        static char *szMsgs[] = {
                "File format is invalid.",
                "Log file version is
unknown.",
                "Log file is broken.",
                "Log file is not sorted",
                "Internal Error: Record
Time Sequence invalid.",
                ""
        };

        for(int i = 0; szMsgs[i][0]; i++)
        {
                if ( m_idMsg == i )
                        break;
        }

        return(szMsgs[i][0] ? szMsgs[i] :
ERR_UNKNOWN);
};
};

```

Stored Procedures

delivery.sql

```

-- File: DELIVERY.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.10
-- Copyright Microsoft, 1999
-- Purpose: Creates delivery transaction stored procedure

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONSOFF
go

use tpcc
go

if exists (select name from sysobjects where name = 'tpcc_delivery' )
drop procedure tpcc_delivery
go

create proc tpcc_delivery @w_id int,
                        @o_carrier_id smallint

```



```

as

declare @d_id tinyint,
        @o_id int,
        @c_id int,
        @total numeric(12,2),
        @oid1 int,
        @oid2 int,
        @oid3 int,
        @oid4 int,
        @oid5 int,
        @oid6 int,
        @oid7 int,
        @oid8 int,
        @oid9 int,
        @oid10 int,
        @time datetime

begin tran d

        select @d_id = 0,
               @time = getdate()

        while (@d_id < 10)
        begin
                select @d_id = @d_id + 1,
                       @total = 0,
                       @o_id = 0

                select top 1
                       @o_id = no_o_id
                from new_order (serializable uplock)
                where no_w_id = @w_id and
                       no_d_id = @d_id
                order by no_o_id asc

                if (@@rowcount <> 0)
                begin
-- claim the order for this district

                        delete new_order
                        where no_w_id = @w_id and
                               no_d_id = @d_id and
                               no_o_id = @o_id

-- set carrier_id on this order (and get customer id)

                        update orders
                        set o_carrier_id =
@o_carrier_id,
                                @c_id =
o_c_id
                                where o_w_id =
@w_id and
                                        o_d_id =
@d_id and
                                                o_id =
@o_id

-- set date in all lineitems for this order (and sum amounts)

                        update order_line
                        set ol_delivery_d =
@time,
                                @total =
@total + ol_amount

```

```

                                where ol_w_id =
@w_id and
                                        ol_d_id =
@d_id and
                                                ol_o_id =
@o_id

-- accumulate lineitem amounts for this order into customer

                                update customer
                                set c_balance = c_balance +
@total,
                                        c_delivery_cnt =
c_delivery_cnt + 1
                                where c_w_id =
@w_id and
                                        c_d_id =
@d_id and
                                                c_id =
@c_id

                                end

                                select @oid1 = case @d_id when 1 then @o_id else
@oid1 end,
                                        @oid2 = case @d_id when 2 then @o_id else @oid2 end,
                                        @oid3 = case @d_id when 3 then @o_id else @oid3 end,
                                        @oid4 = case @d_id when 4 then @o_id else @oid4 end,
                                        @oid5 = case @d_id when 5 then @o_id else @oid5 end,
                                        @oid6 = case @d_id when 6 then @o_id else @oid6 end,
                                        @oid7 = case @d_id when 7 then @o_id else @oid7 end,
                                        @oid8 = case @d_id when 8 then @o_id else @oid8 end,
                                        @oid9 = case @d_id when 9 then @o_id else @oid9 end,
                                        @oid10 = case @d_id when 10 then @o_id else @oid10 end

                                end

commit tran d

-- return delivery data to client

select @oid1,
        @oid2,
        @oid3,
        @oid4,
        @oid5,
        @oid6,
        @oid7,
        @oid8,
        @oid9,
        @oid10
go

neword.sql

-- File: NEWORD.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.30.000
-- Copyright Microsoft, 1999
-- Purpose: Creates new order transaction stored procedure
--
-- Interface Level: 4.10.000

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONS off
go

```

```

use tpcc
go

if exists ( select name from sysobjects where name = 'tpcc_neworder' )
    drop procedure tpcc_neworder
go

create proc tpcc_neworder
    int,
    tinyint,
    int,
    tinyint,
    tinyint,
    tinyint,
    0, @s_w_id1 int = 0, @ol_qty1 smallint = 0,
    0, @s_w_id2 int = 0, @ol_qty2 smallint = 0,
    0, @s_w_id3 int = 0, @ol_qty3 smallint = 0,
    0, @s_w_id4 int = 0, @ol_qty4 smallint = 0,
    0, @s_w_id5 int = 0, @ol_qty5 smallint = 0,
    0, @s_w_id6 int = 0, @ol_qty6 smallint = 0,
    0, @s_w_id7 int = 0, @ol_qty7 smallint = 0,
    0, @s_w_id8 int = 0, @ol_qty8 smallint = 0,
    0, @s_w_id9 int = 0, @ol_qty9 smallint = 0,
    0, @s_w_id10 int = 0, @ol_qty10 smallint = 0,
    0, @s_w_id11 int = 0, @ol_qty11 smallint = 0,
    0, @s_w_id12 int = 0, @ol_qty12 smallint = 0,
    0, @s_w_id13 int = 0, @ol_qty13 smallint = 0,
    0, @s_w_id14 int = 0, @ol_qty14 smallint = 0,
    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,
                 @li_no     = 0,
                 @commit_flag = 1
        where    d_w_id     = @w_id and
                 d_id      = @d_id

-- process orderlines

        while (@li_no < @o_ol_cnt)
        begin

                select @li_no = @li_no + 1

-- set i_id, s_w_id, and qty for this lineitem

                select    @li_id = case @li_no
                           when 1 then @i_id1
                           when 2 then @i_id2
                           when 3 then @i_id3
                           when 4 then @i_id4
                           when 5 then @i_id5
                           when 6 then @i_id6
                           when 7 then @i_id7
                           when 8 then @i_id8
                           when 9 then @i_id9
                           when 10 then @i_id10
                           when 11 then @i_id11
                           when 12 then @i_id12
                           when 13 then @i_id13
                           when 14 then @i_id14
                           when 15 then @i_id15
                           end,

                           @li_s_w_id = case @li_no
                                           when 1 then @s_w_id1
                                           when 2 then @s_w_id2
                                           when 3 then @s_w_id3
                                           when 4 then @s_w_id4
                                           when 5 then @s_w_id5
                                           when 6 then @s_w_id6
                                           when 7 then @s_w_id7
                                           when 8 then @s_w_id8
                                           when 9 then @s_w_id9

```

```

when 10 then
@s_w_id10
when 11 then
@s_w_id11
when 12 then
@s_w_id12
when 13 then
@s_w_id13
when 14 then
@s_w_id14
when 15 then
@s_w_id15
end,

-- get item data (no one updates item)
select
    @i_price = i_price,
    @i_name = i_name,
    @i_data = i_data
from
    item (tablock repeatableread)
where
    i_id = @li_id

-- update stock values
update
    stock
set
    s_ytd = s_ytd +
@s_li_qty,
@s_quantity = s_quantity - @li_qty +
case when (s_quantity - @li_qty < 10) then 91 else 0 end,
    s_order_cnt = s_order_cnt + 1,
    s_remote_cnt = s_remote_cnt + case when (@li_s_w_id = @w_id) then 0 else 1 end,
    @s_data = s_data,
    @s_dist = case @d_id
when 1 then
s_dist_01
when 2 then
s_dist_02
when 3 then
s_dist_03
when 4 then
s_dist_04
when 5 then
s_dist_05
when 6 then
s_dist_06
when 7 then
s_dist_07
when 8 then
s_dist_08
when 9 then
s_dist_09
when 10 then
s_dist_10
end
where
    s_i_id = @li_id and
    s_w_id = @li_s_w_id

-- if there actually is a stock (and item) with these ids, go to work
if (@@rowcount > 0)
begin
-- insert order_line data (using data from item and stock)
insert into order_line values(@o_id,
@d_id,
@w_id,
@li_no,
@li_id,
@li_s_w_id,
'dec 31, 1899',
@li_qty,
@i_price * @li_qty,
@s_dist)

-- send line-item data to client
select
    @i_name,
    @s_quantity,
    case when (
(patindex('%ORIGINAL%',@i_data) > 0) and
(patindex('%ORIGINAL%',@s_data) > 0) )
then 'B' else 'G' end,
    @i_price,
    @i_price * @li_qty
end
else
begin
-- no item (or stock) found - triggers rollback condition
select ",0",0,0
select @commit_flag = 0
end
end

-- get customer last name, discount, and credit rating
select
    @c_last = c_last,

```

```

        @c_discount = c_discount,
        @c_credit = c_credit,
        @c_id_local = c_id,
        @o_entry_d = getdate()
from customer (repeatableread)
where c_id = @c_id and
       c_w_id = @w_id and
       c_d_id = @d_id

-- insert fresh row into orders table

insert into orders values ( @o_id,
                           @d_id,
                           @w_id,
                           @c_id_local,
                           @o_entry_d,
                           0,
                           @o_ol_cnt,
                           @o_all_local)

-- insert corresponding row into new-order table

insert into new_order values ( @o_id,
                               @d_id,
                               @w_id)

-- select warehouse tax

select @w_tax = w_tax
from warehouse (repeatableread)
where w_id = @w_id

if (@commit_flag = 1)
    commit transaction n
else

-- all that work for nuthin!!!

rollback transaction n

-- return order data to client

select @w_tax,
       @d_tax,
       @o_id,
       @c_last,
       @c_discount,
       @c_credit,
       @o_entry_d,
       @commit_flag

end

go

```

ordstat.sql

```

-- File:  ORDSTAT.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.10
-- Copyright Microsoft, 1999
-- Purpose: Creates order status transaction stored procedure

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONS off
go

```

```

use tpcc
go

if exists ( select name from sysobjects where name = 'tpcc_orderstatus' )
    drop procedure tpcc_orderstatus
go

create proc tpcc_orderstatus @w_id int,
                             @d_id tinyint,
                             @c_id int,
                             @c_last char(16) = ''
as

declare @c_balance numeric(12,2),
        @c_first char(16),
        @c_middle char(2),
        @o_id int,
        @o_entry_d datetime,
        @o_carrier_id smallint,
        @cnt smallint

begin tran o

if (@c_id = 0)
    begin

-- get customer id and info using last name

select @cnt = (count(*)+1)/2
from customer (repeatableread)
where c_last = @c_last and
       c_w_id = @w_id and
       c_d_id = @d_id

set rowcount @cnt

select @c_id = c_id,
       @c_balance = c_balance,
       @c_first = c_first,
       @c_last = c_last,
       @c_middle = c_middle
from customer (repeatableread)
where c_last = @c_last and
       c_w_id = @w_id and
       c_d_id = @d_id

order by c_w_id, c_d_id, c_last, c_first

set rowcount 0

end

else

begin

-- get customer info if by id

select @c_balance = c_balance,
       @c_first = c_first,
       @c_middle = c_middle,
       @c_last = c_last
from customer (repeatableread)
where c_id = @c_id and
       c_d_id = @d_id and
       c_w_id = @w_id

```

```

                select      @cnt      = @@rowcount
            end
-- if no such customer
        if (@cnt = 0)
            begin
                raiserror('Customer not found',18,1)
                goto custnotfound
            end
-- get order info
        select      @o_id      = o_id,
                   @o_entry_d  = o_entry_d,
                   @o_carrier_id = o_carrier_id
        from        orders (serializable)
        where       o_c_id      = @c_id and
                   o_d_id      = @d_id and
                   o_w_id      = @w_id
        order      by o_id asc
-- select order lines for the current order
        select      ol_supply_w_id,
                   ol_i_id,
                   ol_quantity,
                   ol_amount,
                   ol_delivery_d
        from        order_line (repeatableread)
        where       ol_o_id = @o_id and
                   ol_d_id = @d_id and
                   ol_w_id = @w_id
custnotfound:
commit tran o
-- return data to client
select      @c_id,
           @c_last,
           @c_first,
           @c_middle,
           @o_entry_d,
           @o_carrier_id,
           @c_balance,
           @o_id
go

```

payment.sql

```

-- File:  PAYMENT.SQL
--       Microsoft TPC-C Benchmark Kit Ver. 4.10
--       Copyright Microsoft, 1999
-- Purpose:  Creates payment transaction stored procedure

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONS off
go

```

```

use tpcc
go
if exists (select name from sysobjects where name = 'tpcc_payment')
    drop procedure tpcc_payment
go
create proc tpcc_payment      @w_id      int,
                             @c_w_id    int,
                             @h_amount  numeric(6,2),
                             @d_id      tinyint,
                             @c_d_id    tinyint,
                             @c_id      int,
                             @c_last    char(16) = "
as
declare @w_street_1 char(20),
        @w_street_2 char(20),
        @w_city     char(20),
        @w_state    char(2),
        @w_zip      char(9),
        @w_name     char(10),
        @d_street_1 char(20),
        @d_street_2 char(20),
        @d_city     char(20),
        @d_state    char(2),
        @d_zip      char(9),
        @d_name     char(10),
        @c_first    char(16),
        @c_middle   char(2),
        @c_street_1 char(20),
        @c_street_2 char(20),
        @c_city     char(20),
        @c_state    char(2),
        @c_zip      char(9),
        @c_phone    char(16),
        @c_since    datetime,
        @c_credit   char(2),
        @c_credit_lim numeric(12,2),
        @c_balance  numeric(12,2),
        @c_discount numeric(4,4),
        @data       char(500),
        @c_data     char(500),
        @datetime   datetime,
        @w_ytd      numeric(12,2),
        @d_ytd      numeric(12,2),
        @cnt        smallint,
        @val        smallint,
        @screen_data char(200),
        @d_id_local tinyint,
        @w_id_local  int,
        @c_id_local  int

select @screen_data = "
begin tran p
-- get payment date
        select      @datetime = getdate()
        if (@c_id = 0)
            begin
-- get customer id and info using last name

```

```

select top 1 @c_id = c_id
from (
    select top 50 percent c_id, c_first
    from customer (repeatable read)
    where c_last = @c_last and
          c_w_id = @c_w_id
    and
          c_d_id = @c_d_id
    order by c_first
) top_fifty
order by c_first desc
end

-- get customer info and update balances

update customer
set @c_balance = c_balance -
@c_h_amount,
    c_payment_cnt = c_payment_cnt + 1,
    c_ytd_payment = c_ytd_payment +
@c_h_amount,
    @c_first = c_first,
    @c_middle = c_middle,
    @c_last = c_last,
    @c_street_1 = c_street_1,
    @c_street_2 = c_street_2,
    @c_city = c_city,
    @c_state = c_state,
    @c_zip = c_zip,
    @c_phone = c_phone,
    @c_credit = c_credit,
    @c_credit_lim = c_credit_lim,
    @c_discount = c_discount,
    @c_since = c_since,
    @data = c_data,
    @c_id_local = c_id
where c_id = @c_id and
      c_w_id = @c_w_id and
      c_d_id = @c_d_id

-- if customer has bad credit get some more info

if (@c_credit = 'BC')
begin
-- compute new info

select @c_data = convert(char(5),@c_id)
+
convert(char(4),@c_d_id) +
convert(char(5),@c_w_id) +
+
convert(char(4),@d_id)
+
convert(char(5),@w_id)
+
convert(char(19),@h_amount) +
substring(@data, 1, 458)

-- update customer info

update customer
set c_data = @c_data

```

```

where c_id = @c_id and
      c_w_id = @c_w_id and
      c_d_id = @c_d_id

select @screen_data = substring
(@c_data,1,200)
end

-- get district data and update year-to-date

update district
set d_ytd = d_ytd + @h_amount,
    @d_street_1 = d_street_1,
    @d_street_2 = d_street_2,
    @d_city = d_city,
    @d_state = d_state,
    @d_zip = d_zip,
    @d_name = d_name,
    @d_id_local = d_id
where d_w_id = @w_id and
      d_id = @d_id

-- get warehouse data and update year-to-date

update warehouse
set w_ytd = w_ytd + @h_amount,
    @w_street_1 = w_street_1,
    @w_street_2 = w_street_2,
    @w_city = w_city,
    @w_state = w_state,
    @w_zip = w_zip,
    @w_name = w_name,
    @w_id_local = w_id
where w_id = @w_id

-- create history record

insert into history values (
    @c_id_local,
    @c_d_id,
    @c_w_id,
    @d_id_local,
    @w_id_local,
    @datetime,
    @h_amount,
    @w_name +
' + @d_name)
commit tran p

-- return data to client

select @c_id,
       @c_last,
       @datetime,
       @w_street_1,
       @w_street_2,
       @w_city,
       @w_state,
       @w_zip,
       @d_street_1,
       @d_street_2,
       @d_city,
       @d_state,
       @d_zip,
       @c_first,
       @c_middle,
       @c_street_1,
       @c_street_2,

```

```
@c_city,  
@c_state,  
@c_zip,  
@c_phone,  
@c_since,  
@c_credit,  
@c_credit_lim,  
@c_discount,  
@c_balance,  
@screen_data
```

```
go
```

stocklev.sql

```
-- File: STOCKLEV.SQL  
-- Microsoft TPC-C Benchmark Kit Ver. 4.10  
-- Copyright Microsoft, 1999  
-- Purpose: Creates stock level transaction stored procedure
```

```
set ANSI_DEFAULTS on  
set IMPLICIT_TRANSACTIONS off  
go
```

```
use tpcc  
go
```

```
if exists (select name from sysobjects where name = 'tpcc_stocklevel' )  
    drop procedure tpcc_stocklevel  
go
```

```
create proc tpcc_stocklevel    @w_id    int,  
                               @d_id    tinyint,  
                               @threshold smallint  
as
```

```
declare    @o_id_low int,  
           @o_id_high int
```

```
select    @o_id_low    = (d_next_o_id - 20),  
           @o_id_high    = (d_next_o_id - 1)  
from      district  
where     d_w_id        = @w_id and  
           d_id         = @d_id
```

```
select    count(distinct(s_i_id))  
from      stock, order_line  
where     ol_w_id        = @w_id and  
           ol_d_id        = @d_id and  
           ol_o_id        between @o_id_low and  
                               @o_id_high and  
           s_w_id        = ol_w_id and  
           s_i_id        = ol_i_id and  
           s_quantity < @threshold
```

```
option (order group)
```

```
go
```

Appendix B: Database Design

Database Build

16x1800

1	rtnode01	1	1800
2	rtnode02	1801	3600
3	rtnode03	3601	5400
4	rtnode04	5401	7200
5	rtnode05	7201	9000
6	rtnode06	9001	10800
7	rtnode07	10801	12600
8	rtnode08	12601	14400
9	rtnode09	14401	16200
10	rtnode10	16201	18000
11	rtnode11	18001	19800
12	rtnode12	19801	21600
13	rtnode13	21601	23400
14	rtnode14	23401	25200
15	rtnode15	25201	27000
16	rtnode16	27001	28800

add_constraints_t1.sql

```
--file 1_to_1800\add_constraints_t.sql

set ansi_warnings on
set ansi_nulls on
go

use tpcc
go

-- Drop any existing partitioning constraints
--

alter table t_warehouse drop constraint cnst_w_id
go

alter table t_district drop constraint cnst_d_w_id
go

alter table t_customer drop constraint cnst_c_w_id
go

alter table t_history drop constraint cnst_h_w_id
go

alter table t_stock drop constraint cnst_s_w_id
go

alter table t_orders drop constraint cnst_o_w_id
go

alter table t_order_line drop constraint cnst_ol_w_id
go

alter table t_new_order drop constraint cnst_no_w_id
go

-- Add partitioning constraints between 1 and 1800
--
```

```
alter table t_warehouse add constraint cnst_w_id check (w_id <=
convert(int,1800))
go

alter table t_district add constraint cnst_d_w_id check (d_w_id <=
convert(int,1800))
go

alter table t_customer add constraint cnst_c_w_id check (c_w_id <=
convert(int,1800))
go

alter table t_history add constraint cnst_h_w_id check (h_w_id <=
convert(int,1800))
go

alter table t_stock add constraint cnst_s_w_id check (s_w_id <=
convert(int,1800))
go

alter table t_orders add constraint cnst_o_w_id check (o_w_id <=
convert(int,1800))
go

alter table t_order_line add constraint cnst_ol_w_id check (ol_w_id <=
convert(int,1800))
go

alter table t_new_order add constraint cnst_no_w_id check (no_w_id
<= convert(int,1800))
go
```

add_constraints_t2.sql

```
--file 1801_to_3600\add_constraints_t.sql

set ansi_warnings on
set ansi_nulls on
go

use tpcc
go

-- Drop any existing partitioning constraints
--

alter table t_warehouse drop constraint cnst_w_id
go

alter table t_district drop constraint cnst_d_w_id
go

alter table t_customer drop constraint cnst_c_w_id
go

alter table t_history drop constraint cnst_h_w_id
go

alter table t_stock drop constraint cnst_s_w_id
go

alter table t_orders drop constraint cnst_o_w_id
go

alter table t_order_line drop constraint cnst_ol_w_id
go
```



```

alter table t_new_order drop constraint cnst_no_w_id
go

-- Add partitioning constraints between 1801 and 3600
--

alter table t_warehouse add constraint cnst_w_id check (w_id between
convert(int,1801) and convert(int,3600))
go

alter table t_district add constraint cnst_d_w_id check (d_w_id
between convert(int,1801) and convert(int,3600))
go

alter table t_customer add constraint cnst_c_w_id check (c_w_id
between convert(int,1801) and convert(int,3600))
go

alter table t_history add constraint cnst_h_w_id check (h_w_id
between convert(int,1801) and convert(int,3600))
go

alter table t_stock add constraint cnst_s_w_id check (s_w_id between
convert(int,1801) and convert(int,3600))
go

alter table t_orders add constraint cnst_o_w_id check (o_w_id between
convert(int,1801) and convert(int,3600))
go

alter table t_order_line add constraint cnst_ol_w_id check (ol_w_id
between convert(int,1801) and convert(int,3600))
go

alter table t_new_order add constraint cnst_no_w_id check (no_w_id
between convert(int,1801) and convert(int,3600))
go

```

add_constraints_t3.sql

```

--file 3601_to_5400\add_constraints_t.sql

set ansi_warnings on
set ansi_nulls on
go

use tpcc
go

-- Drop any existing partitioning constraints
--

alter table t_warehouse drop constraint cnst_w_id
go

alter table t_district drop constraint cnst_d_w_id
go

alter table t_customer drop constraint cnst_c_w_id
go

alter table t_history drop constraint cnst_h_w_id
go

alter table t_stock drop constraint cnst_s_w_id

```

```

go

alter table t_orders drop constraint cnst_o_w_id
go

alter table t_order_line drop constraint cnst_ol_w_id
go

alter table t_new_order drop constraint cnst_no_w_id
go

-- Add partitioning constraints between 3601 and 5400
--

alter table t_warehouse add constraint cnst_w_id check (w_id between
convert(int,3601) and convert(int,5400))
go

alter table t_district add constraint cnst_d_w_id check (d_w_id
between convert(int,3601) and convert(int,5400))
go

alter table t_customer add constraint cnst_c_w_id check (c_w_id
between convert(int,3601) and convert(int,5400))
go

alter table t_history add constraint cnst_h_w_id check (h_w_id
between convert(int,3601) and convert(int,5400))
go

alter table t_stock add constraint cnst_s_w_id check (s_w_id between
convert(int,3601) and convert(int,5400))
go

alter table t_orders add constraint cnst_o_w_id check (o_w_id between
convert(int,3601) and convert(int,5400))
go

alter table t_order_line add constraint cnst_ol_w_id check (ol_w_id
between convert(int,3601) and convert(int,5400))
go

alter table t_new_order add constraint cnst_no_w_id check (no_w_id
between convert(int,3601) and convert(int,5400))
go

```

add_constraints_t4.sql

```

--file 5401_to_7200\add_constraints_t.sql

set ansi_warnings on
set ansi_nulls on
go

use tpcc
go

-- Drop any existing partitioning constraints
--

alter table t_warehouse drop constraint cnst_w_id
go

alter table t_district drop constraint cnst_d_w_id
go

```

```

alter table t_customer drop constraint cnst_c_w_id
go

alter table t_history drop constraint cnst_h_w_id
go

alter table t_stock drop constraint cnst_s_w_id
go

alter table t_orders drop constraint cnst_o_w_id
go

alter table t_order_line drop constraint cnst_ol_w_id
go

alter table t_new_order drop constraint cnst_no_w_id
go

-- Add partitioning constraints between 5401 and 7200
--

alter table t_warehouse add constraint cnst_w_id check (w_id between
convert(int,5401) and convert(int,7200))
go

alter table t_district add constraint cnst_d_w_id check (d_w_id
between convert(int,5401) and convert(int,7200))
go

alter table t_customer add constraint cnst_c_w_id check (c_w_id
between convert(int,5401) and convert(int,7200))
go

alter table t_history add constraint cnst_h_w_id check (h_w_id
between convert(int,5401) and convert(int,7200))
go

alter table t_stock add constraint cnst_s_w_id check (s_w_id between
convert(int,5401) and convert(int,7200))
go

alter table t_orders add constraint cnst_o_w_id check (o_w_id between
convert(int,5401) and convert(int,7200))
go

alter table t_order_line add constraint cnst_ol_w_id check (ol_w_id
between convert(int,5401) and convert(int,7200))
go

alter table t_new_order add constraint cnst_no_w_id check (no_w_id
between convert(int,5401) and convert(int,7200))
go

```

add_constraints_t5.sql

```

--file 7201_to_9000\add_constraints_t.sql

set ansi_warnings on
set ansi_nulls on
go

use tpcc
go

-- Drop any existing partitioning constraints
--

```

```

alter table t_warehouse drop constraint cnst_w_id
go

alter table t_district drop constraint cnst_d_w_id
go

alter table t_customer drop constraint cnst_c_w_id
go

alter table t_history drop constraint cnst_h_w_id
go

alter table t_stock drop constraint cnst_s_w_id
go

alter table t_orders drop constraint cnst_o_w_id
go

alter table t_order_line drop constraint cnst_ol_w_id
go

alter table t_new_order drop constraint cnst_no_w_id
go

-- Add partitioning constraints between 7201 and 9000
--

alter table t_warehouse add constraint cnst_w_id check (w_id between
convert(int,7201) and convert(int,9000))
go

alter table t_district add constraint cnst_d_w_id check (d_w_id
between convert(int,7201) and convert(int,9000))
go

alter table t_customer add constraint cnst_c_w_id check (c_w_id
between convert(int,7201) and convert(int,9000))
go

alter table t_history add constraint cnst_h_w_id check (h_w_id
between convert(int,7201) and convert(int,9000))
go

alter table t_stock add constraint cnst_s_w_id check (s_w_id between
convert(int,7201) and convert(int,9000))
go

alter table t_orders add constraint cnst_o_w_id check (o_w_id between
convert(int,7201) and convert(int,9000))
go

alter table t_order_line add constraint cnst_ol_w_id check (ol_w_id
between convert(int,7201) and convert(int,9000))
go

alter table t_new_order add constraint cnst_no_w_id check (no_w_id
between convert(int,7201) and convert(int,9000))
go

```

add_constraints_t6.sql

```

--file 9001_to_10800\add_constraints_t.sql

set ansi_warnings on
set ansi_nulls on

```

```

go

use tpcc
go

-- Drop any existing partitioning constraints
--

alter table t_warehouse drop constraint cnst_w_id
go

alter table t_district drop constraint cnst_d_w_id
go

alter table t_customer drop constraint cnst_c_w_id
go

alter table t_history drop constraint cnst_h_w_id
go

alter table t_stock drop constraint cnst_s_w_id
go

alter table t_orders drop constraint cnst_o_w_id
go

alter table t_order_line drop constraint cnst_ol_w_id
go

alter table t_new_order drop constraint cnst_no_w_id
go

-- Add partitioning constraints between 9001 and 10800
--

alter table t_warehouse add constraint cnst_w_id check (w_id between
convert(int,9001) and convert(int,10800))
go

alter table t_district add constraint cnst_d_w_id check (d_w_id
between convert(int,9001) and convert(int,10800))
go

alter table t_customer add constraint cnst_c_w_id check (c_w_id
between convert(int,9001) and convert(int,10800))
go

alter table t_history add constraint cnst_h_w_id check (h_w_id
between convert(int,9001) and convert(int,10800))
go

alter table t_stock add constraint cnst_s_w_id check (s_w_id between
convert(int,9001) and convert(int,10800))
go

alter table t_orders add constraint cnst_o_w_id check (o_w_id between
convert(int,9001) and convert(int,10800))
go

alter table t_order_line add constraint cnst_ol_w_id check (ol_w_id
between convert(int,9001) and convert(int,10800))
go

alter table t_new_order add constraint cnst_no_w_id check (no_w_id
between convert(int,9001) and convert(int,10800))
go

```

add_constraints_t7.sql

```

--file 10801_to_12600\add_constraints_t.sql

set ansi_warnings on
set ansi_nulls on
go

use tpcc
go

-- Drop any existing partitioning constraints
--

alter table t_warehouse drop constraint cnst_w_id
go

alter table t_district drop constraint cnst_d_w_id
go

alter table t_customer drop constraint cnst_c_w_id
go

alter table t_history drop constraint cnst_h_w_id
go

alter table t_stock drop constraint cnst_s_w_id
go

alter table t_orders drop constraint cnst_o_w_id
go

alter table t_order_line drop constraint cnst_ol_w_id
go

alter table t_new_order drop constraint cnst_no_w_id
go

-- Add partitioning constraints between 10801 and 12600
--

alter table t_warehouse add constraint cnst_w_id check (w_id between
convert(int,10801) and convert(int,12600))
go

alter table t_district add constraint cnst_d_w_id check (d_w_id
between convert(int,10801) and convert(int,12600))
go

alter table t_customer add constraint cnst_c_w_id check (c_w_id
between convert(int,10801) and convert(int,12600))
go

alter table t_history add constraint cnst_h_w_id check (h_w_id
between convert(int,10801) and convert(int,12600))
go

alter table t_stock add constraint cnst_s_w_id check (s_w_id between
convert(int,10801) and convert(int,12600))
go

alter table t_orders add constraint cnst_o_w_id check (o_w_id between
convert(int,10801) and convert(int,12600))
go

alter table t_order_line add constraint cnst_ol_w_id check (ol_w_id
between convert(int,10801) and convert(int,12600))
go

```

```

go

alter table t_new_order add constraint cnst_no_w_id check (no_w_id
between convert(int,10801) and convert(int,12600))
go

```

add_constraints_t8.sql

```

--file 12601_to_14400\add_constraints_t.sql

set ansi_warnings on
set ansi_nulls on
go

use tpcc
go

-- Drop any existing partitioning constraints
--

alter table t_warehouse drop constraint cnst_w_id
go

alter table t_district drop constraint cnst_d_w_id
go

alter table t_customer drop constraint cnst_c_w_id
go

alter table t_history drop constraint cnst_h_w_id
go

alter table t_stock drop constraint cnst_s_w_id
go

alter table t_orders drop constraint cnst_o_w_id
go

alter table t_order_line drop constraint cnst_ol_w_id
go

alter table t_new_order drop constraint cnst_no_w_id
go

-- Add partitioning constraints between 12601 and 14400
--

alter table t_warehouse add constraint cnst_w_id check (w_id between
convert(int,12601) and convert(int,14400))
go

alter table t_district add constraint cnst_d_w_id check (d_w_id
between convert(int,12601) and convert(int,14400))
go

alter table t_customer add constraint cnst_c_w_id check (c_w_id
between convert(int,12601) and convert(int,14400))
go

alter table t_history add constraint cnst_h_w_id check (h_w_id
between convert(int,12601) and convert(int,14400))
go

alter table t_stock add constraint cnst_s_w_id check (s_w_id between
convert(int,12601) and convert(int,14400))
go

```

```

alter table t_orders add constraint cnst_o_w_id check (o_w_id between
convert(int,12601) and convert(int,14400))
go

```

```

alter table t_order_line add constraint cnst_ol_w_id check (ol_w_id
between convert(int,12601) and convert(int,14400))
go

```

```

alter table t_new_order add constraint cnst_no_w_id check (no_w_id
between convert(int,12601) and convert(int,14400))
go

```

add_constraints_t9.sql

```

--file 14401_to_16200\add_constraints_t.sql

set ansi_warnings on
set ansi_nulls on
go

use tpcc
go

-- Drop any existing partitioning constraints
--

alter table t_warehouse drop constraint cnst_w_id
go

alter table t_district drop constraint cnst_d_w_id
go

alter table t_customer drop constraint cnst_c_w_id
go

alter table t_history drop constraint cnst_h_w_id
go

alter table t_stock drop constraint cnst_s_w_id
go

alter table t_orders drop constraint cnst_o_w_id
go

alter table t_order_line drop constraint cnst_ol_w_id
go

alter table t_new_order drop constraint cnst_no_w_id
go

-- Add partitioning constraints between 14401 and 16200
--

alter table t_warehouse add constraint cnst_w_id check (w_id between
convert(int,14401) and convert(int,16200))
go

alter table t_district add constraint cnst_d_w_id check (d_w_id
between convert(int,14401) and convert(int,16200))
go

alter table t_customer add constraint cnst_c_w_id check (c_w_id
between convert(int,14401) and convert(int,16200))
go

```

```

alter table t_history add constraint cnst_h_w_id check (h_w_id
between convert(int,14401) and convert(int,16200))
go

alter table t_stock add constraint cnst_s_w_id check (s_w_id between
convert(int,14401) and convert(int,16200))
go

alter table t_orders add constraint cnst_o_w_id check (o_w_id between
convert(int,14401) and convert(int,16200))
go

alter table t_order_line add constraint cnst_ol_w_id check (ol_w_id
between convert(int,14401) and convert(int,16200))
go

alter table t_new_order add constraint cnst_no_w_id check (no_w_id
between convert(int,14401) and convert(int,16200))
go

```

add_constraints_t10.sql

```

--file 16201_to_18000\add_constraints_t.sql

set ansi_warnings on
set ansi_nulls on
go

use tpcc
go

-- Drop any existing partitioning constraints
--

alter table t_warehouse drop constraint cnst_w_id
go

alter table t_district drop constraint cnst_d_w_id
go

alter table t_customer drop constraint cnst_c_w_id
go

alter table t_history drop constraint cnst_h_w_id
go

alter table t_stock drop constraint cnst_s_w_id
go

alter table t_orders drop constraint cnst_o_w_id
go

alter table t_order_line drop constraint cnst_ol_w_id
go

alter table t_new_order drop constraint cnst_no_w_id
go

-- Add partitioning constraints between 16201 and 18000
--

alter table t_warehouse add constraint cnst_w_id check (w_id between
convert(int,16201) and convert(int,18000))
go

```

```

alter table t_district add constraint cnst_d_w_id check (d_w_id
between convert(int,16201) and convert(int,18000))
go

alter table t_customer add constraint cnst_c_w_id check (c_w_id
between convert(int,16201) and convert(int,18000))
go

alter table t_history add constraint cnst_h_w_id check (h_w_id
between convert(int,16201) and convert(int,18000))
go

alter table t_stock add constraint cnst_s_w_id check (s_w_id between
convert(int,16201) and convert(int,18000))
go

alter table t_orders add constraint cnst_o_w_id check (o_w_id between
convert(int,16201) and convert(int,18000))
go

alter table t_order_line add constraint cnst_ol_w_id check (ol_w_id
between convert(int,16201) and convert(int,18000))
go

alter table t_new_order add constraint cnst_no_w_id check (no_w_id
between convert(int,16201) and convert(int,18000))
go

```

add_constraints_t11.sql

```

--file 18001_to_19800\add_constraints_t.sql

set ansi_warnings on
set ansi_nulls on
go

use tpcc
go

-- Drop any existing partitioning constraints
--

alter table t_warehouse drop constraint cnst_w_id
go

alter table t_district drop constraint cnst_d_w_id
go

alter table t_customer drop constraint cnst_c_w_id
go

alter table t_history drop constraint cnst_h_w_id
go

alter table t_stock drop constraint cnst_s_w_id
go

alter table t_orders drop constraint cnst_o_w_id
go

alter table t_order_line drop constraint cnst_ol_w_id
go

alter table t_new_order drop constraint cnst_no_w_id
go

-- Add partitioning constraints between 18001 and 19800

```

```

--
alter table t_warehouse add constraint cnst_w_id check (w_id between
convert(int,18001) and convert(int,19800))
go

alter table t_district add constraint cnst_d_w_id check (d_w_id
between convert(int,18001) and convert(int,19800))
go

alter table t_customer add constraint cnst_c_w_id check (c_w_id
between convert(int,18001) and convert(int,19800))
go

alter table t_history add constraint cnst_h_w_id check (h_w_id
between convert(int,18001) and convert(int,19800))
go

alter table t_stock add constraint cnst_s_w_id check (s_w_id between
convert(int,18001) and convert(int,19800))
go

alter table t_orders add constraint cnst_o_w_id check (o_w_id between
convert(int,18001) and convert(int,19800))
go

alter table t_order_line add constraint cnst_ol_w_id check (ol_w_id
between convert(int,18001) and convert(int,19800))
go

alter table t_new_order add constraint cnst_no_w_id check (no_w_id
between convert(int,18001) and convert(int,19800))
go

```

add_constraints_t12.sql

```

--file 19801_to_21600\add_constraints_t.sql

set ansi_warnings on
set ansi_nulls on
go

use tpcc
go

-- Drop any existing partitioning constraints
--

alter table t_warehouse drop constraint cnst_w_id
go

alter table t_district drop constraint cnst_d_w_id
go

alter table t_customer drop constraint cnst_c_w_id
go

alter table t_history drop constraint cnst_h_w_id
go

alter table t_stock drop constraint cnst_s_w_id
go

alter table t_orders drop constraint cnst_o_w_id
go

```

```

alter table t_order_line drop constraint cnst_ol_w_id
go

alter table t_new_order drop constraint cnst_no_w_id
go

-- Add partitioning constraints between 19801 and 21600
--

alter table t_warehouse add constraint cnst_w_id check (w_id between
convert(int,19801) and convert(int,21600))
go

alter table t_district add constraint cnst_d_w_id check (d_w_id
between convert(int,19801) and convert(int,21600))
go

alter table t_customer add constraint cnst_c_w_id check (c_w_id
between convert(int,19801) and convert(int,21600))
go

alter table t_history add constraint cnst_h_w_id check (h_w_id
between convert(int,19801) and convert(int,21600))
go

alter table t_stock add constraint cnst_s_w_id check (s_w_id between
convert(int,19801) and convert(int,21600))
go

alter table t_orders add constraint cnst_o_w_id check (o_w_id between
convert(int,19801) and convert(int,21600))
go

alter table t_order_line add constraint cnst_ol_w_id check (ol_w_id
between convert(int,19801) and convert(int,21600))
go

alter table t_new_order add constraint cnst_no_w_id check (no_w_id
between convert(int,19801) and convert(int,21600))
go

```

add_constraints_t13.sql

```

--file 23401_to_25200\add_constraints_t.sql

set ansi_warnings on
set ansi_nulls on
go

use tpcc
go

-- Drop any existing partitioning constraints
--

alter table t_warehouse drop constraint cnst_w_id
go

alter table t_district drop constraint cnst_d_w_id
go

alter table t_customer drop constraint cnst_c_w_id
go

alter table t_history drop constraint cnst_h_w_id
go

```

```

alter table t_stock drop constraint cnst_s_w_id
go

alter table t_orders drop constraint cnst_o_w_id
go

alter table t_order_line drop constraint cnst_ol_w_id
go

alter table t_new_order drop constraint cnst_no_w_id
go

-- Add partitioning constraints between 23401 and 25200
--

alter table t_warehouse add constraint cnst_w_id check (w_id between
convert(int,23401) and convert(int,25200))
go

alter table t_district add constraint cnst_d_w_id check (d_w_id
between convert(int,23401) and convert(int,25200))
go

alter table t_customer add constraint cnst_c_w_id check (c_w_id
between convert(int,23401) and convert(int,25200))
go

alter table t_history add constraint cnst_h_w_id check (h_w_id
between convert(int,23401) and convert(int,25200))
go

alter table t_stock add constraint cnst_s_w_id check (s_w_id between
convert(int,23401) and convert(int,25200))
go

alter table t_orders add constraint cnst_o_w_id check (o_w_id between
convert(int,23401) and convert(int,25200))
go

alter table t_order_line add constraint cnst_ol_w_id check (ol_w_id
between convert(int,23401) and convert(int,25200))
go

alter table t_new_order add constraint cnst_no_w_id check (no_w_id
between convert(int,23401) and convert(int,25200))
go

```

add_constraints_t14.sql

```

--file 23401_to_25200\add_constraints_t.sql

set ansi_warnings on
set ansi_nulls on
go

use tpcc
go

-- Drop any existing partitioning constraints
--

alter table t_warehouse drop constraint cnst_w_id
go

alter table t_district drop constraint cnst_d_w_id
go

```

```

alter table t_customer drop constraint cnst_c_w_id
go

alter table t_history drop constraint cnst_h_w_id
go

alter table t_stock drop constraint cnst_s_w_id
go

alter table t_orders drop constraint cnst_o_w_id
go

alter table t_order_line drop constraint cnst_ol_w_id
go

alter table t_new_order drop constraint cnst_no_w_id
go

-- Add partitioning constraints between 23401 and 25200
--

alter table t_warehouse add constraint cnst_w_id check (w_id between
convert(int,23401) and convert(int,25200))
go

alter table t_district add constraint cnst_d_w_id check (d_w_id
between convert(int,23401) and convert(int,25200))
go

alter table t_customer add constraint cnst_c_w_id check (c_w_id
between convert(int,23401) and convert(int,25200))
go

alter table t_history add constraint cnst_h_w_id check (h_w_id
between convert(int,23401) and convert(int,25200))
go

alter table t_stock add constraint cnst_s_w_id check (s_w_id between
convert(int,23401) and convert(int,25200))
go

alter table t_orders add constraint cnst_o_w_id check (o_w_id between
convert(int,23401) and convert(int,25200))
go

alter table t_order_line add constraint cnst_ol_w_id check (ol_w_id
between convert(int,23401) and convert(int,25200))
go

alter table t_new_order add constraint cnst_no_w_id check (no_w_id
between convert(int,23401) and convert(int,25200))
go

```

add_constraints_t15.sql

```

--file 25201_to_27000\add_constraints_t.sql

set ansi_warnings on
set ansi_nulls on
go

use tpcc
go

-- Drop any existing partitioning constraints

```

```

--
alter table t_warehouse drop constraint cnst_w_id
go

alter table t_district drop constraint cnst_d_w_id
go

alter table t_customer drop constraint cnst_c_w_id
go

alter table t_history drop constraint cnst_h_w_id
go

alter table t_stock drop constraint cnst_s_w_id
go

alter table t_orders drop constraint cnst_o_w_id
go

alter table t_order_line drop constraint cnst_ol_w_id
go

alter table t_new_order drop constraint cnst_no_w_id
go

-- Add partitioning constraints between 25201 and 27000
--

alter table t_warehouse add constraint cnst_w_id check (w_id between
convert(int,25201) and convert(int,27000))
go

alter table t_district add constraint cnst_d_w_id check (d_w_id
between convert(int,25201) and convert(int,27000))
go

alter table t_customer add constraint cnst_c_w_id check (c_w_id
between convert(int,25201) and convert(int,27000))
go

alter table t_history add constraint cnst_h_w_id check (h_w_id
between convert(int,25201) and convert(int,27000))
go

alter table t_stock add constraint cnst_s_w_id check (s_w_id between
convert(int,25201) and convert(int,27000))
go

alter table t_orders add constraint cnst_o_w_id check (o_w_id between
convert(int,25201) and convert(int,27000))
go

alter table t_order_line add constraint cnst_ol_w_id check (ol_w_id
between convert(int,25201) and convert(int,27000))
go

alter table t_new_order add constraint cnst_no_w_id check (no_w_id
between convert(int,25201) and convert(int,27000))
go

add_constraints_t16.sql

--file 27001_to_28800\add_constraints_t.sql

set ansi_warnings on

```

```

set ansi_nulls on
go

use tpcc
go

-- Drop any existing partitioning constraints
--

alter table t_warehouse drop constraint cnst_w_id
go

alter table t_district drop constraint cnst_d_w_id
go

alter table t_customer drop constraint cnst_c_w_id
go

alter table t_history drop constraint cnst_h_w_id
go

alter table t_stock drop constraint cnst_s_w_id
go

alter table t_orders drop constraint cnst_o_w_id
go

alter table t_order_line drop constraint cnst_ol_w_id
go

alter table t_new_order drop constraint cnst_no_w_id
go

-- Add partitioning constraints between 27001 and 28800
--

alter table t_warehouse add constraint cnst_w_id check (w_id >=
convert(int,27001))
go

alter table t_district add constraint cnst_d_w_id check (d_w_id >=
convert(int,27001))
go

alter table t_customer add constraint cnst_c_w_id check (c_w_id >=
convert(int,27001))
go

alter table t_history add constraint cnst_h_w_id check (h_w_id >=
convert(int,27001))
go

alter table t_stock add constraint cnst_s_w_id check (s_w_id >=
convert(int,27001))
go

alter table t_orders add constraint cnst_o_w_id check (o_w_id >=
convert(int,27001))
go

alter table t_order_line add constraint cnst_ol_w_id check (ol_w_id >=
convert(int,27001))
go

alter table t_new_order add constraint cnst_no_w_id check (no_w_id
>= convert(int,27001))
go

```


add_remote_servers.sql

```
--file add_remote_servers.sql
--add remote servers for all the partitions
--
-- partition 1 is on host rtnode01
--
exec sp_dropserver rmt1, droplogins
exec sp_addlinkedserver 'rmt1'
exec sp_setnetname 'rmt1', 'rtnode01'
exec sp_serveroption 'rmt1', 'lazy schema validation', 'true'
exec sp_droplinkedsrvlogin rmt1, NULL
exec sp_addlinkedsrvlogin rmt1, 'false', 'sa', 'sa'

--
-- partition 2 is on host rtnode02
--
exec sp_dropserver rmt2, droplogins
exec sp_addlinkedserver 'rmt2'
exec sp_setnetname 'rmt2', 'rtnode02'
exec sp_serveroption 'rmt2', 'lazy schema validation', 'true'
exec sp_droplinkedsrvlogin rmt2, NULL
exec sp_addlinkedsrvlogin rmt2, 'false', 'sa', 'sa'

--
-- partition 3 is on host rtnode03
--
exec sp_dropserver rmt3, droplogins
exec sp_addlinkedserver 'rmt3'
exec sp_setnetname 'rmt3', 'rtnode03'
exec sp_serveroption 'rmt3', 'lazy schema validation', 'true'
exec sp_droplinkedsrvlogin rmt3, NULL
exec sp_addlinkedsrvlogin rmt3, 'false', 'sa', 'sa'

--
-- partition 4 is on host rtnode04
--
exec sp_dropserver rmt4, droplogins
exec sp_addlinkedserver 'rmt4'
exec sp_setnetname 'rmt4', 'rtnode04'
exec sp_serveroption 'rmt4', 'lazy schema validation', 'true'
exec sp_droplinkedsrvlogin rmt4, NULL
exec sp_addlinkedsrvlogin rmt4, 'false', 'sa', 'sa'

--
-- partition 5 is on host rtnode05
--
exec sp_dropserver rmt5, droplogins
exec sp_addlinkedserver 'rmt5'
exec sp_setnetname 'rmt5', 'rtnode05'
exec sp_serveroption 'rmt5', 'lazy schema validation', 'true'
exec sp_droplinkedsrvlogin rmt5, NULL
exec sp_addlinkedsrvlogin rmt5, 'false', 'sa', 'sa'

--
-- partition 6 is on host rtnode06
--
exec sp_dropserver rmt6, droplogins
exec sp_addlinkedserver 'rmt6'
exec sp_setnetname 'rmt6', 'rtnode06'
exec sp_serveroption 'rmt6', 'lazy schema validation', 'true'
exec sp_droplinkedsrvlogin rmt6, NULL
exec sp_addlinkedsrvlogin rmt6, 'false', 'sa', 'sa'

--
-- partition 7 is on host rtnode07
--
exec sp_dropserver rmt7, droplogins
exec sp_addlinkedserver 'rmt7'
exec sp_setnetname 'rmt7', 'rtnode07'
exec sp_serveroption 'rmt7', 'lazy schema validation', 'true'
exec sp_droplinkedsrvlogin rmt7, NULL
exec sp_addlinkedsrvlogin rmt7, 'false', 'sa', 'sa'

--
-- partition 8 is on host rtnode08
--
exec sp_dropserver rmt8, droplogins
exec sp_addlinkedserver 'rmt8'
exec sp_setnetname 'rmt8', 'rtnode08'
exec sp_serveroption 'rmt8', 'lazy schema validation', 'true'
exec sp_droplinkedsrvlogin rmt8, NULL
exec sp_addlinkedsrvlogin rmt8, 'false', 'sa', 'sa'

--
-- partition 9 is on host rtnode09
--
exec sp_dropserver rmt9, droplogins
exec sp_addlinkedserver 'rmt9'
exec sp_setnetname 'rmt9', 'rtnode09'
exec sp_serveroption 'rmt9', 'lazy schema validation', 'true'
exec sp_droplinkedsrvlogin rmt9, NULL
exec sp_addlinkedsrvlogin rmt9, 'false', 'sa', 'sa'

--
-- partition 10 is on host rtnode10
--
exec sp_dropserver rmt10, droplogins
exec sp_addlinkedserver 'rmt10'
exec sp_setnetname 'rmt10', 'rtnode10'
exec sp_serveroption 'rmt10', 'lazy schema validation', 'true'
exec sp_droplinkedsrvlogin rmt10, NULL
exec sp_addlinkedsrvlogin rmt10, 'false', 'sa', 'sa'

--
-- partition 11 is on host rtnode11
--
exec sp_dropserver rmt11, droplogins
exec sp_addlinkedserver 'rmt11'
exec sp_setnetname 'rmt11', 'rtnode11'
exec sp_serveroption 'rmt11', 'lazy schema validation', 'true'
exec sp_droplinkedsrvlogin rmt11, NULL
exec sp_addlinkedsrvlogin rmt11, 'false', 'sa', 'sa'

--
-- partition 12 is on host rtnode12
--
exec sp_dropserver rmt12, droplogins
exec sp_addlinkedserver 'rmt12'
exec sp_setnetname 'rmt12', 'rtnode12'
exec sp_serveroption 'rmt12', 'lazy schema validation', 'true'
```

```

exec sp_droplinkedserver rmt12, NULL
exec sp_addlinkedserver rmt12, 'false', 'sa', 'sa'

--
-- partition 13 is on host rtnode13
--

exec sp_dropserver rmt13, droplogins
exec sp_addlinkedserver 'rmt13'
exec sp_setnetname 'rmt13', 'rtnode13'
exec sp_serveroption 'rmt13', 'lazy schema validation', 'true'
exec sp_droplinkedserver rmt13, NULL
exec sp_addlinkedserver rmt13, 'false', 'sa', 'sa'

--
-- partition 14 is on host rtnode14
--

exec sp_dropserver rmt14, droplogins
exec sp_addlinkedserver 'rmt14'
exec sp_setnetname 'rmt14', 'rtnode14'
exec sp_serveroption 'rmt14', 'lazy schema validation', 'true'
exec sp_droplinkedserver rmt14, NULL
exec sp_addlinkedserver rmt14, 'false', 'sa', 'sa'

--
-- partition 15 is on host rtnode15
--

exec sp_dropserver rmt15, droplogins
exec sp_addlinkedserver 'rmt15'
exec sp_setnetname 'rmt15', 'rtnode15'
exec sp_serveroption 'rmt15', 'lazy schema validation', 'true'
exec sp_droplinkedserver rmt15, NULL
exec sp_addlinkedserver rmt15, 'false', 'sa', 'sa'

--
-- partition 16 is on host rtnode16
--

exec sp_dropserver rmt16, droplogins
exec sp_addlinkedserver 'rmt16'
exec sp_setnetname 'rmt16', 'rtnode16'
exec sp_serveroption 'rmt16', 'lazy schema validation', 'true'
exec sp_droplinkedserver rmt16, NULL
exec sp_addlinkedserver rmt16, 'false', 'sa', 'sa'

```

add_views1.sql

```

-- file 1_to_1800\add_views.sql

set ansi_warnings on
set ansi_nulls on
go

use tpcc
go

-- drop any existing views
if object_id('warehouse') is not null drop view warehouse
if object_id('district') is not null drop view district
if object_id('customer') is not null drop view customer
if object_id('history') is not null drop view history
if object_id('stock') is not null drop view stock
if object_id('orders') is not null drop view orders

```

```

if object_id('order_line') is not null drop view order_line
if object_id('new_order') is not null drop view new_order
if object_id('item') is not null drop view item
go

--add views for partition 1
create view warehouse as
select * from /*rmt1.tpc.dbo.t_warehouse
union all
select * from rmt2.tpc.dbo.t_warehouse
union all
select * from rmt3.tpc.dbo.t_warehouse
union all
select * from rmt4.tpc.dbo.t_warehouse
union all
select * from rmt5.tpc.dbo.t_warehouse
union all
select * from rmt6.tpc.dbo.t_warehouse
union all
select * from rmt7.tpc.dbo.t_warehouse
union all
select * from rmt8.tpc.dbo.t_warehouse
union all
select * from rmt9.tpc.dbo.t_warehouse
union all
select * from rmt10.tpc.dbo.t_warehouse
union all
select * from rmt11.tpc.dbo.t_warehouse
union all
select * from rmt12.tpc.dbo.t_warehouse
union all
select * from rmt13.tpc.dbo.t_warehouse
union all
select * from rmt14.tpc.dbo.t_warehouse
union all
select * from rmt15.tpc.dbo.t_warehouse
union all
select * from rmt16.tpc.dbo.t_warehouse
go

create view district as
select * from /*rmt1.tpc.dbo.t_district
union all
select * from rmt2.tpc.dbo.t_district
union all
select * from rmt3.tpc.dbo.t_district
union all
select * from rmt4.tpc.dbo.t_district
union all
select * from rmt5.tpc.dbo.t_district
union all
select * from rmt6.tpc.dbo.t_district
union all
select * from rmt7.tpc.dbo.t_district
union all
select * from rmt8.tpc.dbo.t_district
union all
select * from rmt9.tpc.dbo.t_district
union all
select * from rmt10.tpc.dbo.t_district
union all
select * from rmt11.tpc.dbo.t_district
union all
select * from rmt12.tpc.dbo.t_district
union all
select * from rmt13.tpc.dbo.t_district
union all
select * from rmt14.tpc.dbo.t_district

```

```

union all
select * from rmt15.tpcc.dbo.t_district
union all
select * from rmt16.tpcc.dbo.t_district
go

```

```

create view customer as
select * from /*rmt1.tpcc.dbo.*/t_customer
union all
select * from rmt2.tpcc.dbo.t_customer
union all
select * from rmt3.tpcc.dbo.t_customer
union all
select * from rmt4.tpcc.dbo.t_customer
union all
select * from rmt5.tpcc.dbo.t_customer
union all
select * from rmt6.tpcc.dbo.t_customer
union all
select * from rmt7.tpcc.dbo.t_customer
union all
select * from rmt8.tpcc.dbo.t_customer
union all
select * from rmt9.tpcc.dbo.t_customer
union all
select * from rmt10.tpcc.dbo.t_customer
union all
select * from rmt11.tpcc.dbo.t_customer
union all
select * from rmt12.tpcc.dbo.t_customer
union all
select * from rmt13.tpcc.dbo.t_customer
union all
select * from rmt14.tpcc.dbo.t_customer
union all
select * from rmt15.tpcc.dbo.t_customer
union all
select * from rmt16.tpcc.dbo.t_customer
go

```

```

create view history as
select * from /*rmt1.tpcc.dbo.*/t_history
union all
select * from rmt2.tpcc.dbo.t_history
union all
select * from rmt3.tpcc.dbo.t_history
union all
select * from rmt4.tpcc.dbo.t_history
union all
select * from rmt5.tpcc.dbo.t_history
union all
select * from rmt6.tpcc.dbo.t_history
union all
select * from rmt7.tpcc.dbo.t_history
union all
select * from rmt8.tpcc.dbo.t_history
union all
select * from rmt9.tpcc.dbo.t_history
union all
select * from rmt10.tpcc.dbo.t_history
union all
select * from rmt11.tpcc.dbo.t_history
union all
select * from rmt12.tpcc.dbo.t_history
union all
select * from rmt13.tpcc.dbo.t_history
union all
select * from rmt14.tpcc.dbo.t_history

```

```

union all
select * from rmt15.tpcc.dbo.t_history
union all
select * from rmt16.tpcc.dbo.t_history
go

```

```

create view stock as
select * from /*rmt1.tpcc.dbo.*/t_stock
union all
select * from rmt2.tpcc.dbo.t_stock
union all
select * from rmt3.tpcc.dbo.t_stock
union all
select * from rmt4.tpcc.dbo.t_stock
union all
select * from rmt5.tpcc.dbo.t_stock
union all
select * from rmt6.tpcc.dbo.t_stock
union all
select * from rmt7.tpcc.dbo.t_stock
union all
select * from rmt8.tpcc.dbo.t_stock
union all
select * from rmt9.tpcc.dbo.t_stock
union all
select * from rmt10.tpcc.dbo.t_stock
union all
select * from rmt11.tpcc.dbo.t_stock
union all
select * from rmt12.tpcc.dbo.t_stock
union all
select * from rmt13.tpcc.dbo.t_stock
union all
select * from rmt14.tpcc.dbo.t_stock
union all
select * from rmt15.tpcc.dbo.t_stock
union all
select * from rmt16.tpcc.dbo.t_stock
go

```

```

create view orders as
select * from /*rmt1.tpcc.dbo.*/t_orders
union all
select * from rmt2.tpcc.dbo.t_orders
union all
select * from rmt3.tpcc.dbo.t_orders
union all
select * from rmt4.tpcc.dbo.t_orders
union all
select * from rmt5.tpcc.dbo.t_orders
union all
select * from rmt6.tpcc.dbo.t_orders
union all
select * from rmt7.tpcc.dbo.t_orders
union all
select * from rmt8.tpcc.dbo.t_orders
union all
select * from rmt9.tpcc.dbo.t_orders
union all
select * from rmt10.tpcc.dbo.t_orders
union all
select * from rmt11.tpcc.dbo.t_orders
union all
select * from rmt12.tpcc.dbo.t_orders
union all
select * from rmt13.tpcc.dbo.t_orders
union all
select * from rmt14.tpcc.dbo.t_orders

```

```

union all
select * from rmt15.tpcc.dbo.t_orders
union all
select * from rmt16.tpcc.dbo.t_orders
go

create view order_line as
select * from /*rmt1.tpcc.dbo.*/t_order_line
union all
select * from rmt2.tpcc.dbo.t_order_line
union all
select * from rmt3.tpcc.dbo.t_order_line
union all
select * from rmt4.tpcc.dbo.t_order_line
union all
select * from rmt5.tpcc.dbo.t_order_line
union all
select * from rmt6.tpcc.dbo.t_order_line
union all
select * from rmt7.tpcc.dbo.t_order_line
union all
select * from rmt8.tpcc.dbo.t_order_line
union all
select * from rmt9.tpcc.dbo.t_order_line
union all
select * from rmt10.tpcc.dbo.t_order_line
union all
select * from rmt11.tpcc.dbo.t_order_line
union all
select * from rmt12.tpcc.dbo.t_order_line
union all
select * from rmt13.tpcc.dbo.t_order_line
union all
select * from rmt14.tpcc.dbo.t_order_line
union all
select * from rmt15.tpcc.dbo.t_order_line
union all
select * from rmt16.tpcc.dbo.t_order_line
go

create view new_order as
select * from /*rmt1.tpcc.dbo.*/t_new_order
union all
select * from rmt2.tpcc.dbo.t_new_order
union all
select * from rmt3.tpcc.dbo.t_new_order
union all
select * from rmt4.tpcc.dbo.t_new_order
union all
select * from rmt5.tpcc.dbo.t_new_order
union all
select * from rmt6.tpcc.dbo.t_new_order
union all
select * from rmt7.tpcc.dbo.t_new_order
union all
select * from rmt8.tpcc.dbo.t_new_order
union all
select * from rmt9.tpcc.dbo.t_new_order
union all
select * from rmt10.tpcc.dbo.t_new_order
union all
select * from rmt11.tpcc.dbo.t_new_order
union all
select * from rmt12.tpcc.dbo.t_new_order
union all
select * from rmt13.tpcc.dbo.t_new_order
union all
select * from rmt14.tpcc.dbo.t_new_order

```

```

union all
select * from rmt15.tpcc.dbo.t_new_order
union all
select * from rmt16.tpcc.dbo.t_new_order
go

```

```

create view item as
select * from t_item
go

```

add_views2.sql

```
-- file 1801_to_3600\add_views.sql
```

```

set ansi_warnings on
set ansi_nulls on
go

```

```

use tpcc
go

```

```

-- drop any existing views
if object_id('warehouse') is not null drop view warehouse
if object_id('district') is not null drop view district
if object_id('customer') is not null drop view customer
if object_id('history') is not null drop view history
if object_id('stock') is not null drop view stock
if object_id('orders') is not null drop view orders
if object_id('order_line') is not null drop view order_line
if object_id('new_order') is not null drop view new_order
if object_id('item') is not null drop view item
go

```

```

--add views for partition 2
create view warehouse as
select * from /*rmt2.tpcc.dbo.*/t_warehouse
union all
select * from rmt1.tpcc.dbo.t_warehouse
union all
select * from rmt3.tpcc.dbo.t_warehouse
union all
select * from rmt4.tpcc.dbo.t_warehouse
union all
select * from rmt5.tpcc.dbo.t_warehouse
union all
select * from rmt6.tpcc.dbo.t_warehouse
union all
select * from rmt7.tpcc.dbo.t_warehouse
union all
select * from rmt8.tpcc.dbo.t_warehouse
union all
select * from rmt9.tpcc.dbo.t_warehouse
union all
select * from rmt10.tpcc.dbo.t_warehouse
union all
select * from rmt11.tpcc.dbo.t_warehouse
union all
select * from rmt12.tpcc.dbo.t_warehouse
union all
select * from rmt13.tpcc.dbo.t_warehouse
union all
select * from rmt14.tpcc.dbo.t_warehouse
union all
select * from rmt15.tpcc.dbo.t_warehouse
union all
select * from rmt16.tpcc.dbo.t_warehouse

```

go

```
create view district as
select * from /*rmt2.tpc.dbo.*/t_district
union all
select * from rmt1.tpc.dbo.t_district
union all
select * from rmt3.tpc.dbo.t_district
union all
select * from rmt4.tpc.dbo.t_district
union all
select * from rmt5.tpc.dbo.t_district
union all
select * from rmt6.tpc.dbo.t_district
union all
select * from rmt7.tpc.dbo.t_district
union all
select * from rmt8.tpc.dbo.t_district
union all
select * from rmt9.tpc.dbo.t_district
union all
select * from rmt10.tpc.dbo.t_district
union all
select * from rmt11.tpc.dbo.t_district
union all
select * from rmt12.tpc.dbo.t_district
union all
select * from rmt13.tpc.dbo.t_district
union all
select * from rmt14.tpc.dbo.t_district
union all
select * from rmt15.tpc.dbo.t_district
union all
select * from rmt16.tpc.dbo.t_district
go
```

```
create view customer as
select * from /*rmt2.tpc.dbo.*/t_customer
union all
select * from rmt1.tpc.dbo.t_customer
union all
select * from rmt3.tpc.dbo.t_customer
union all
select * from rmt4.tpc.dbo.t_customer
union all
select * from rmt5.tpc.dbo.t_customer
union all
select * from rmt6.tpc.dbo.t_customer
union all
select * from rmt7.tpc.dbo.t_customer
union all
select * from rmt8.tpc.dbo.t_customer
union all
select * from rmt9.tpc.dbo.t_customer
union all
select * from rmt10.tpc.dbo.t_customer
union all
select * from rmt11.tpc.dbo.t_customer
union all
select * from rmt12.tpc.dbo.t_customer
union all
select * from rmt13.tpc.dbo.t_customer
union all
select * from rmt14.tpc.dbo.t_customer
union all
select * from rmt15.tpc.dbo.t_customer
union all
select * from rmt16.tpc.dbo.t_customer
```

go

```
create view history as
select * from /*rmt2.tpc.dbo.*/t_history
union all
select * from rmt1.tpc.dbo.t_history
union all
select * from rmt3.tpc.dbo.t_history
union all
select * from rmt4.tpc.dbo.t_history
union all
select * from rmt5.tpc.dbo.t_history
union all
select * from rmt6.tpc.dbo.t_history
union all
select * from rmt7.tpc.dbo.t_history
union all
select * from rmt8.tpc.dbo.t_history
union all
select * from rmt9.tpc.dbo.t_history
union all
select * from rmt10.tpc.dbo.t_history
union all
select * from rmt11.tpc.dbo.t_history
union all
select * from rmt12.tpc.dbo.t_history
union all
select * from rmt13.tpc.dbo.t_history
union all
select * from rmt14.tpc.dbo.t_history
union all
select * from rmt15.tpc.dbo.t_history
union all
select * from rmt16.tpc.dbo.t_history
go
```

```
create view stock as
select * from /*rmt2.tpc.dbo.*/t_stock
union all
select * from rmt1.tpc.dbo.t_stock
union all
select * from rmt3.tpc.dbo.t_stock
union all
select * from rmt4.tpc.dbo.t_stock
union all
select * from rmt5.tpc.dbo.t_stock
union all
select * from rmt6.tpc.dbo.t_stock
union all
select * from rmt7.tpc.dbo.t_stock
union all
select * from rmt8.tpc.dbo.t_stock
union all
select * from rmt9.tpc.dbo.t_stock
union all
select * from rmt10.tpc.dbo.t_stock
union all
select * from rmt11.tpc.dbo.t_stock
union all
select * from rmt12.tpc.dbo.t_stock
union all
select * from rmt13.tpc.dbo.t_stock
union all
select * from rmt14.tpc.dbo.t_stock
union all
select * from rmt15.tpc.dbo.t_stock
union all
select * from rmt16.tpc.dbo.t_stock
```

```

go

create view orders as
select * from /*rmt2.tpc.dbo.*/t_orders
union all
select * from rmt1.tpc.dbo.t_orders
union all
select * from rmt3.tpc.dbo.t_orders
union all
select * from rmt4.tpc.dbo.t_orders
union all
select * from rmt5.tpc.dbo.t_orders
union all
select * from rmt6.tpc.dbo.t_orders
union all
select * from rmt7.tpc.dbo.t_orders
union all
select * from rmt8.tpc.dbo.t_orders
union all
select * from rmt9.tpc.dbo.t_orders
union all
select * from rmt10.tpc.dbo.t_orders
union all
select * from rmt11.tpc.dbo.t_orders
union all
select * from rmt12.tpc.dbo.t_orders
union all
select * from rmt13.tpc.dbo.t_orders
union all
select * from rmt14.tpc.dbo.t_orders
union all
select * from rmt15.tpc.dbo.t_orders
union all
select * from rmt16.tpc.dbo.t_orders
go

create view order_line as
select * from /*rmt2.tpc.dbo.*/t_order_line
union all
select * from rmt1.tpc.dbo.t_order_line
union all
select * from rmt3.tpc.dbo.t_order_line
union all
select * from rmt4.tpc.dbo.t_order_line
union all
select * from rmt5.tpc.dbo.t_order_line
union all
select * from rmt6.tpc.dbo.t_order_line
union all
select * from rmt7.tpc.dbo.t_order_line
union all
select * from rmt8.tpc.dbo.t_order_line
union all
select * from rmt9.tpc.dbo.t_order_line
union all
select * from rmt10.tpc.dbo.t_order_line
union all
select * from rmt11.tpc.dbo.t_order_line
union all
select * from rmt12.tpc.dbo.t_order_line
union all
select * from rmt13.tpc.dbo.t_order_line
union all
select * from rmt14.tpc.dbo.t_order_line
union all
select * from rmt15.tpc.dbo.t_order_line
union all
select * from rmt16.tpc.dbo.t_order_line

```

```

go

create view new_order as
select * from /*rmt2.tpc.dbo.*/t_new_order
union all
select * from rmt1.tpc.dbo.t_new_order
union all
select * from rmt3.tpc.dbo.t_new_order
union all
select * from rmt4.tpc.dbo.t_new_order
union all
select * from rmt5.tpc.dbo.t_new_order
union all
select * from rmt6.tpc.dbo.t_new_order
union all
select * from rmt7.tpc.dbo.t_new_order
union all
select * from rmt8.tpc.dbo.t_new_order
union all
select * from rmt9.tpc.dbo.t_new_order
union all
select * from rmt10.tpc.dbo.t_new_order
union all
select * from rmt11.tpc.dbo.t_new_order
union all
select * from rmt12.tpc.dbo.t_new_order
union all
select * from rmt13.tpc.dbo.t_new_order
union all
select * from rmt14.tpc.dbo.t_new_order
union all
select * from rmt15.tpc.dbo.t_new_order
union all
select * from rmt16.tpc.dbo.t_new_order
go

create view item as
select * from t_item
go

add_views3.sql

-- file 3601_to_5400\add_views.sql

set ansi_warnings on
set ansi_nulls on
go

use tpcc
go

-- drop any existing views
if object_id('warehouse') is not null drop view warehouse
if object_id('district') is not null drop view district
if object_id('customer') is not null drop view customer
if object_id('history') is not null drop view history
if object_id('stock') is not null drop view stock
if object_id('orders') is not null drop view orders
if object_id('order_line') is not null drop view order_line
if object_id('new_order') is not null drop view new_order
if object_id('item') is not null drop view item
go

--add views for partition 3
create view warehouse as
select * from /*rmt3.tpc.dbo.*/t_warehouse

```

```

union all
select * from rmt1.tpc.dbo.t_warehouse
union all
select * from rmt2.tpc.dbo.t_warehouse
union all
select * from rmt4.tpc.dbo.t_warehouse
union all
select * from rmt5.tpc.dbo.t_warehouse
union all
select * from rmt6.tpc.dbo.t_warehouse
union all
select * from rmt7.tpc.dbo.t_warehouse
union all
select * from rmt8.tpc.dbo.t_warehouse
union all
select * from rmt9.tpc.dbo.t_warehouse
union all
select * from rmt10.tpc.dbo.t_warehouse
union all
select * from rmt11.tpc.dbo.t_warehouse
union all
select * from rmt12.tpc.dbo.t_warehouse
union all
select * from rmt13.tpc.dbo.t_warehouse
union all
select * from rmt14.tpc.dbo.t_warehouse
union all
select * from rmt15.tpc.dbo.t_warehouse
union all
select * from rmt16.tpc.dbo.t_warehouse
go

```

```

create view district as
select * from /*rmt3.tpc.dbo.*/t_district
union all
select * from rmt1.tpc.dbo.t_district
union all
select * from rmt2.tpc.dbo.t_district
union all
select * from rmt4.tpc.dbo.t_district
union all
select * from rmt5.tpc.dbo.t_district
union all
select * from rmt6.tpc.dbo.t_district
union all
select * from rmt7.tpc.dbo.t_district
union all
select * from rmt8.tpc.dbo.t_district
union all
select * from rmt9.tpc.dbo.t_district
union all
select * from rmt10.tpc.dbo.t_district
union all
select * from rmt11.tpc.dbo.t_district
union all
select * from rmt12.tpc.dbo.t_district
union all
select * from rmt13.tpc.dbo.t_district
union all
select * from rmt14.tpc.dbo.t_district
union all
select * from rmt15.tpc.dbo.t_district
union all
select * from rmt16.tpc.dbo.t_district
go

```

```

create view customer as
select * from /*rmt3.tpc.dbo.*/t_customer

```

```

union all
select * from rmt1.tpc.dbo.t_customer
union all
select * from rmt2.tpc.dbo.t_customer
union all
select * from rmt4.tpc.dbo.t_customer
union all
select * from rmt5.tpc.dbo.t_customer
union all
select * from rmt6.tpc.dbo.t_customer
union all
select * from rmt7.tpc.dbo.t_customer
union all
select * from rmt8.tpc.dbo.t_customer
union all
select * from rmt9.tpc.dbo.t_customer
union all
select * from rmt10.tpc.dbo.t_customer
union all
select * from rmt11.tpc.dbo.t_customer
union all
select * from rmt12.tpc.dbo.t_customer
union all
select * from rmt13.tpc.dbo.t_customer
union all
select * from rmt14.tpc.dbo.t_customer
union all
select * from rmt15.tpc.dbo.t_customer
union all
select * from rmt16.tpc.dbo.t_customer
go

```

```

create view history as
select * from /*rmt3.tpc.dbo.*/t_history
union all
select * from rmt1.tpc.dbo.t_history
union all
select * from rmt2.tpc.dbo.t_history
union all
select * from rmt4.tpc.dbo.t_history
union all
select * from rmt5.tpc.dbo.t_history
union all
select * from rmt6.tpc.dbo.t_history
union all
select * from rmt7.tpc.dbo.t_history
union all
select * from rmt8.tpc.dbo.t_history
union all
select * from rmt9.tpc.dbo.t_history
union all
select * from rmt10.tpc.dbo.t_history
union all
select * from rmt11.tpc.dbo.t_history
union all
select * from rmt12.tpc.dbo.t_history
union all
select * from rmt13.tpc.dbo.t_history
union all
select * from rmt14.tpc.dbo.t_history
union all
select * from rmt15.tpc.dbo.t_history
union all
select * from rmt16.tpc.dbo.t_history
go

```

```

create view stock as
select * from /*rmt3.tpc.dbo.*/t_stock

```

```

union all
select * from rmt1.tpcc.dbo.t_stock
union all
select * from rmt2.tpcc.dbo.t_stock
union all
select * from rmt4.tpcc.dbo.t_stock
union all
select * from rmt5.tpcc.dbo.t_stock
union all
select * from rmt6.tpcc.dbo.t_stock
union all
select * from rmt7.tpcc.dbo.t_stock
union all
select * from rmt8.tpcc.dbo.t_stock
union all
select * from rmt9.tpcc.dbo.t_stock
union all
select * from rmt10.tpcc.dbo.t_stock
union all
select * from rmt11.tpcc.dbo.t_stock
union all
select * from rmt12.tpcc.dbo.t_stock
union all
select * from rmt13.tpcc.dbo.t_stock
union all
select * from rmt14.tpcc.dbo.t_stock
union all
select * from rmt15.tpcc.dbo.t_stock
union all
select * from rmt16.tpcc.dbo.t_stock
go

```

```

create view orders as
select * from /*rmt3.tpcc.dbo.*/t_orders
union all
select * from rmt1.tpcc.dbo.t_orders
union all
select * from rmt2.tpcc.dbo.t_orders
union all
select * from rmt4.tpcc.dbo.t_orders
union all
select * from rmt5.tpcc.dbo.t_orders
union all
select * from rmt6.tpcc.dbo.t_orders
union all
select * from rmt7.tpcc.dbo.t_orders
union all
select * from rmt8.tpcc.dbo.t_orders
union all
select * from rmt9.tpcc.dbo.t_orders
union all
select * from rmt10.tpcc.dbo.t_orders
union all
select * from rmt11.tpcc.dbo.t_orders
union all
select * from rmt12.tpcc.dbo.t_orders
union all
select * from rmt13.tpcc.dbo.t_orders
union all
select * from rmt14.tpcc.dbo.t_orders
union all
select * from rmt15.tpcc.dbo.t_orders
union all
select * from rmt16.tpcc.dbo.t_orders
go

```

```

create view order_line as
select * from /*rmt3.tpcc.dbo.*/t_order_line

```

```

union all
select * from rmt1.tpcc.dbo.t_order_line
union all
select * from rmt2.tpcc.dbo.t_order_line
union all
select * from rmt4.tpcc.dbo.t_order_line
union all
select * from rmt5.tpcc.dbo.t_order_line
union all
select * from rmt6.tpcc.dbo.t_order_line
union all
select * from rmt7.tpcc.dbo.t_order_line
union all
select * from rmt8.tpcc.dbo.t_order_line
union all
select * from rmt9.tpcc.dbo.t_order_line
union all
select * from rmt10.tpcc.dbo.t_order_line
union all
select * from rmt11.tpcc.dbo.t_order_line
union all
select * from rmt12.tpcc.dbo.t_order_line
union all
select * from rmt13.tpcc.dbo.t_order_line
union all
select * from rmt14.tpcc.dbo.t_order_line
union all
select * from rmt15.tpcc.dbo.t_order_line
union all
select * from rmt16.tpcc.dbo.t_order_line
go

```

```

create view new_order as
select * from /*rmt3.tpcc.dbo.*/t_new_order
union all
select * from rmt1.tpcc.dbo.t_new_order
union all
select * from rmt2.tpcc.dbo.t_new_order
union all
select * from rmt4.tpcc.dbo.t_new_order
union all
select * from rmt5.tpcc.dbo.t_new_order
union all
select * from rmt6.tpcc.dbo.t_new_order
union all
select * from rmt7.tpcc.dbo.t_new_order
union all
select * from rmt8.tpcc.dbo.t_new_order
union all
select * from rmt9.tpcc.dbo.t_new_order
union all
select * from rmt10.tpcc.dbo.t_new_order
union all
select * from rmt11.tpcc.dbo.t_new_order
union all
select * from rmt12.tpcc.dbo.t_new_order
union all
select * from rmt13.tpcc.dbo.t_new_order
union all
select * from rmt14.tpcc.dbo.t_new_order
union all
select * from rmt15.tpcc.dbo.t_new_order
union all
select * from rmt16.tpcc.dbo.t_new_order
go

```

```

create view item as
select * from t_item

```



```
go
```

add_views4.sql

```
-- file 5401_to_7200\add_views.sql
```

```
set ansi_warnings on  
set ansi_nulls on  
go
```

```
use tpcc  
go
```

```
-- drop any existing views  
if object_id('warehouse') is not null drop view warehouse  
if object_id('district') is not null drop view district  
if object_id('customer') is not null drop view customer  
if object_id('history') is not null drop view history  
if object_id('stock') is not null drop view stock  
if object_id('orders') is not null drop view orders  
if object_id('order_line') is not null drop view order_line  
if object_id('new_order') is not null drop view new_order  
if object_id('item') is not null drop view item  
go
```

```
--add views for partition 4  
create view warehouse as  
select * from /*rmt4.tpcc.dbo.t_warehouse  
union all  
select * from rmt1.tpcc.dbo.t_warehouse  
union all  
select * from rmt2.tpcc.dbo.t_warehouse  
union all  
select * from rmt3.tpcc.dbo.t_warehouse  
union all  
select * from rmt5.tpcc.dbo.t_warehouse  
union all  
select * from rmt6.tpcc.dbo.t_warehouse  
union all  
select * from rmt7.tpcc.dbo.t_warehouse  
union all  
select * from rmt8.tpcc.dbo.t_warehouse  
union all  
select * from rmt9.tpcc.dbo.t_warehouse  
union all  
select * from rmt10.tpcc.dbo.t_warehouse  
union all  
select * from rmt11.tpcc.dbo.t_warehouse  
union all  
select * from rmt12.tpcc.dbo.t_warehouse  
union all  
select * from rmt13.tpcc.dbo.t_warehouse  
union all  
select * from rmt14.tpcc.dbo.t_warehouse  
union all  
select * from rmt15.tpcc.dbo.t_warehouse  
union all  
select * from rmt16.tpcc.dbo.t_warehouse  
go
```

```
create view district as  
select * from /*rmt4.tpcc.dbo.t_district  
union all  
select * from rmt1.tpcc.dbo.t_district  
union all  
select * from rmt2.tpcc.dbo.t_district
```

```
union all  
select * from rmt3.tpcc.dbo.t_district  
union all  
select * from rmt5.tpcc.dbo.t_district  
union all  
select * from rmt6.tpcc.dbo.t_district  
union all  
select * from rmt7.tpcc.dbo.t_district  
union all  
select * from rmt8.tpcc.dbo.t_district  
union all  
select * from rmt9.tpcc.dbo.t_district  
union all  
select * from rmt10.tpcc.dbo.t_district  
union all  
select * from rmt11.tpcc.dbo.t_district  
union all  
select * from rmt12.tpcc.dbo.t_district  
union all  
select * from rmt13.tpcc.dbo.t_district  
union all  
select * from rmt14.tpcc.dbo.t_district  
union all  
select * from rmt15.tpcc.dbo.t_district  
union all  
select * from rmt16.tpcc.dbo.t_district  
go
```

```
create view customer as  
select * from /*rmt4.tpcc.dbo.t_customer  
union all  
select * from rmt1.tpcc.dbo.t_customer  
union all  
select * from rmt2.tpcc.dbo.t_customer  
union all  
select * from rmt3.tpcc.dbo.t_customer  
union all  
select * from rmt5.tpcc.dbo.t_customer  
union all  
select * from rmt6.tpcc.dbo.t_customer  
union all  
select * from rmt7.tpcc.dbo.t_customer  
union all  
select * from rmt8.tpcc.dbo.t_customer  
union all  
select * from rmt9.tpcc.dbo.t_customer  
union all  
select * from rmt10.tpcc.dbo.t_customer  
union all  
select * from rmt11.tpcc.dbo.t_customer  
union all  
select * from rmt12.tpcc.dbo.t_customer  
union all  
select * from rmt13.tpcc.dbo.t_customer  
union all  
select * from rmt14.tpcc.dbo.t_customer  
union all  
select * from rmt15.tpcc.dbo.t_customer  
union all  
select * from rmt16.tpcc.dbo.t_customer  
go
```

```
create view history as  
select * from /*rmt4.tpcc.dbo.t_history  
union all  
select * from rmt1.tpcc.dbo.t_history  
union all  
select * from rmt2.tpcc.dbo.t_history
```

```

union all
select * from rmt3.tpc.dbo.t_history
union all
select * from rmt5.tpc.dbo.t_history
union all
select * from rmt6.tpc.dbo.t_history
union all
select * from rmt7.tpc.dbo.t_history
union all
select * from rmt8.tpc.dbo.t_history
union all
select * from rmt9.tpc.dbo.t_history
union all
select * from rmt10.tpc.dbo.t_history
union all
select * from rmt11.tpc.dbo.t_history
union all
select * from rmt12.tpc.dbo.t_history
union all
select * from rmt13.tpc.dbo.t_history
union all
select * from rmt14.tpc.dbo.t_history
union all
select * from rmt15.tpc.dbo.t_history
union all
select * from rmt16.tpc.dbo.t_history
go

```

```

create view stock as
select * from /*rmt4.tpc.dbo.*/t_stock
union all
select * from rmt1.tpc.dbo.t_stock
union all
select * from rmt2.tpc.dbo.t_stock
union all
select * from rmt3.tpc.dbo.t_stock
union all
select * from rmt5.tpc.dbo.t_stock
union all
select * from rmt6.tpc.dbo.t_stock
union all
select * from rmt7.tpc.dbo.t_stock
union all
select * from rmt8.tpc.dbo.t_stock
union all
select * from rmt9.tpc.dbo.t_stock
union all
select * from rmt10.tpc.dbo.t_stock
union all
select * from rmt11.tpc.dbo.t_stock
union all
select * from rmt12.tpc.dbo.t_stock
union all
select * from rmt13.tpc.dbo.t_stock
union all
select * from rmt14.tpc.dbo.t_stock
union all
select * from rmt15.tpc.dbo.t_stock
union all
select * from rmt16.tpc.dbo.t_stock
go

```

```

create view orders as
select * from /*rmt4.tpc.dbo.*/t_orders
union all
select * from rmt1.tpc.dbo.t_orders
union all
select * from rmt2.tpc.dbo.t_orders

```

```

union all
select * from rmt3.tpc.dbo.t_orders
union all
select * from rmt5.tpc.dbo.t_orders
union all
select * from rmt6.tpc.dbo.t_orders
union all
select * from rmt7.tpc.dbo.t_orders
union all
select * from rmt8.tpc.dbo.t_orders
union all
select * from rmt9.tpc.dbo.t_orders
union all
select * from rmt10.tpc.dbo.t_orders
union all
select * from rmt11.tpc.dbo.t_orders
union all
select * from rmt12.tpc.dbo.t_orders
union all
select * from rmt13.tpc.dbo.t_orders
union all
select * from rmt14.tpc.dbo.t_orders
union all
select * from rmt15.tpc.dbo.t_orders
union all
select * from rmt16.tpc.dbo.t_orders
go

```

```

create view order_line as
select * from /*rmt4.tpc.dbo.*/t_order_line
union all
select * from rmt1.tpc.dbo.t_order_line
union all
select * from rmt2.tpc.dbo.t_order_line
union all
select * from rmt3.tpc.dbo.t_order_line
union all
select * from rmt5.tpc.dbo.t_order_line
union all
select * from rmt6.tpc.dbo.t_order_line
union all
select * from rmt7.tpc.dbo.t_order_line
union all
select * from rmt8.tpc.dbo.t_order_line
union all
select * from rmt9.tpc.dbo.t_order_line
union all
select * from rmt10.tpc.dbo.t_order_line
union all
select * from rmt11.tpc.dbo.t_order_line
union all
select * from rmt12.tpc.dbo.t_order_line
union all
select * from rmt13.tpc.dbo.t_order_line
union all
select * from rmt14.tpc.dbo.t_order_line
union all
select * from rmt15.tpc.dbo.t_order_line
union all
select * from rmt16.tpc.dbo.t_order_line
go

```

```

create view new_order as
select * from /*rmt4.tpc.dbo.*/t_new_order
union all
select * from rmt1.tpc.dbo.t_new_order
union all
select * from rmt2.tpc.dbo.t_new_order

```

```

union all
select * from rmt3.tpc.dbo.t_new_order
union all
select * from rmt5.tpc.dbo.t_new_order
union all
select * from rmt6.tpc.dbo.t_new_order
union all
select * from rmt7.tpc.dbo.t_new_order
union all
select * from rmt8.tpc.dbo.t_new_order
union all
select * from rmt9.tpc.dbo.t_new_order
union all
select * from rmt10.tpc.dbo.t_new_order
union all
select * from rmt11.tpc.dbo.t_new_order
union all
select * from rmt12.tpc.dbo.t_new_order
union all
select * from rmt13.tpc.dbo.t_new_order
union all
select * from rmt14.tpc.dbo.t_new_order
union all
select * from rmt15.tpc.dbo.t_new_order
union all
select * from rmt16.tpc.dbo.t_new_order
go

create view item as
select * from t_item
go

```

add_views5.sql

```

-- file 7201_to_9000\add_views.sql

set ansi_warnings on
set ansi_nulls on
go

use tpcc
go

-- drop any existing views
if object_id('warehouse') is not null drop view warehouse
if object_id('district') is not null drop view district
if object_id('customer') is not null drop view customer
if object_id('history') is not null drop view history
if object_id('stock') is not null drop view stock
if object_id('orders') is not null drop view orders
if object_id('order_line') is not null drop view order_line
if object_id('new_order') is not null drop view new_order
if object_id('item') is not null drop view item
go

--add views for partition 5
create view warehouse as
select * from /*rmt5.tpc.dbo.*/t_warehouse
union all
select * from rmt1.tpc.dbo.t_warehouse
union all
select * from rmt2.tpc.dbo.t_warehouse
union all
select * from rmt3.tpc.dbo.t_warehouse
union all
select * from rmt4.tpc.dbo.t_warehouse

```

```

union all
select * from rmt6.tpc.dbo.t_warehouse
union all
select * from rmt7.tpc.dbo.t_warehouse
union all
select * from rmt8.tpc.dbo.t_warehouse
union all
select * from rmt9.tpc.dbo.t_warehouse
union all
select * from rmt10.tpc.dbo.t_warehouse
union all
select * from rmt11.tpc.dbo.t_warehouse
union all
select * from rmt12.tpc.dbo.t_warehouse
union all
select * from rmt13.tpc.dbo.t_warehouse
union all
select * from rmt14.tpc.dbo.t_warehouse
union all
select * from rmt15.tpc.dbo.t_warehouse
union all
select * from rmt16.tpc.dbo.t_warehouse
go

create view district as
select * from /*rmt5.tpc.dbo.*/t_district
union all
select * from rmt1.tpc.dbo.t_district
union all
select * from rmt2.tpc.dbo.t_district
union all
select * from rmt3.tpc.dbo.t_district
union all
select * from rmt4.tpc.dbo.t_district
union all
select * from rmt6.tpc.dbo.t_district
union all
select * from rmt7.tpc.dbo.t_district
union all
select * from rmt8.tpc.dbo.t_district
union all
select * from rmt9.tpc.dbo.t_district
union all
select * from rmt10.tpc.dbo.t_district
union all
select * from rmt11.tpc.dbo.t_district
union all
select * from rmt12.tpc.dbo.t_district
union all
select * from rmt13.tpc.dbo.t_district
union all
select * from rmt14.tpc.dbo.t_district
union all
select * from rmt15.tpc.dbo.t_district
union all
select * from rmt16.tpc.dbo.t_district
go

create view customer as
select * from /*rmt5.tpc.dbo.*/t_customer
union all
select * from rmt1.tpc.dbo.t_customer
union all
select * from rmt2.tpc.dbo.t_customer
union all
select * from rmt3.tpc.dbo.t_customer
union all
select * from rmt4.tpc.dbo.t_customer

```

```

union all
select * from rmt6.tpcc.dbo.t_customer
union all
select * from rmt7.tpcc.dbo.t_customer
union all
select * from rmt8.tpcc.dbo.t_customer
union all
select * from rmt9.tpcc.dbo.t_customer
union all
select * from rmt10.tpcc.dbo.t_customer
union all
select * from rmt11.tpcc.dbo.t_customer
union all
select * from rmt12.tpcc.dbo.t_customer
union all
select * from rmt13.tpcc.dbo.t_customer
union all
select * from rmt14.tpcc.dbo.t_customer
union all
select * from rmt15.tpcc.dbo.t_customer
union all
select * from rmt16.tpcc.dbo.t_customer
go

```

```

create view history as
select * from /*rmt5.tpcc.dbo.*/t_history
union all
select * from rmt1.tpcc.dbo.t_history
union all
select * from rmt2.tpcc.dbo.t_history
union all
select * from rmt3.tpcc.dbo.t_history
union all
select * from rmt4.tpcc.dbo.t_history
union all
select * from rmt6.tpcc.dbo.t_history
union all
select * from rmt7.tpcc.dbo.t_history
union all
select * from rmt8.tpcc.dbo.t_history
union all
select * from rmt9.tpcc.dbo.t_history
union all
select * from rmt10.tpcc.dbo.t_history
union all
select * from rmt11.tpcc.dbo.t_history
union all
select * from rmt12.tpcc.dbo.t_history
union all
select * from rmt13.tpcc.dbo.t_history
union all
select * from rmt14.tpcc.dbo.t_history
union all
select * from rmt15.tpcc.dbo.t_history
union all
select * from rmt16.tpcc.dbo.t_history
go

```

```

create view stock as
select * from /*rmt5.tpcc.dbo.*/t_stock
union all
select * from rmt1.tpcc.dbo.t_stock
union all
select * from rmt2.tpcc.dbo.t_stock
union all
select * from rmt3.tpcc.dbo.t_stock
union all
select * from rmt4.tpcc.dbo.t_stock

```

```

union all
select * from rmt6.tpcc.dbo.t_stock
union all
select * from rmt7.tpcc.dbo.t_stock
union all
select * from rmt8.tpcc.dbo.t_stock
union all
select * from rmt9.tpcc.dbo.t_stock
union all
select * from rmt10.tpcc.dbo.t_stock
union all
select * from rmt11.tpcc.dbo.t_stock
union all
select * from rmt12.tpcc.dbo.t_stock
union all
select * from rmt13.tpcc.dbo.t_stock
union all
select * from rmt14.tpcc.dbo.t_stock
union all
select * from rmt15.tpcc.dbo.t_stock
union all
select * from rmt16.tpcc.dbo.t_stock
go

```

```

create view orders as
select * from /*rmt5.tpcc.dbo.*/t_orders
union all
select * from rmt1.tpcc.dbo.t_orders
union all
select * from rmt2.tpcc.dbo.t_orders
union all
select * from rmt3.tpcc.dbo.t_orders
union all
select * from rmt4.tpcc.dbo.t_orders
union all
select * from rmt6.tpcc.dbo.t_orders
union all
select * from rmt7.tpcc.dbo.t_orders
union all
select * from rmt8.tpcc.dbo.t_orders
union all
select * from rmt9.tpcc.dbo.t_orders
union all
select * from rmt10.tpcc.dbo.t_orders
union all
select * from rmt11.tpcc.dbo.t_orders
union all
select * from rmt12.tpcc.dbo.t_orders
union all
select * from rmt13.tpcc.dbo.t_orders
union all
select * from rmt14.tpcc.dbo.t_orders
union all
select * from rmt15.tpcc.dbo.t_orders
union all
select * from rmt16.tpcc.dbo.t_orders
go

```

```

create view order_line as
select * from /*rmt5.tpcc.dbo.*/t_order_line
union all
select * from rmt1.tpcc.dbo.t_order_line
union all
select * from rmt2.tpcc.dbo.t_order_line
union all
select * from rmt3.tpcc.dbo.t_order_line
union all
select * from rmt4.tpcc.dbo.t_order_line

```

```

union all
select * from rmt6.tpcc.dbo.t_order_line
union all
select * from rmt7.tpcc.dbo.t_order_line
union all
select * from rmt8.tpcc.dbo.t_order_line
union all
select * from rmt9.tpcc.dbo.t_order_line
union all
select * from rmt10.tpcc.dbo.t_order_line
union all
select * from rmt11.tpcc.dbo.t_order_line
union all
select * from rmt12.tpcc.dbo.t_order_line
union all
select * from rmt13.tpcc.dbo.t_order_line
union all
select * from rmt14.tpcc.dbo.t_order_line
union all
select * from rmt15.tpcc.dbo.t_order_line
union all
select * from rmt16.tpcc.dbo.t_order_line
go

```

```

create view new_order as
select * from /*rmt5.tpcc.dbo.*/t_new_order
union all
select * from rmt1.tpcc.dbo.t_new_order
union all
select * from rmt2.tpcc.dbo.t_new_order
union all
select * from rmt3.tpcc.dbo.t_new_order
union all
select * from rmt4.tpcc.dbo.t_new_order
union all
select * from rmt6.tpcc.dbo.t_new_order
union all
select * from rmt7.tpcc.dbo.t_new_order
union all
select * from rmt8.tpcc.dbo.t_new_order
union all
select * from rmt9.tpcc.dbo.t_new_order
union all
select * from rmt10.tpcc.dbo.t_new_order
union all
select * from rmt11.tpcc.dbo.t_new_order
union all
select * from rmt12.tpcc.dbo.t_new_order
union all
select * from rmt13.tpcc.dbo.t_new_order
union all
select * from rmt14.tpcc.dbo.t_new_order
union all
select * from rmt15.tpcc.dbo.t_new_order
union all
select * from rmt16.tpcc.dbo.t_new_order
go

```

```

create view item as
select * from t_item
go

```

add_views6.sql

```
-- file 9001_to_10800\add_views.sql
```

```

set ansi_warnings on
set ansi_nulls on
go

use tpcc
go

-- drop any existing views
if object_id('warehouse') is not null drop view warehouse
if object_id('district') is not null drop view district
if object_id('customer') is not null drop view customer
if object_id('history') is not null drop view history
if object_id('stock') is not null drop view stock
if object_id('orders') is not null drop view orders
if object_id('order_line') is not null drop view order_line
if object_id('new_order') is not null drop view new_order
if object_id('item') is not null drop view item
go

--add views for partition 6
create view warehouse as
select * from /*rmt6.tpcc.dbo.*/t_warehouse
union all
select * from rmt1.tpcc.dbo.t_warehouse
union all
select * from rmt2.tpcc.dbo.t_warehouse
union all
select * from rmt3.tpcc.dbo.t_warehouse
union all
select * from rmt4.tpcc.dbo.t_warehouse
union all
select * from rmt5.tpcc.dbo.t_warehouse
union all
select * from rmt7.tpcc.dbo.t_warehouse
union all
select * from rmt8.tpcc.dbo.t_warehouse
union all
select * from rmt9.tpcc.dbo.t_warehouse
union all
select * from rmt10.tpcc.dbo.t_warehouse
union all
select * from rmt11.tpcc.dbo.t_warehouse
union all
select * from rmt12.tpcc.dbo.t_warehouse
union all
select * from rmt13.tpcc.dbo.t_warehouse
union all
select * from rmt14.tpcc.dbo.t_warehouse
union all

```

```
select * from rmt15.tpc.dbo.t_warehouse
union all
select * from rmt16.tpc.dbo.t_warehouse
go
```

```
create view district as
select * from /*rmt6.tpc.dbo.*/t_district
union all
select * from rmt1.tpc.dbo.t_district
union all
select * from rmt2.tpc.dbo.t_district
union all
select * from rmt3.tpc.dbo.t_district
union all
select * from rmt4.tpc.dbo.t_district
union all
select * from rmt5.tpc.dbo.t_district
union all
select * from rmt7.tpc.dbo.t_district
union all
select * from rmt8.tpc.dbo.t_district
union all
select * from rmt9.tpc.dbo.t_district
union all
select * from rmt10.tpc.dbo.t_district
union all
select * from rmt11.tpc.dbo.t_district
union all
select * from rmt12.tpc.dbo.t_district
union all
select * from rmt13.tpc.dbo.t_district
union all
select * from rmt14.tpc.dbo.t_district
union all
select * from rmt15.tpc.dbo.t_district
union all
select * from rmt16.tpc.dbo.t_district
go
```

```
create view customer as
select * from /*rmt6.tpc.dbo.*/t_customer
union all
select * from rmt1.tpc.dbo.t_customer
union all
select * from rmt2.tpc.dbo.t_customer
union all
select * from rmt3.tpc.dbo.t_customer
union all
select * from rmt4.tpc.dbo.t_customer
```

```
union all
select * from rmt5.tpc.dbo.t_customer
union all
select * from rmt7.tpc.dbo.t_customer
union all
select * from rmt8.tpc.dbo.t_customer
union all
select * from rmt9.tpc.dbo.t_customer
union all
select * from rmt10.tpc.dbo.t_customer
union all
select * from rmt11.tpc.dbo.t_customer
union all
select * from rmt12.tpc.dbo.t_customer
union all
select * from rmt13.tpc.dbo.t_customer
union all
select * from rmt14.tpc.dbo.t_customer
union all
select * from rmt15.tpc.dbo.t_customer
union all
select * from rmt16.tpc.dbo.t_customer
go
```

```
create view history as
select * from /*rmt6.tpc.dbo.*/t_history
union all
select * from rmt1.tpc.dbo.t_history
union all
select * from rmt2.tpc.dbo.t_history
union all
select * from rmt3.tpc.dbo.t_history
union all
select * from rmt4.tpc.dbo.t_history
union all
select * from rmt5.tpc.dbo.t_history
union all
select * from rmt7.tpc.dbo.t_history
union all
select * from rmt8.tpc.dbo.t_history
union all
select * from rmt9.tpc.dbo.t_history
union all
select * from rmt10.tpc.dbo.t_history
union all
select * from rmt11.tpc.dbo.t_history
union all
select * from rmt12.tpc.dbo.t_history
union all
```

```

select * from rmt13.tpcc.dbo.t_history
union all
select * from rmt14.tpcc.dbo.t_history
union all
select * from rmt15.tpcc.dbo.t_history
union all
select * from rmt16.tpcc.dbo.t_history
go

```

```

create view stock as
select * from /*rmt6.tpcc.dbo.*/t_stock
union all
select * from rmt1.tpcc.dbo.t_stock
union all
select * from rmt2.tpcc.dbo.t_stock
union all
select * from rmt3.tpcc.dbo.t_stock
union all
select * from rmt4.tpcc.dbo.t_stock
union all
select * from rmt5.tpcc.dbo.t_stock
union all
select * from rmt7.tpcc.dbo.t_stock
union all
select * from rmt8.tpcc.dbo.t_stock
union all
select * from rmt9.tpcc.dbo.t_stock
union all
select * from rmt10.tpcc.dbo.t_stock
union all
select * from rmt11.tpcc.dbo.t_stock
union all
select * from rmt12.tpcc.dbo.t_stock
union all
select * from rmt13.tpcc.dbo.t_stock
union all
select * from rmt14.tpcc.dbo.t_stock
union all
select * from rmt15.tpcc.dbo.t_stock
union all
select * from rmt16.tpcc.dbo.t_stock
go

```

```

create view orders as
select * from /*rmt6.tpcc.dbo.*/t_orders
union all
select * from rmt1.tpcc.dbo.t_orders
union all
select * from rmt2.tpcc.dbo.t_orders

```

```

union all
select * from rmt3.tpcc.dbo.t_orders
union all
select * from rmt4.tpcc.dbo.t_orders
union all
select * from rmt5.tpcc.dbo.t_orders
union all
select * from rmt7.tpcc.dbo.t_orders
union all
select * from rmt8.tpcc.dbo.t_orders
union all
select * from rmt9.tpcc.dbo.t_orders
union all
select * from rmt10.tpcc.dbo.t_orders
union all
select * from rmt11.tpcc.dbo.t_orders
union all
select * from rmt12.tpcc.dbo.t_orders
union all
select * from rmt13.tpcc.dbo.t_orders
union all
select * from rmt14.tpcc.dbo.t_orders
union all
select * from rmt15.tpcc.dbo.t_orders
union all
select * from rmt16.tpcc.dbo.t_orders
go

```

```

create view order_line as
select * from /*rmt6.tpcc.dbo.*/t_order_line
union all
select * from rmt1.tpcc.dbo.t_order_line
union all
select * from rmt2.tpcc.dbo.t_order_line
union all
select * from rmt3.tpcc.dbo.t_order_line
union all
select * from rmt4.tpcc.dbo.t_order_line
union all
select * from rmt5.tpcc.dbo.t_order_line
union all
select * from rmt7.tpcc.dbo.t_order_line
union all
select * from rmt8.tpcc.dbo.t_order_line
union all
select * from rmt9.tpcc.dbo.t_order_line
union all
select * from rmt10.tpcc.dbo.t_order_line
union all

```

```

select * from rmt11.tpcc.dbo.t_order_line
union all
select * from rmt12.tpcc.dbo.t_order_line
union all
select * from rmt13.tpcc.dbo.t_order_line
union all
select * from rmt14.tpcc.dbo.t_order_line
union all
select * from rmt15.tpcc.dbo.t_order_line
union all
select * from rmt16.tpcc.dbo.t_order_line
go

create view new_order as
select * from /*rmt6.tpcc.dbo.*/t_new_order
union all
select * from rmt1.tpcc.dbo.t_new_order
union all
select * from rmt2.tpcc.dbo.t_new_order
union all
select * from rmt3.tpcc.dbo.t_new_order
union all
select * from rmt4.tpcc.dbo.t_new_order
union all
select * from rmt5.tpcc.dbo.t_new_order
union all
select * from rmt7.tpcc.dbo.t_new_order
union all
select * from rmt8.tpcc.dbo.t_new_order
union all
select * from rmt9.tpcc.dbo.t_new_order
union all
select * from rmt10.tpcc.dbo.t_new_order
union all
select * from rmt11.tpcc.dbo.t_new_order
union all
select * from rmt12.tpcc.dbo.t_new_order
union all
select * from rmt13.tpcc.dbo.t_new_order
union all
select * from rmt14.tpcc.dbo.t_new_order
union all
select * from rmt15.tpcc.dbo.t_new_order
union all
select * from rmt16.tpcc.dbo.t_new_order
go

create view item as
select * from t_item

```

```

go

add_views7.sql

-- file 10801_to_12600\add_views.sql

set ansi_warnings on
set ansi_nulls on
go

use tpcc
go

-- drop any existing views
if object_id('warehouse') is not null drop view warehouse
if object_id('district') is not null drop view district
if object_id('customer') is not null drop view customer
if object_id('history') is not null drop view history
if object_id('stock') is not null drop view stock
if object_id('orders') is not null drop view orders
if object_id('order_line') is not null drop view order_line
if object_id('new_order') is not null drop view new_order
if object_id('item') is not null drop view item
go

--add views for partition 7
create view warehouse as
select * from /*rmt7.tpcc.dbo.*/t_warehouse
union all
select * from rmt1.tpcc.dbo.t_warehouse
union all
select * from rmt2.tpcc.dbo.t_warehouse
union all
select * from rmt3.tpcc.dbo.t_warehouse
union all
select * from rmt4.tpcc.dbo.t_warehouse
union all
select * from rmt5.tpcc.dbo.t_warehouse
union all
select * from rmt6.tpcc.dbo.t_warehouse
union all
select * from rmt8.tpcc.dbo.t_warehouse
union all
select * from rmt9.tpcc.dbo.t_warehouse
union all
select * from rmt10.tpcc.dbo.t_warehouse
union all
select * from rmt11.tpcc.dbo.t_warehouse
union all
select * from rmt12.tpcc.dbo.t_warehouse
union all
select * from rmt13.tpcc.dbo.t_warehouse
union all
select * from rmt14.tpcc.dbo.t_warehouse
union all
select * from rmt15.tpcc.dbo.t_warehouse
union all
select * from rmt16.tpcc.dbo.t_warehouse
go

create view district as
select * from /*rmt7.tpcc.dbo.*/t_district
union all
select * from rmt1.tpcc.dbo.t_district
union all
select * from rmt2.tpcc.dbo.t_district

```



```

union all
select * from rmt3.tpcc.dbo.t_district
union all
select * from rmt4.tpcc.dbo.t_district
union all
select * from rmt5.tpcc.dbo.t_district
union all
select * from rmt6.tpcc.dbo.t_district
union all
select * from rmt8.tpcc.dbo.t_district
union all
select * from rmt9.tpcc.dbo.t_district
union all
select * from rmt10.tpcc.dbo.t_district
union all
select * from rmt11.tpcc.dbo.t_district
union all
select * from rmt12.tpcc.dbo.t_district
union all
select * from rmt13.tpcc.dbo.t_district
union all
select * from rmt14.tpcc.dbo.t_district
union all
select * from rmt15.tpcc.dbo.t_district
union all
select * from rmt16.tpcc.dbo.t_district
go

```

```

create view customer as
select * from /*rmt7.tpcc.dbo.*/t_customer
union all
select * from rmt1.tpcc.dbo.t_customer
union all
select * from rmt2.tpcc.dbo.t_customer
union all
select * from rmt3.tpcc.dbo.t_customer
union all
select * from rmt4.tpcc.dbo.t_customer
union all
select * from rmt5.tpcc.dbo.t_customer
union all
select * from rmt6.tpcc.dbo.t_customer
union all
select * from rmt8.tpcc.dbo.t_customer
union all
select * from rmt9.tpcc.dbo.t_customer
union all
select * from rmt10.tpcc.dbo.t_customer
union all
select * from rmt11.tpcc.dbo.t_customer
union all
select * from rmt12.tpcc.dbo.t_customer
union all
select * from rmt13.tpcc.dbo.t_customer
union all
select * from rmt14.tpcc.dbo.t_customer
union all
select * from rmt15.tpcc.dbo.t_customer
union all
select * from rmt16.tpcc.dbo.t_customer
go

```

```

create view history as
select * from /*rmt7.tpcc.dbo.*/t_history
union all
select * from rmt1.tpcc.dbo.t_history
union all
select * from rmt2.tpcc.dbo.t_history

```

```

union all
select * from rmt3.tpcc.dbo.t_history
union all
select * from rmt4.tpcc.dbo.t_history
union all
select * from rmt5.tpcc.dbo.t_history
union all
select * from rmt6.tpcc.dbo.t_history
union all
select * from rmt8.tpcc.dbo.t_history
union all
select * from rmt9.tpcc.dbo.t_history
union all
select * from rmt10.tpcc.dbo.t_history
union all
select * from rmt11.tpcc.dbo.t_history
union all
select * from rmt12.tpcc.dbo.t_history
union all
select * from rmt13.tpcc.dbo.t_history
union all
select * from rmt14.tpcc.dbo.t_history
union all
select * from rmt15.tpcc.dbo.t_history
union all
select * from rmt16.tpcc.dbo.t_history
go

```

```

create view stock as
select * from /*rmt7.tpcc.dbo.*/t_stock
union all
select * from rmt1.tpcc.dbo.t_stock
union all
select * from rmt2.tpcc.dbo.t_stock
union all
select * from rmt3.tpcc.dbo.t_stock
union all
select * from rmt4.tpcc.dbo.t_stock
union all
select * from rmt5.tpcc.dbo.t_stock
union all
select * from rmt6.tpcc.dbo.t_stock
union all
select * from rmt8.tpcc.dbo.t_stock
union all
select * from rmt9.tpcc.dbo.t_stock
union all
select * from rmt10.tpcc.dbo.t_stock
union all
select * from rmt11.tpcc.dbo.t_stock
union all
select * from rmt12.tpcc.dbo.t_stock
union all
select * from rmt13.tpcc.dbo.t_stock
union all
select * from rmt14.tpcc.dbo.t_stock
union all
select * from rmt15.tpcc.dbo.t_stock
union all
select * from rmt16.tpcc.dbo.t_stock
go

```

```

create view orders as
select * from /*rmt7.tpcc.dbo.*/t_orders
union all
select * from rmt1.tpcc.dbo.t_orders
union all
select * from rmt2.tpcc.dbo.t_orders

```

```

union all
select * from rmt3.tpcc.dbo.t_orders
union all
select * from rmt4.tpcc.dbo.t_orders
union all
select * from rmt5.tpcc.dbo.t_orders
union all
select * from rmt6.tpcc.dbo.t_orders
union all
select * from rmt8.tpcc.dbo.t_orders
union all
select * from rmt9.tpcc.dbo.t_orders
union all
select * from rmt10.tpcc.dbo.t_orders
union all
select * from rmt11.tpcc.dbo.t_orders
union all
select * from rmt12.tpcc.dbo.t_orders
union all
select * from rmt13.tpcc.dbo.t_orders
union all
select * from rmt14.tpcc.dbo.t_orders
union all
select * from rmt15.tpcc.dbo.t_orders
union all
select * from rmt16.tpcc.dbo.t_orders
go

```

```

create view order_line as
select * from /*rmt7.tpcc.dbo.*/t_order_line
union all
select * from rmt1.tpcc.dbo.t_order_line
union all
select * from rmt2.tpcc.dbo.t_order_line
union all
select * from rmt3.tpcc.dbo.t_order_line
union all
select * from rmt4.tpcc.dbo.t_order_line
union all
select * from rmt5.tpcc.dbo.t_order_line
union all
select * from rmt6.tpcc.dbo.t_order_line
union all
select * from rmt8.tpcc.dbo.t_order_line
union all
select * from rmt9.tpcc.dbo.t_order_line
union all
select * from rmt10.tpcc.dbo.t_order_line
union all
select * from rmt11.tpcc.dbo.t_order_line
union all
select * from rmt12.tpcc.dbo.t_order_line
union all
select * from rmt13.tpcc.dbo.t_order_line
union all
select * from rmt14.tpcc.dbo.t_order_line
union all
select * from rmt15.tpcc.dbo.t_order_line
union all
select * from rmt16.tpcc.dbo.t_order_line
go

```

```

create view new_order as
select * from /*rmt7.tpcc.dbo.*/t_new_order
union all
select * from rmt1.tpcc.dbo.t_new_order
union all
select * from rmt2.tpcc.dbo.t_new_order

```

```

union all
select * from rmt3.tpcc.dbo.t_new_order
union all
select * from rmt4.tpcc.dbo.t_new_order
union all
select * from rmt5.tpcc.dbo.t_new_order
union all
select * from rmt6.tpcc.dbo.t_new_order
union all
select * from rmt8.tpcc.dbo.t_new_order
union all
select * from rmt9.tpcc.dbo.t_new_order
union all
select * from rmt10.tpcc.dbo.t_new_order
union all
select * from rmt11.tpcc.dbo.t_new_order
union all
select * from rmt12.tpcc.dbo.t_new_order
union all
select * from rmt13.tpcc.dbo.t_new_order
union all
select * from rmt14.tpcc.dbo.t_new_order
union all
select * from rmt15.tpcc.dbo.t_new_order
union all
select * from rmt16.tpcc.dbo.t_new_order
go

```

```

create view item as
select * from t_item
go

```

add_views8.sql

```
-- file 12601_to_14400\add_views.sql
```

```

set ansi_warnings on
set ansi_nulls on
go

```

```

use tpcc
go

```

```

-- drop any existing views
if object_id('warehouse') is not null drop view warehouse
if object_id('district') is not null drop view district
if object_id('customer') is not null drop view customer
if object_id('history') is not null drop view history
if object_id('stock') is not null drop view stock
if object_id('orders') is not null drop view orders
if object_id('order_line') is not null drop view order_line
if object_id('new_order') is not null drop view new_order
if object_id('item') is not null drop view item
go

```

```

--add views for partition 8
create view warehouse as
select * from /*rmt8.tpcc.dbo.*/t_warehouse
union all
select * from rmt1.tpcc.dbo.t_warehouse
union all
select * from rmt2.tpcc.dbo.t_warehouse
union all
select * from rmt3.tpcc.dbo.t_warehouse
union all
select * from rmt4.tpcc.dbo.t_warehouse
union all

```

```

select * from rmt5.tpc.dbo.t_warehouse
union all
select * from rmt6.tpc.dbo.t_warehouse
union all
select * from rmt7.tpc.dbo.t_warehouse
union all
select * from rmt9.tpc.dbo.t_warehouse
union all
select * from rmt10.tpc.dbo.t_warehouse
union all
select * from rmt11.tpc.dbo.t_warehouse
union all
select * from rmt12.tpc.dbo.t_warehouse
union all
select * from rmt13.tpc.dbo.t_warehouse
union all
select * from rmt14.tpc.dbo.t_warehouse
union all
select * from rmt15.tpc.dbo.t_warehouse
union all
select * from rmt16.tpc.dbo.t_warehouse
go

```

```

create view district as
select * from /*rmt8.tpc.dbo.*/t_district
union all
select * from rmt1.tpc.dbo.t_district
union all
select * from rmt2.tpc.dbo.t_district
union all
select * from rmt3.tpc.dbo.t_district
union all
select * from rmt4.tpc.dbo.t_district
union all
select * from rmt5.tpc.dbo.t_district
union all
select * from rmt6.tpc.dbo.t_district
union all
select * from rmt7.tpc.dbo.t_district
union all
select * from rmt9.tpc.dbo.t_district
union all
select * from rmt10.tpc.dbo.t_district
union all
select * from rmt11.tpc.dbo.t_district
union all
select * from rmt12.tpc.dbo.t_district
union all
select * from rmt13.tpc.dbo.t_district
union all
select * from rmt14.tpc.dbo.t_district
union all
select * from rmt15.tpc.dbo.t_district
union all
select * from rmt16.tpc.dbo.t_district
go

```

```

create view customer as
select * from /*rmt8.tpc.dbo.*/t_customer
union all
select * from rmt1.tpc.dbo.t_customer
union all
select * from rmt2.tpc.dbo.t_customer
union all
select * from rmt3.tpc.dbo.t_customer
union all
select * from rmt4.tpc.dbo.t_customer
union all

```

```

select * from rmt5.tpc.dbo.t_customer
union all
select * from rmt6.tpc.dbo.t_customer
union all
select * from rmt7.tpc.dbo.t_customer
union all
select * from rmt9.tpc.dbo.t_customer
union all
select * from rmt10.tpc.dbo.t_customer
union all
select * from rmt11.tpc.dbo.t_customer
union all
select * from rmt12.tpc.dbo.t_customer
union all
select * from rmt13.tpc.dbo.t_customer
union all
select * from rmt14.tpc.dbo.t_customer
union all
select * from rmt15.tpc.dbo.t_customer
union all
select * from rmt16.tpc.dbo.t_customer
go

```

```

create view history as
select * from /*rmt8.tpc.dbo.*/t_history
union all
select * from rmt1.tpc.dbo.t_history
union all
select * from rmt2.tpc.dbo.t_history
union all
select * from rmt3.tpc.dbo.t_history
union all
select * from rmt4.tpc.dbo.t_history
union all
select * from rmt5.tpc.dbo.t_history
union all
select * from rmt6.tpc.dbo.t_history
union all
select * from rmt7.tpc.dbo.t_history
union all
select * from rmt9.tpc.dbo.t_history
union all
select * from rmt10.tpc.dbo.t_history
union all
select * from rmt11.tpc.dbo.t_history
union all
select * from rmt12.tpc.dbo.t_history
union all
select * from rmt13.tpc.dbo.t_history
union all
select * from rmt14.tpc.dbo.t_history
union all
select * from rmt15.tpc.dbo.t_history
union all
select * from rmt16.tpc.dbo.t_history
go

```

```

create view stock as
select * from /*rmt8.tpc.dbo.*/t_stock
union all
select * from rmt1.tpc.dbo.t_stock
union all
select * from rmt2.tpc.dbo.t_stock
union all
select * from rmt3.tpc.dbo.t_stock
union all
select * from rmt4.tpc.dbo.t_stock
union all

```

```

select * from rmt5.tpcc.dbo.t_stock
union all
select * from rmt6.tpcc.dbo.t_stock
union all
select * from rmt7.tpcc.dbo.t_stock
union all
select * from rmt9.tpcc.dbo.t_stock
union all
select * from rmt10.tpcc.dbo.t_stock
union all
select * from rmt11.tpcc.dbo.t_stock
union all
select * from rmt12.tpcc.dbo.t_stock
union all
select * from rmt13.tpcc.dbo.t_stock
union all
select * from rmt14.tpcc.dbo.t_stock
union all
select * from rmt15.tpcc.dbo.t_stock
union all
select * from rmt16.tpcc.dbo.t_stock
go

```

```

create view orders as
select * from /*rmt8.tpcc.dbo.*/t_orders
union all
select * from rmt1.tpcc.dbo.t_orders
union all
select * from rmt2.tpcc.dbo.t_orders
union all
select * from rmt3.tpcc.dbo.t_orders
union all
select * from rmt4.tpcc.dbo.t_orders
union all
select * from rmt5.tpcc.dbo.t_orders
union all
select * from rmt6.tpcc.dbo.t_orders
union all
select * from rmt7.tpcc.dbo.t_orders
union all
select * from rmt9.tpcc.dbo.t_orders
union all
select * from rmt10.tpcc.dbo.t_orders
union all
select * from rmt11.tpcc.dbo.t_orders
union all
select * from rmt12.tpcc.dbo.t_orders
union all
select * from rmt13.tpcc.dbo.t_orders
union all
select * from rmt14.tpcc.dbo.t_orders
union all
select * from rmt15.tpcc.dbo.t_orders
union all
select * from rmt16.tpcc.dbo.t_orders
go

```

```

create view order_line as
select * from /*rmt8.tpcc.dbo.*/t_order_line
union all
select * from rmt1.tpcc.dbo.t_order_line
union all
select * from rmt2.tpcc.dbo.t_order_line
union all
select * from rmt3.tpcc.dbo.t_order_line
union all
select * from rmt4.tpcc.dbo.t_order_line
union all

```

```

select * from rmt5.tpcc.dbo.t_order_line
union all
select * from rmt6.tpcc.dbo.t_order_line
union all
select * from rmt7.tpcc.dbo.t_order_line
union all
select * from rmt9.tpcc.dbo.t_order_line
union all
select * from rmt10.tpcc.dbo.t_order_line
union all
select * from rmt11.tpcc.dbo.t_order_line
union all
select * from rmt12.tpcc.dbo.t_order_line
union all
select * from rmt13.tpcc.dbo.t_order_line
union all
select * from rmt14.tpcc.dbo.t_order_line
union all
select * from rmt15.tpcc.dbo.t_order_line
union all
select * from rmt16.tpcc.dbo.t_order_line
go

```

```

create view new_order as
select * from /*rmt8.tpcc.dbo.*/t_new_order
union all
select * from rmt1.tpcc.dbo.t_new_order
union all
select * from rmt2.tpcc.dbo.t_new_order
union all
select * from rmt3.tpcc.dbo.t_new_order
union all
select * from rmt4.tpcc.dbo.t_new_order
union all
select * from rmt5.tpcc.dbo.t_new_order
union all
select * from rmt6.tpcc.dbo.t_new_order
union all
select * from rmt7.tpcc.dbo.t_new_order
union all
select * from rmt9.tpcc.dbo.t_new_order
union all
select * from rmt10.tpcc.dbo.t_new_order
union all
select * from rmt11.tpcc.dbo.t_new_order
union all
select * from rmt12.tpcc.dbo.t_new_order
union all
select * from rmt13.tpcc.dbo.t_new_order
union all
select * from rmt14.tpcc.dbo.t_new_order
union all
select * from rmt15.tpcc.dbo.t_new_order
union all
select * from rmt16.tpcc.dbo.t_new_order
go

```

```

create view item as
select * from t_item
go

```

add_views9.sql

```
-- file 14401_to_16200\add_views.sql
```

```
set ansi_warnings on
```

```

set ansi_nulls on
go

use tpcc
go

-- drop any existing views
if object_id('warehouse') is not null drop view warehouse
if object_id('district') is not null drop view district
if object_id('customer') is not null drop view customer
if object_id('history') is not null drop view history
if object_id('stock') is not null drop view stock
if object_id('orders') is not null drop view orders
if object_id('order_line') is not null drop view order_line
if object_id('new_order') is not null drop view new_order
if object_id('item') is not null drop view item
go

```

```

--add views for partition 9
create view warehouse as
select * from /*rmt9,tpcc.dbo.*/t_warehouse
union all
select * from rmt1.tpcc.dbo.t_warehouse
union all
select * from rmt2.tpcc.dbo.t_warehouse
union all
select * from rmt3.tpcc.dbo.t_warehouse
union all
select * from rmt4.tpcc.dbo.t_warehouse
union all
select * from rmt5.tpcc.dbo.t_warehouse
union all
select * from rmt6.tpcc.dbo.t_warehouse
union all
select * from rmt7.tpcc.dbo.t_warehouse
union all
select * from rmt8.tpcc.dbo.t_warehouse
union all
select * from rmt10.tpcc.dbo.t_warehouse
union all
select * from rmt11.tpcc.dbo.t_warehouse
union all
select * from rmt12.tpcc.dbo.t_warehouse
union all
select * from rmt13.tpcc.dbo.t_warehouse
union all
select * from rmt14.tpcc.dbo.t_warehouse
union all
select * from rmt15.tpcc.dbo.t_warehouse
union all
select * from rmt16.tpcc.dbo.t_warehouse
go

```

```

create view district as
select * from /*rmt9,tpcc.dbo.*/t_district
union all
select * from rmt1.tpcc.dbo.t_district
union all
select * from rmt2.tpcc.dbo.t_district
union all
select * from rmt3.tpcc.dbo.t_district
union all
select * from rmt4.tpcc.dbo.t_district
union all
select * from rmt5.tpcc.dbo.t_district
union all
select * from rmt6.tpcc.dbo.t_district
union all

```

```

select * from rmt7.tpcc.dbo.t_district
union all
select * from rmt8.tpcc.dbo.t_district
union all
select * from rmt10.tpcc.dbo.t_district
union all
select * from rmt11.tpcc.dbo.t_district
union all
select * from rmt12.tpcc.dbo.t_district
union all
select * from rmt13.tpcc.dbo.t_district
union all
select * from rmt14.tpcc.dbo.t_district
union all
select * from rmt15.tpcc.dbo.t_district
union all
select * from rmt16.tpcc.dbo.t_district
go

```

```

create view customer as
select * from /*rmt9,tpcc.dbo.*/t_customer
union all
select * from rmt1.tpcc.dbo.t_customer
union all
select * from rmt2.tpcc.dbo.t_customer
union all
select * from rmt3.tpcc.dbo.t_customer
union all
select * from rmt4.tpcc.dbo.t_customer
union all
select * from rmt5.tpcc.dbo.t_customer
union all
select * from rmt6.tpcc.dbo.t_customer
union all
select * from rmt7.tpcc.dbo.t_customer
union all
select * from rmt8.tpcc.dbo.t_customer
union all
select * from rmt10.tpcc.dbo.t_customer
union all
select * from rmt11.tpcc.dbo.t_customer
union all
select * from rmt12.tpcc.dbo.t_customer
union all
select * from rmt13.tpcc.dbo.t_customer
union all
select * from rmt14.tpcc.dbo.t_customer
union all
select * from rmt15.tpcc.dbo.t_customer
union all
select * from rmt16.tpcc.dbo.t_customer
go

```

```

create view history as
select * from /*rmt9,tpcc.dbo.*/t_history
union all
select * from rmt1.tpcc.dbo.t_history
union all
select * from rmt2.tpcc.dbo.t_history
union all
select * from rmt3.tpcc.dbo.t_history
union all
select * from rmt4.tpcc.dbo.t_history
union all
select * from rmt5.tpcc.dbo.t_history
union all
select * from rmt6.tpcc.dbo.t_history
union all

```

```

select * from rmt7.tpcc.dbo.t_history
union all
select * from rmt8.tpcc.dbo.t_history
union all
select * from rmt10.tpcc.dbo.t_history
union all
select * from rmt11.tpcc.dbo.t_history
union all
select * from rmt12.tpcc.dbo.t_history
union all
select * from rmt13.tpcc.dbo.t_history
union all
select * from rmt14.tpcc.dbo.t_history
union all
select * from rmt15.tpcc.dbo.t_history
union all
select * from rmt16.tpcc.dbo.t_history
go

```

```

create view stock as
select * from /*rmt9.tpcc.dbo.*/t_stock
union all
select * from rmt1.tpcc.dbo.t_stock
union all
select * from rmt2.tpcc.dbo.t_stock
union all
select * from rmt3.tpcc.dbo.t_stock
union all
select * from rmt4.tpcc.dbo.t_stock
union all
select * from rmt5.tpcc.dbo.t_stock
union all
select * from rmt6.tpcc.dbo.t_stock
union all
select * from rmt7.tpcc.dbo.t_stock
union all
select * from rmt8.tpcc.dbo.t_stock
union all
select * from rmt10.tpcc.dbo.t_stock
union all
select * from rmt11.tpcc.dbo.t_stock
union all
select * from rmt12.tpcc.dbo.t_stock
union all
select * from rmt13.tpcc.dbo.t_stock
union all
select * from rmt14.tpcc.dbo.t_stock
union all
select * from rmt15.tpcc.dbo.t_stock
union all
select * from rmt16.tpcc.dbo.t_stock
go

```

```

create view orders as
select * from /*rmt9.tpcc.dbo.*/t_orders
union all
select * from rmt1.tpcc.dbo.t_orders
union all
select * from rmt2.tpcc.dbo.t_orders
union all
select * from rmt3.tpcc.dbo.t_orders
union all
select * from rmt4.tpcc.dbo.t_orders
union all
select * from rmt5.tpcc.dbo.t_orders
union all
select * from rmt6.tpcc.dbo.t_orders
union all

```

```

select * from rmt7.tpcc.dbo.t_orders
union all
select * from rmt8.tpcc.dbo.t_orders
union all
select * from rmt10.tpcc.dbo.t_orders
union all
select * from rmt11.tpcc.dbo.t_orders
union all
select * from rmt12.tpcc.dbo.t_orders
union all
select * from rmt13.tpcc.dbo.t_orders
union all
select * from rmt14.tpcc.dbo.t_orders
union all
select * from rmt15.tpcc.dbo.t_orders
union all
select * from rmt16.tpcc.dbo.t_orders
go

```

```

create view order_line as
select * from /*rmt9.tpcc.dbo.*/t_order_line
union all
select * from rmt1.tpcc.dbo.t_order_line
union all
select * from rmt2.tpcc.dbo.t_order_line
union all
select * from rmt3.tpcc.dbo.t_order_line
union all
select * from rmt4.tpcc.dbo.t_order_line
union all
select * from rmt5.tpcc.dbo.t_order_line
union all
select * from rmt6.tpcc.dbo.t_order_line
union all
select * from rmt7.tpcc.dbo.t_order_line
union all
select * from rmt8.tpcc.dbo.t_order_line
union all
select * from rmt10.tpcc.dbo.t_order_line
union all
select * from rmt11.tpcc.dbo.t_order_line
union all
select * from rmt12.tpcc.dbo.t_order_line
union all
select * from rmt13.tpcc.dbo.t_order_line
union all
select * from rmt14.tpcc.dbo.t_order_line
union all
select * from rmt15.tpcc.dbo.t_order_line
union all
select * from rmt16.tpcc.dbo.t_order_line
go

```

```

create view new_order as
select * from /*rmt9.tpcc.dbo.*/t_new_order
union all
select * from rmt1.tpcc.dbo.t_new_order
union all
select * from rmt2.tpcc.dbo.t_new_order
union all
select * from rmt3.tpcc.dbo.t_new_order
union all
select * from rmt4.tpcc.dbo.t_new_order
union all
select * from rmt5.tpcc.dbo.t_new_order
union all
select * from rmt6.tpcc.dbo.t_new_order
union all

```

```

select * from rmt7.tpcc.dbo.t_new_order
union all
select * from rmt8.tpcc.dbo.t_new_order
union all
select * from rmt10.tpcc.dbo.t_new_order
union all
select * from rmt11.tpcc.dbo.t_new_order
union all
select * from rmt12.tpcc.dbo.t_new_order
union all
select * from rmt13.tpcc.dbo.t_new_order
union all
select * from rmt14.tpcc.dbo.t_new_order
union all
select * from rmt15.tpcc.dbo.t_new_order
union all
select * from rmt16.tpcc.dbo.t_new_order
go

```

```

create view item as
select * from t_item
go

```

add_views10.sql

```
-- file 16201_to_18000\add_views.sql
```

```

set ansi_warnings on
set ansi_nulls on
go

```

```

use tpcc
go

```

```

-- drop any existing views
if object_id('warehouse') is not null drop view warehouse
if object_id('district') is not null drop view district
if object_id('customer') is not null drop view customer
if object_id('history') is not null drop view history
if object_id('stock') is not null drop view stock
if object_id('orders') is not null drop view orders
if object_id('order_line') is not null drop view order_line
if object_id('new_order') is not null drop view new_order
if object_id('item') is not null drop view item
go

```

```

--add views for partition 10
create view warehouse as
select * from /*rmt10.tpcc.dbo.*/t_warehouse
union all
select * from rmt1.tpcc.dbo.t_warehouse
union all
select * from rmt2.tpcc.dbo.t_warehouse
union all
select * from rmt3.tpcc.dbo.t_warehouse
union all
select * from rmt4.tpcc.dbo.t_warehouse
union all
select * from rmt5.tpcc.dbo.t_warehouse
union all
select * from rmt6.tpcc.dbo.t_warehouse
union all
select * from rmt7.tpcc.dbo.t_warehouse
union all
select * from rmt8.tpcc.dbo.t_warehouse
union all

```

```

select * from rmt9.tpcc.dbo.t_warehouse
union all
select * from rmt11.tpcc.dbo.t_warehouse
union all
select * from rmt12.tpcc.dbo.t_warehouse
union all
select * from rmt13.tpcc.dbo.t_warehouse
union all
select * from rmt14.tpcc.dbo.t_warehouse
union all
select * from rmt15.tpcc.dbo.t_warehouse
union all
select * from rmt16.tpcc.dbo.t_warehouse
go

```

```

create view district as
select * from /*rmt10.tpcc.dbo.*/t_district
union all
select * from rmt1.tpcc.dbo.t_district
union all
select * from rmt2.tpcc.dbo.t_district
union all
select * from rmt3.tpcc.dbo.t_district
union all
select * from rmt4.tpcc.dbo.t_district
union all
select * from rmt5.tpcc.dbo.t_district
union all
select * from rmt6.tpcc.dbo.t_district
union all
select * from rmt7.tpcc.dbo.t_district
union all
select * from rmt8.tpcc.dbo.t_district
union all
select * from rmt9.tpcc.dbo.t_district
union all
select * from rmt11.tpcc.dbo.t_district
union all
select * from rmt12.tpcc.dbo.t_district
union all
select * from rmt13.tpcc.dbo.t_district
union all
select * from rmt14.tpcc.dbo.t_district
union all
select * from rmt15.tpcc.dbo.t_district
union all
select * from rmt16.tpcc.dbo.t_district
go

```

```

create view customer as
select * from /*rmt10.tpcc.dbo.*/t_customer
union all
select * from rmt1.tpcc.dbo.t_customer
union all
select * from rmt2.tpcc.dbo.t_customer
union all
select * from rmt3.tpcc.dbo.t_customer
union all
select * from rmt4.tpcc.dbo.t_customer
union all
select * from rmt5.tpcc.dbo.t_customer
union all
select * from rmt6.tpcc.dbo.t_customer
union all
select * from rmt7.tpcc.dbo.t_customer
union all
select * from rmt8.tpcc.dbo.t_customer
union all

```

```

select * from rmt9.tpc.dbo.t_customer
union all
select * from rmt11.tpc.dbo.t_customer
union all
select * from rmt12.tpc.dbo.t_customer
union all
select * from rmt13.tpc.dbo.t_customer
union all
select * from rmt14.tpc.dbo.t_customer
union all
select * from rmt15.tpc.dbo.t_customer
union all
select * from rmt16.tpc.dbo.t_customer
go

```

```

create view history as
select * from /*rmt10.tpc.dbo.*/t_history
union all
select * from rmt1.tpc.dbo.t_history
union all
select * from rmt2.tpc.dbo.t_history
union all
select * from rmt3.tpc.dbo.t_history
union all
select * from rmt4.tpc.dbo.t_history
union all
select * from rmt5.tpc.dbo.t_history
union all
select * from rmt6.tpc.dbo.t_history
union all
select * from rmt7.tpc.dbo.t_history
union all
select * from rmt8.tpc.dbo.t_history
union all
select * from rmt9.tpc.dbo.t_history
union all
select * from rmt11.tpc.dbo.t_history
union all
select * from rmt12.tpc.dbo.t_history
union all
select * from rmt13.tpc.dbo.t_history
union all
select * from rmt14.tpc.dbo.t_history
union all
select * from rmt15.tpc.dbo.t_history
union all
select * from rmt16.tpc.dbo.t_history
go

```

```

create view stock as
select * from /*rmt10.tpc.dbo.*/t_stock
union all
select * from rmt1.tpc.dbo.t_stock
union all
select * from rmt2.tpc.dbo.t_stock
union all
select * from rmt3.tpc.dbo.t_stock
union all
select * from rmt4.tpc.dbo.t_stock
union all
select * from rmt5.tpc.dbo.t_stock
union all
select * from rmt6.tpc.dbo.t_stock
union all
select * from rmt7.tpc.dbo.t_stock
union all
select * from rmt8.tpc.dbo.t_stock
union all

```

```

select * from rmt9.tpc.dbo.t_stock
union all
select * from rmt11.tpc.dbo.t_stock
union all
select * from rmt12.tpc.dbo.t_stock
union all
select * from rmt13.tpc.dbo.t_stock
union all
select * from rmt14.tpc.dbo.t_stock
union all
select * from rmt15.tpc.dbo.t_stock
union all
select * from rmt16.tpc.dbo.t_stock
go

```

```

create view orders as
select * from /*rmt10.tpc.dbo.*/t_orders
union all
select * from rmt1.tpc.dbo.t_orders
union all
select * from rmt2.tpc.dbo.t_orders
union all
select * from rmt3.tpc.dbo.t_orders
union all
select * from rmt4.tpc.dbo.t_orders
union all
select * from rmt5.tpc.dbo.t_orders
union all
select * from rmt6.tpc.dbo.t_orders
union all
select * from rmt7.tpc.dbo.t_orders
union all
select * from rmt8.tpc.dbo.t_orders
union all
select * from rmt9.tpc.dbo.t_orders
union all
select * from rmt11.tpc.dbo.t_orders
union all
select * from rmt12.tpc.dbo.t_orders
union all
select * from rmt13.tpc.dbo.t_orders
union all
select * from rmt14.tpc.dbo.t_orders
union all
select * from rmt15.tpc.dbo.t_orders
union all
select * from rmt16.tpc.dbo.t_orders
go

```

```

create view order_line as
select * from /*rmt10.tpc.dbo.*/t_order_line
union all
select * from rmt1.tpc.dbo.t_order_line
union all
select * from rmt2.tpc.dbo.t_order_line
union all
select * from rmt3.tpc.dbo.t_order_line
union all
select * from rmt4.tpc.dbo.t_order_line
union all
select * from rmt5.tpc.dbo.t_order_line
union all
select * from rmt6.tpc.dbo.t_order_line
union all
select * from rmt7.tpc.dbo.t_order_line
union all
select * from rmt8.tpc.dbo.t_order_line
union all

```



```

select * from rmt9.tpcc.dbo.t_order_line
union all
select * from rmt11.tpcc.dbo.t_order_line
union all
select * from rmt12.tpcc.dbo.t_order_line
union all
select * from rmt13.tpcc.dbo.t_order_line
union all
select * from rmt14.tpcc.dbo.t_order_line
union all
select * from rmt15.tpcc.dbo.t_order_line
union all
select * from rmt16.tpcc.dbo.t_order_line
go

```

```

create view new_order as
select * from /*rmt10.tpcc.dbo.*/t_new_order
union all
select * from rmt1.tpcc.dbo.t_new_order
union all
select * from rmt2.tpcc.dbo.t_new_order
union all
select * from rmt3.tpcc.dbo.t_new_order
union all
select * from rmt4.tpcc.dbo.t_new_order
union all
select * from rmt5.tpcc.dbo.t_new_order
union all
select * from rmt6.tpcc.dbo.t_new_order
union all
select * from rmt7.tpcc.dbo.t_new_order
union all
select * from rmt8.tpcc.dbo.t_new_order
union all
select * from rmt9.tpcc.dbo.t_new_order
union all
select * from rmt11.tpcc.dbo.t_new_order
union all
select * from rmt12.tpcc.dbo.t_new_order
union all
select * from rmt13.tpcc.dbo.t_new_order
union all
select * from rmt14.tpcc.dbo.t_new_order
union all
select * from rmt15.tpcc.dbo.t_new_order
union all
select * from rmt16.tpcc.dbo.t_new_order
go

```

```

create view item as
select * from t_item
go

```

add_views11.sql

```
-- file 18001_to_19800\add_views.sql
```

```

set ansi_warnings on
set ansi_nulls on
go

```

```

use tpcc
go

```

```

-- drop any existing views
if object_id('warehouse') is not null drop view warehouse

```

```

if object_id('district') is not null drop view district
if object_id('customer') is not null drop view customer
if object_id('history') is not null drop view history
if object_id('stock') is not null drop view stock
if object_id('orders') is not null drop view orders
if object_id('order_line') is not null drop view order_line
if object_id('new_order') is not null drop view new_order
if object_id('item') is not null drop view item
go

```

```

--add views for partition 11
create view warehouse as
select * from /*rmt11.tpcc.dbo.*/t_warehouse
union all
select * from rmt1.tpcc.dbo.t_warehouse
union all
select * from rmt2.tpcc.dbo.t_warehouse
union all
select * from rmt3.tpcc.dbo.t_warehouse
union all
select * from rmt4.tpcc.dbo.t_warehouse
union all
select * from rmt5.tpcc.dbo.t_warehouse
union all
select * from rmt6.tpcc.dbo.t_warehouse
union all
select * from rmt7.tpcc.dbo.t_warehouse
union all
select * from rmt8.tpcc.dbo.t_warehouse
union all
select * from rmt9.tpcc.dbo.t_warehouse
union all
select * from rmt10.tpcc.dbo.t_warehouse
union all
select * from rmt12.tpcc.dbo.t_warehouse
union all
select * from rmt13.tpcc.dbo.t_warehouse
union all
select * from rmt14.tpcc.dbo.t_warehouse
union all
select * from rmt15.tpcc.dbo.t_warehouse
union all
select * from rmt16.tpcc.dbo.t_warehouse
go

```

```

create view district as
select * from /*rmt11.tpcc.dbo.*/t_district
union all
select * from rmt1.tpcc.dbo.t_district
union all
select * from rmt2.tpcc.dbo.t_district
union all
select * from rmt3.tpcc.dbo.t_district
union all
select * from rmt4.tpcc.dbo.t_district
union all
select * from rmt5.tpcc.dbo.t_district
union all
select * from rmt6.tpcc.dbo.t_district
union all
select * from rmt7.tpcc.dbo.t_district
union all
select * from rmt8.tpcc.dbo.t_district
union all
select * from rmt9.tpcc.dbo.t_district
union all
select * from rmt10.tpcc.dbo.t_district
union all

```

```

select * from rmt12.tpcc.dbo.t_district
union all
select * from rmt13.tpcc.dbo.t_district
union all
select * from rmt14.tpcc.dbo.t_district
union all
select * from rmt15.tpcc.dbo.t_district
union all
select * from rmt16.tpcc.dbo.t_district
go

```

```

create view customer as
select * from /*rmt11.tpcc.dbo.*/t_customer
union all
select * from rmt1.tpcc.dbo.t_customer
union all
select * from rmt2.tpcc.dbo.t_customer
union all
select * from rmt3.tpcc.dbo.t_customer
union all
select * from rmt4.tpcc.dbo.t_customer
union all
select * from rmt5.tpcc.dbo.t_customer
union all
select * from rmt6.tpcc.dbo.t_customer
union all
select * from rmt7.tpcc.dbo.t_customer
union all
select * from rmt8.tpcc.dbo.t_customer
union all
select * from rmt9.tpcc.dbo.t_customer
union all
select * from rmt10.tpcc.dbo.t_customer
union all
select * from rmt12.tpcc.dbo.t_customer
union all
select * from rmt13.tpcc.dbo.t_customer
union all
select * from rmt14.tpcc.dbo.t_customer
union all
select * from rmt15.tpcc.dbo.t_customer
union all
select * from rmt16.tpcc.dbo.t_customer
go

```

```

create view history as
select * from /*rmt11.tpcc.dbo.*/t_history
union all
select * from rmt1.tpcc.dbo.t_history
union all
select * from rmt2.tpcc.dbo.t_history
union all
select * from rmt3.tpcc.dbo.t_history
union all
select * from rmt4.tpcc.dbo.t_history
union all
select * from rmt5.tpcc.dbo.t_history
union all
select * from rmt6.tpcc.dbo.t_history
union all
select * from rmt7.tpcc.dbo.t_history
union all
select * from rmt8.tpcc.dbo.t_history
union all
select * from rmt9.tpcc.dbo.t_history
union all
select * from rmt10.tpcc.dbo.t_history
union all

```

```

select * from rmt12.tpcc.dbo.t_history
union all
select * from rmt13.tpcc.dbo.t_history
union all
select * from rmt14.tpcc.dbo.t_history
union all
select * from rmt15.tpcc.dbo.t_history
union all
select * from rmt16.tpcc.dbo.t_history
go

```

```

create view stock as
select * from /*rmt11.tpcc.dbo.*/t_stock
union all
select * from rmt1.tpcc.dbo.t_stock
union all
select * from rmt2.tpcc.dbo.t_stock
union all
select * from rmt3.tpcc.dbo.t_stock
union all
select * from rmt4.tpcc.dbo.t_stock
union all
select * from rmt5.tpcc.dbo.t_stock
union all
select * from rmt6.tpcc.dbo.t_stock
union all
select * from rmt7.tpcc.dbo.t_stock
union all
select * from rmt8.tpcc.dbo.t_stock
union all
select * from rmt9.tpcc.dbo.t_stock
union all
select * from rmt10.tpcc.dbo.t_stock
union all
select * from rmt12.tpcc.dbo.t_stock
union all
select * from rmt13.tpcc.dbo.t_stock
union all
select * from rmt14.tpcc.dbo.t_stock
union all
select * from rmt15.tpcc.dbo.t_stock
union all
select * from rmt16.tpcc.dbo.t_stock
go

```

```

create view orders as
select * from /*rmt11.tpcc.dbo.*/t_orders
union all
select * from rmt1.tpcc.dbo.t_orders
union all
select * from rmt2.tpcc.dbo.t_orders
union all
select * from rmt3.tpcc.dbo.t_orders
union all
select * from rmt4.tpcc.dbo.t_orders
union all
select * from rmt5.tpcc.dbo.t_orders
union all
select * from rmt6.tpcc.dbo.t_orders
union all
select * from rmt7.tpcc.dbo.t_orders
union all
select * from rmt8.tpcc.dbo.t_orders
union all
select * from rmt9.tpcc.dbo.t_orders
union all
select * from rmt10.tpcc.dbo.t_orders
union all

```

```

select * from rmt12.tpcc.dbo.t_orders
union all
select * from rmt13.tpcc.dbo.t_orders
union all
select * from rmt14.tpcc.dbo.t_orders
union all
select * from rmt15.tpcc.dbo.t_orders
union all
select * from rmt16.tpcc.dbo.t_orders
go

```

```

create view order_line as
select * from /*rmt11.tpcc.dbo.*/t_order_line
union all
select * from rmt1.tpcc.dbo.t_order_line
union all
select * from rmt2.tpcc.dbo.t_order_line
union all
select * from rmt3.tpcc.dbo.t_order_line
union all
select * from rmt4.tpcc.dbo.t_order_line
union all
select * from rmt5.tpcc.dbo.t_order_line
union all
select * from rmt6.tpcc.dbo.t_order_line
union all
select * from rmt7.tpcc.dbo.t_order_line
union all
select * from rmt8.tpcc.dbo.t_order_line
union all
select * from rmt9.tpcc.dbo.t_order_line
union all
select * from rmt10.tpcc.dbo.t_order_line
union all
select * from rmt12.tpcc.dbo.t_order_line
union all
select * from rmt13.tpcc.dbo.t_order_line
union all
select * from rmt14.tpcc.dbo.t_order_line
union all
select * from rmt15.tpcc.dbo.t_order_line
union all
select * from rmt16.tpcc.dbo.t_order_line
go

```

```

create view new_order as
select * from /*rmt11.tpcc.dbo.*/t_new_order
union all
select * from rmt1.tpcc.dbo.t_new_order
union all
select * from rmt2.tpcc.dbo.t_new_order
union all
select * from rmt3.tpcc.dbo.t_new_order
union all
select * from rmt4.tpcc.dbo.t_new_order
union all
select * from rmt5.tpcc.dbo.t_new_order
union all
select * from rmt6.tpcc.dbo.t_new_order
union all
select * from rmt7.tpcc.dbo.t_new_order
union all
select * from rmt8.tpcc.dbo.t_new_order
union all
select * from rmt9.tpcc.dbo.t_new_order
union all
select * from rmt10.tpcc.dbo.t_new_order
union all

```

```

select * from rmt12.tpcc.dbo.t_new_order
union all
select * from rmt13.tpcc.dbo.t_new_order
union all
select * from rmt14.tpcc.dbo.t_new_order
union all
select * from rmt15.tpcc.dbo.t_new_order
union all
select * from rmt16.tpcc.dbo.t_new_order
go

```

```

create view item as
select * from t_item
go

```

add_views12.sql

```
-- file 19801_to_21600\add_views.sql
```

```

set ansi_warnings on
set ansi_nulls on
go

```

```

use tpcc
go

```

```

-- drop any existing views
if object_id('warehouse') is not null drop view warehouse
if object_id('district') is not null drop view district
if object_id('customer') is not null drop view customer
if object_id('history') is not null drop view history
if object_id('stock') is not null drop view stock
if object_id('orders') is not null drop view orders
if object_id('order_line') is not null drop view order_line
if object_id('new_order') is not null drop view new_order
if object_id('item') is not null drop view item
go

```

```

--add views for partition 12
create view warehouse as
select * from /*rmt12.tpcc.dbo.*/t_warehouse
union all
select * from rmt1.tpcc.dbo.t_warehouse
union all
select * from rmt2.tpcc.dbo.t_warehouse
union all
select * from rmt3.tpcc.dbo.t_warehouse
union all
select * from rmt4.tpcc.dbo.t_warehouse
union all
select * from rmt5.tpcc.dbo.t_warehouse
union all
select * from rmt6.tpcc.dbo.t_warehouse
union all
select * from rmt7.tpcc.dbo.t_warehouse
union all
select * from rmt8.tpcc.dbo.t_warehouse
union all
select * from rmt9.tpcc.dbo.t_warehouse
union all
select * from rmt10.tpcc.dbo.t_warehouse
union all
select * from rmt11.tpcc.dbo.t_warehouse
union all
select * from rmt13.tpcc.dbo.t_warehouse
union all

```

```
select * from rmt14.tpcc.dbo.t_warehouse
union all
select * from rmt15.tpcc.dbo.t_warehouse
union all
select * from rmt16.tpcc.dbo.t_warehouse
go
```

```
create view district as
select * from /*rmt12.tpcc.dbo.*/t_district
union all
select * from rmt1.tpcc.dbo.t_district
union all
select * from rmt2.tpcc.dbo.t_district
union all
select * from rmt3.tpcc.dbo.t_district
union all
select * from rmt4.tpcc.dbo.t_district
union all
select * from rmt5.tpcc.dbo.t_district
union all
select * from rmt6.tpcc.dbo.t_district
union all
select * from rmt7.tpcc.dbo.t_district
union all
select * from rmt8.tpcc.dbo.t_district
union all
select * from rmt9.tpcc.dbo.t_district
union all
select * from rmt10.tpcc.dbo.t_district
union all
select * from rmt11.tpcc.dbo.t_district
union all
select * from rmt13.tpcc.dbo.t_district
union all
select * from rmt14.tpcc.dbo.t_district
union all
select * from rmt15.tpcc.dbo.t_district
union all
select * from rmt16.tpcc.dbo.t_district
go
```

```
create view customer as
select * from /*rmt12.tpcc.dbo.*/t_customer
union all
select * from rmt1.tpcc.dbo.t_customer
union all
select * from rmt2.tpcc.dbo.t_customer
union all
select * from rmt3.tpcc.dbo.t_customer
union all
select * from rmt4.tpcc.dbo.t_customer
union all
select * from rmt5.tpcc.dbo.t_customer
union all
select * from rmt6.tpcc.dbo.t_customer
union all
select * from rmt7.tpcc.dbo.t_customer
union all
select * from rmt8.tpcc.dbo.t_customer
union all
select * from rmt9.tpcc.dbo.t_customer
union all
select * from rmt10.tpcc.dbo.t_customer
union all
select * from rmt11.tpcc.dbo.t_customer
union all
select * from rmt13.tpcc.dbo.t_customer
union all
```

```
select * from rmt14.tpcc.dbo.t_customer
union all
select * from rmt15.tpcc.dbo.t_customer
union all
select * from rmt16.tpcc.dbo.t_customer
go
```

```
create view history as
select * from /*rmt12.tpcc.dbo.*/t_history
union all
select * from rmt1.tpcc.dbo.t_history
union all
select * from rmt2.tpcc.dbo.t_history
union all
select * from rmt3.tpcc.dbo.t_history
union all
select * from rmt4.tpcc.dbo.t_history
union all
select * from rmt5.tpcc.dbo.t_history
union all
select * from rmt6.tpcc.dbo.t_history
union all
select * from rmt7.tpcc.dbo.t_history
union all
select * from rmt8.tpcc.dbo.t_history
union all
select * from rmt9.tpcc.dbo.t_history
union all
select * from rmt10.tpcc.dbo.t_history
union all
select * from rmt11.tpcc.dbo.t_history
union all
select * from rmt13.tpcc.dbo.t_history
union all
select * from rmt14.tpcc.dbo.t_history
union all
select * from rmt15.tpcc.dbo.t_history
union all
select * from rmt16.tpcc.dbo.t_history
go
```

```
create view stock as
select * from /*rmt12.tpcc.dbo.*/t_stock
union all
select * from rmt1.tpcc.dbo.t_stock
union all
select * from rmt2.tpcc.dbo.t_stock
union all
select * from rmt3.tpcc.dbo.t_stock
union all
select * from rmt4.tpcc.dbo.t_stock
union all
select * from rmt5.tpcc.dbo.t_stock
union all
select * from rmt6.tpcc.dbo.t_stock
union all
select * from rmt7.tpcc.dbo.t_stock
union all
select * from rmt8.tpcc.dbo.t_stock
union all
select * from rmt9.tpcc.dbo.t_stock
union all
select * from rmt10.tpcc.dbo.t_stock
union all
select * from rmt11.tpcc.dbo.t_stock
union all
select * from rmt13.tpcc.dbo.t_stock
union all
```

```

select * from rmt14.tpcc.dbo.t_stock
union all
select * from rmt15.tpcc.dbo.t_stock
union all
select * from rmt16.tpcc.dbo.t_stock
go

```

```

create view orders as
select * from /*rmt12.tpcc.dbo.*/t_orders
union all
select * from rmt1.tpcc.dbo.t_orders
union all
select * from rmt2.tpcc.dbo.t_orders
union all
select * from rmt3.tpcc.dbo.t_orders
union all
select * from rmt4.tpcc.dbo.t_orders
union all
select * from rmt5.tpcc.dbo.t_orders
union all
select * from rmt6.tpcc.dbo.t_orders
union all
select * from rmt7.tpcc.dbo.t_orders
union all
select * from rmt8.tpcc.dbo.t_orders
union all
select * from rmt9.tpcc.dbo.t_orders
union all
select * from rmt10.tpcc.dbo.t_orders
union all
select * from rmt11.tpcc.dbo.t_orders
union all
select * from rmt13.tpcc.dbo.t_orders
union all
select * from rmt14.tpcc.dbo.t_orders
union all
select * from rmt15.tpcc.dbo.t_orders
union all
select * from rmt16.tpcc.dbo.t_orders
go

```

```

create view order_line as
select * from /*rmt12.tpcc.dbo.*/t_order_line
union all
select * from rmt1.tpcc.dbo.t_order_line
union all
select * from rmt2.tpcc.dbo.t_order_line
union all
select * from rmt3.tpcc.dbo.t_order_line
union all
select * from rmt4.tpcc.dbo.t_order_line
union all
select * from rmt5.tpcc.dbo.t_order_line
union all
select * from rmt6.tpcc.dbo.t_order_line
union all
select * from rmt7.tpcc.dbo.t_order_line
union all
select * from rmt8.tpcc.dbo.t_order_line
union all
select * from rmt9.tpcc.dbo.t_order_line
union all
select * from rmt10.tpcc.dbo.t_order_line
union all
select * from rmt11.tpcc.dbo.t_order_line
union all
select * from rmt13.tpcc.dbo.t_order_line
union all

```

```

select * from rmt14.tpcc.dbo.t_order_line
union all
select * from rmt15.tpcc.dbo.t_order_line
union all
select * from rmt16.tpcc.dbo.t_order_line
go

```

```

create view new_order as
select * from /*rmt12.tpcc.dbo.*/t_new_order
union all
select * from rmt1.tpcc.dbo.t_new_order
union all
select * from rmt2.tpcc.dbo.t_new_order
union all
select * from rmt3.tpcc.dbo.t_new_order
union all
select * from rmt4.tpcc.dbo.t_new_order
union all
select * from rmt5.tpcc.dbo.t_new_order
union all
select * from rmt6.tpcc.dbo.t_new_order
union all
select * from rmt7.tpcc.dbo.t_new_order
union all
select * from rmt8.tpcc.dbo.t_new_order
union all
select * from rmt9.tpcc.dbo.t_new_order
union all
select * from rmt10.tpcc.dbo.t_new_order
union all
select * from rmt11.tpcc.dbo.t_new_order
union all
select * from rmt13.tpcc.dbo.t_new_order
union all
select * from rmt14.tpcc.dbo.t_new_order
union all
select * from rmt15.tpcc.dbo.t_new_order
union all
select * from rmt16.tpcc.dbo.t_new_order
go

```

```

create view item as
select * from t_item
go

```

add_views13.sql

```
-- file 21601_to_23400\add_views.sql
```

```

set ansi_warnings on
set ansi_nulls on
go

```

```

use tpcc
go

```

```

-- drop any existing views
if object_id('warehouse') is not null drop view warehouse
if object_id('district') is not null drop view district
if object_id('customer') is not null drop view customer
if object_id('history') is not null drop view history
if object_id('stock') is not null drop view stock
if object_id('orders') is not null drop view orders
if object_id('order_line') is not null drop view order_line
if object_id('new_order') is not null drop view new_order
if object_id('item') is not null drop view item

```

```

go

--add views for partition 13
create view warehouse as
select * from /*rmt13.tpcc.dbo.*/t_warehouse
union all
select * from rmt1.tpcc.dbo.t_warehouse
union all
select * from rmt2.tpcc.dbo.t_warehouse
union all
select * from rmt3.tpcc.dbo.t_warehouse
union all
select * from rmt4.tpcc.dbo.t_warehouse
union all
select * from rmt5.tpcc.dbo.t_warehouse
union all
select * from rmt6.tpcc.dbo.t_warehouse
union all
select * from rmt7.tpcc.dbo.t_warehouse
union all
select * from rmt8.tpcc.dbo.t_warehouse
union all
select * from rmt9.tpcc.dbo.t_warehouse
union all
select * from rmt10.tpcc.dbo.t_warehouse
union all
select * from rmt11.tpcc.dbo.t_warehouse
union all
select * from rmt12.tpcc.dbo.t_warehouse
union all
select * from rmt14.tpcc.dbo.t_warehouse
union all
select * from rmt15.tpcc.dbo.t_warehouse
union all
select * from rmt16.tpcc.dbo.t_warehouse
go

```

```

create view district as
select * from /*rmt13.tpcc.dbo.*/t_district
union all
select * from rmt1.tpcc.dbo.t_district
union all
select * from rmt2.tpcc.dbo.t_district
union all
select * from rmt3.tpcc.dbo.t_district
union all
select * from rmt4.tpcc.dbo.t_district
union all
select * from rmt5.tpcc.dbo.t_district
union all
select * from rmt6.tpcc.dbo.t_district
union all
select * from rmt7.tpcc.dbo.t_district
union all
select * from rmt8.tpcc.dbo.t_district
union all
select * from rmt9.tpcc.dbo.t_district
union all
select * from rmt10.tpcc.dbo.t_district
union all
select * from rmt11.tpcc.dbo.t_district
union all
select * from rmt12.tpcc.dbo.t_district
union all
select * from rmt14.tpcc.dbo.t_district
union all
select * from rmt15.tpcc.dbo.t_district
union all

```

```

select * from rmt16.tpcc.dbo.t_district
go

create view customer as
select * from /*rmt13.tpcc.dbo.*/t_customer
union all
select * from rmt1.tpcc.dbo.t_customer
union all
select * from rmt2.tpcc.dbo.t_customer
union all
select * from rmt3.tpcc.dbo.t_customer
union all
select * from rmt4.tpcc.dbo.t_customer
union all
select * from rmt5.tpcc.dbo.t_customer
union all
select * from rmt6.tpcc.dbo.t_customer
union all
select * from rmt7.tpcc.dbo.t_customer
union all
select * from rmt8.tpcc.dbo.t_customer
union all
select * from rmt9.tpcc.dbo.t_customer
union all
select * from rmt10.tpcc.dbo.t_customer
union all
select * from rmt11.tpcc.dbo.t_customer
union all
select * from rmt12.tpcc.dbo.t_customer
union all
select * from rmt14.tpcc.dbo.t_customer
union all
select * from rmt15.tpcc.dbo.t_customer
union all
select * from rmt16.tpcc.dbo.t_customer
go

```

```

create view history as
select * from /*rmt13.tpcc.dbo.*/t_history
union all
select * from rmt1.tpcc.dbo.t_history
union all
select * from rmt2.tpcc.dbo.t_history
union all
select * from rmt3.tpcc.dbo.t_history
union all
select * from rmt4.tpcc.dbo.t_history
union all
select * from rmt5.tpcc.dbo.t_history
union all
select * from rmt6.tpcc.dbo.t_history
union all
select * from rmt7.tpcc.dbo.t_history
union all
select * from rmt8.tpcc.dbo.t_history
union all
select * from rmt9.tpcc.dbo.t_history
union all
select * from rmt10.tpcc.dbo.t_history
union all
select * from rmt11.tpcc.dbo.t_history
union all
select * from rmt12.tpcc.dbo.t_history
union all
select * from rmt14.tpcc.dbo.t_history
union all
select * from rmt15.tpcc.dbo.t_history
union all

```

```

select * from rmt16.tpcc.dbo.t_history
go

create view stock as
select * from /*rmt13.tpcc.dbo.*/t_stock
union all
select * from rmt1.tpcc.dbo.t_stock
union all
select * from rmt2.tpcc.dbo.t_stock
union all
select * from rmt3.tpcc.dbo.t_stock
union all
select * from rmt4.tpcc.dbo.t_stock
union all
select * from rmt5.tpcc.dbo.t_stock
union all
select * from rmt6.tpcc.dbo.t_stock
union all
select * from rmt7.tpcc.dbo.t_stock
union all
select * from rmt8.tpcc.dbo.t_stock
union all
select * from rmt9.tpcc.dbo.t_stock
union all
select * from rmt10.tpcc.dbo.t_stock
union all
select * from rmt11.tpcc.dbo.t_stock
union all
select * from rmt12.tpcc.dbo.t_stock
union all
select * from rmt14.tpcc.dbo.t_stock
union all
select * from rmt15.tpcc.dbo.t_stock
union all
select * from rmt16.tpcc.dbo.t_stock
go

```

```

create view orders as
select * from /*rmt13.tpcc.dbo.*/t_orders
union all
select * from rmt1.tpcc.dbo.t_orders
union all
select * from rmt2.tpcc.dbo.t_orders
union all
select * from rmt3.tpcc.dbo.t_orders
union all
select * from rmt4.tpcc.dbo.t_orders
union all
select * from rmt5.tpcc.dbo.t_orders
union all
select * from rmt6.tpcc.dbo.t_orders
union all
select * from rmt7.tpcc.dbo.t_orders
union all
select * from rmt8.tpcc.dbo.t_orders
union all
select * from rmt9.tpcc.dbo.t_orders
union all
select * from rmt10.tpcc.dbo.t_orders
union all
select * from rmt11.tpcc.dbo.t_orders
union all
select * from rmt12.tpcc.dbo.t_orders
union all
select * from rmt14.tpcc.dbo.t_orders
union all
select * from rmt15.tpcc.dbo.t_orders
union all

```

```

select * from rmt16.tpcc.dbo.t_orders
go

create view order_line as
select * from /*rmt13.tpcc.dbo.*/t_order_line
union all
select * from rmt1.tpcc.dbo.t_order_line
union all
select * from rmt2.tpcc.dbo.t_order_line
union all
select * from rmt3.tpcc.dbo.t_order_line
union all
select * from rmt4.tpcc.dbo.t_order_line
union all
select * from rmt5.tpcc.dbo.t_order_line
union all
select * from rmt6.tpcc.dbo.t_order_line
union all
select * from rmt7.tpcc.dbo.t_order_line
union all
select * from rmt8.tpcc.dbo.t_order_line
union all
select * from rmt9.tpcc.dbo.t_order_line
union all
select * from rmt10.tpcc.dbo.t_order_line
union all
select * from rmt11.tpcc.dbo.t_order_line
union all
select * from rmt12.tpcc.dbo.t_order_line
union all
select * from rmt14.tpcc.dbo.t_order_line
union all
select * from rmt15.tpcc.dbo.t_order_line
union all
select * from rmt16.tpcc.dbo.t_order_line
go

```

```

create view new_order as
select * from /*rmt13.tpcc.dbo.*/t_new_order
union all
select * from rmt1.tpcc.dbo.t_new_order
union all
select * from rmt2.tpcc.dbo.t_new_order
union all
select * from rmt3.tpcc.dbo.t_new_order
union all
select * from rmt4.tpcc.dbo.t_new_order
union all
select * from rmt5.tpcc.dbo.t_new_order
union all
select * from rmt6.tpcc.dbo.t_new_order
union all
select * from rmt7.tpcc.dbo.t_new_order
union all
select * from rmt8.tpcc.dbo.t_new_order
union all
select * from rmt9.tpcc.dbo.t_new_order
union all
select * from rmt10.tpcc.dbo.t_new_order
union all
select * from rmt11.tpcc.dbo.t_new_order
union all
select * from rmt12.tpcc.dbo.t_new_order
union all
select * from rmt14.tpcc.dbo.t_new_order
union all
select * from rmt15.tpcc.dbo.t_new_order
union all

```

```
select * from rmt16.tpcc.dbo.t_new_order
go
```

```
create view item as
select * from t_item
go
```

add_views14.sql

```
-- file 23401_to_25200\add_views.sql
```

```
set ansi_warnings on
set ansi_nulls on
go
```

```
use tpcc
go
```

```
-- drop any existing views
if object_id('warehouse') is not null drop view warehouse
if object_id('district') is not null drop view district
if object_id('customer') is not null drop view customer
if object_id('history') is not null drop view history
if object_id('stock') is not null drop view stock
if object_id('orders') is not null drop view orders
if object_id('order_line') is not null drop view order_line
if object_id('new_order') is not null drop view new_order
if object_id('item') is not null drop view item
go
```

```
--add views for partition 14
```

```
create view warehouse as
select * from /*rmt14.tpcc.dbo.*/t_warehouse
union all
select * from rmt1.tpcc.dbo.t_warehouse
union all
select * from rmt2.tpcc.dbo.t_warehouse
union all
select * from rmt3.tpcc.dbo.t_warehouse
union all
select * from rmt4.tpcc.dbo.t_warehouse
union all
select * from rmt5.tpcc.dbo.t_warehouse
union all
select * from rmt6.tpcc.dbo.t_warehouse
union all
select * from rmt7.tpcc.dbo.t_warehouse
union all
select * from rmt8.tpcc.dbo.t_warehouse
union all
select * from rmt9.tpcc.dbo.t_warehouse
union all
select * from rmt10.tpcc.dbo.t_warehouse
union all
select * from rmt11.tpcc.dbo.t_warehouse
union all
select * from rmt12.tpcc.dbo.t_warehouse
union all
select * from rmt13.tpcc.dbo.t_warehouse
union all
select * from rmt15.tpcc.dbo.t_warehouse
union all
select * from rmt16.tpcc.dbo.t_warehouse
go
```

```
create view district as
```

```
select * from /*rmt14.tpcc.dbo.*/t_district
union all
select * from rmt1.tpcc.dbo.t_district
union all
select * from rmt2.tpcc.dbo.t_district
union all
select * from rmt3.tpcc.dbo.t_district
union all
select * from rmt4.tpcc.dbo.t_district
union all
select * from rmt5.tpcc.dbo.t_district
union all
select * from rmt6.tpcc.dbo.t_district
union all
select * from rmt7.tpcc.dbo.t_district
union all
select * from rmt8.tpcc.dbo.t_district
union all
select * from rmt9.tpcc.dbo.t_district
union all
select * from rmt10.tpcc.dbo.t_district
union all
select * from rmt11.tpcc.dbo.t_district
union all
select * from rmt12.tpcc.dbo.t_district
union all
select * from rmt13.tpcc.dbo.t_district
union all
select * from rmt15.tpcc.dbo.t_district
union all
select * from rmt16.tpcc.dbo.t_district
go
```

```
create view customer as
select * from /*rmt14.tpcc.dbo.*/t_customer
union all
select * from rmt1.tpcc.dbo.t_customer
union all
select * from rmt2.tpcc.dbo.t_customer
union all
select * from rmt3.tpcc.dbo.t_customer
union all
select * from rmt4.tpcc.dbo.t_customer
union all
select * from rmt5.tpcc.dbo.t_customer
union all
select * from rmt6.tpcc.dbo.t_customer
union all
select * from rmt7.tpcc.dbo.t_customer
union all
select * from rmt8.tpcc.dbo.t_customer
union all
select * from rmt9.tpcc.dbo.t_customer
union all
select * from rmt10.tpcc.dbo.t_customer
union all
select * from rmt11.tpcc.dbo.t_customer
union all
select * from rmt12.tpcc.dbo.t_customer
union all
select * from rmt13.tpcc.dbo.t_customer
union all
select * from rmt15.tpcc.dbo.t_customer
union all
select * from rmt16.tpcc.dbo.t_customer
go
```

```
create view history as
```



```

select * from /*rmt14.tpcc.dbo.*/t_history
union all
select * from rmt1.tpcc.dbo.t_history
union all
select * from rmt2.tpcc.dbo.t_history
union all
select * from rmt3.tpcc.dbo.t_history
union all
select * from rmt4.tpcc.dbo.t_history
union all
select * from rmt5.tpcc.dbo.t_history
union all
select * from rmt6.tpcc.dbo.t_history
union all
select * from rmt7.tpcc.dbo.t_history
union all
select * from rmt8.tpcc.dbo.t_history
union all
select * from rmt9.tpcc.dbo.t_history
union all
select * from rmt10.tpcc.dbo.t_history
union all
select * from rmt11.tpcc.dbo.t_history
union all
select * from rmt12.tpcc.dbo.t_history
union all
select * from rmt13.tpcc.dbo.t_history
union all
select * from rmt15.tpcc.dbo.t_history
union all
select * from rmt16.tpcc.dbo.t_history
go

```

```

create view stock as
select * from /*rmt14.tpcc.dbo.*/t_stock
union all
select * from rmt1.tpcc.dbo.t_stock
union all
select * from rmt2.tpcc.dbo.t_stock
union all
select * from rmt3.tpcc.dbo.t_stock
union all
select * from rmt4.tpcc.dbo.t_stock
union all
select * from rmt5.tpcc.dbo.t_stock
union all
select * from rmt6.tpcc.dbo.t_stock
union all
select * from rmt7.tpcc.dbo.t_stock
union all
select * from rmt8.tpcc.dbo.t_stock
union all
select * from rmt9.tpcc.dbo.t_stock
union all
select * from rmt10.tpcc.dbo.t_stock
union all
select * from rmt11.tpcc.dbo.t_stock
union all
select * from rmt12.tpcc.dbo.t_stock
union all
select * from rmt13.tpcc.dbo.t_stock
union all
select * from rmt15.tpcc.dbo.t_stock
union all
select * from rmt16.tpcc.dbo.t_stock
go

```

```

create view orders as

```

```

select * from /*rmt14.tpcc.dbo.*/t_orders
union all
select * from rmt1.tpcc.dbo.t_orders
union all
select * from rmt2.tpcc.dbo.t_orders
union all
select * from rmt3.tpcc.dbo.t_orders
union all
select * from rmt4.tpcc.dbo.t_orders
union all
select * from rmt5.tpcc.dbo.t_orders
union all
select * from rmt6.tpcc.dbo.t_orders
union all
select * from rmt7.tpcc.dbo.t_orders
union all
select * from rmt8.tpcc.dbo.t_orders
union all
select * from rmt9.tpcc.dbo.t_orders
union all
select * from rmt10.tpcc.dbo.t_orders
union all
select * from rmt11.tpcc.dbo.t_orders
union all
select * from rmt12.tpcc.dbo.t_orders
union all
select * from rmt13.tpcc.dbo.t_orders
union all
select * from rmt15.tpcc.dbo.t_orders
union all
select * from rmt16.tpcc.dbo.t_orders
go

```

```

create view order_line as
select * from /*rmt14.tpcc.dbo.*/t_order_line
union all
select * from rmt1.tpcc.dbo.t_order_line
union all
select * from rmt2.tpcc.dbo.t_order_line
union all
select * from rmt3.tpcc.dbo.t_order_line
union all
select * from rmt4.tpcc.dbo.t_order_line
union all
select * from rmt5.tpcc.dbo.t_order_line
union all
select * from rmt6.tpcc.dbo.t_order_line
union all
select * from rmt7.tpcc.dbo.t_order_line
union all
select * from rmt8.tpcc.dbo.t_order_line
union all
select * from rmt9.tpcc.dbo.t_order_line
union all
select * from rmt10.tpcc.dbo.t_order_line
union all
select * from rmt11.tpcc.dbo.t_order_line
union all
select * from rmt12.tpcc.dbo.t_order_line
union all
select * from rmt13.tpcc.dbo.t_order_line
union all
select * from rmt15.tpcc.dbo.t_order_line
union all
select * from rmt16.tpcc.dbo.t_order_line
go

```

```

create view new_order as

```

```

select * from /*rmt14.tpc.dbo.*/t_new_order
union all
select * from rmt1.tpc.dbo.t_new_order
union all
select * from rmt2.tpc.dbo.t_new_order
union all
select * from rmt3.tpc.dbo.t_new_order
union all
select * from rmt4.tpc.dbo.t_new_order
union all
select * from rmt5.tpc.dbo.t_new_order
union all
select * from rmt6.tpc.dbo.t_new_order
union all
select * from rmt7.tpc.dbo.t_new_order
union all
select * from rmt8.tpc.dbo.t_new_order
union all
select * from rmt9.tpc.dbo.t_new_order
union all
select * from rmt10.tpc.dbo.t_new_order
union all
select * from rmt11.tpc.dbo.t_new_order
union all
select * from rmt12.tpc.dbo.t_new_order
union all
select * from rmt13.tpc.dbo.t_new_order
union all
select * from rmt15.tpc.dbo.t_new_order
union all
select * from rmt16.tpc.dbo.t_new_order
go

```

```

create view item as
select * from t_item
go

```

add_views15.sql

```
-- file 25201_to_27000\add_views.sql
```

```

set ansi_warnings on
set ansi_nulls on
go

```

```

use tpcc
go

```

```

-- drop any existing views
if object_id('warehouse') is not null drop view warehouse
if object_id('district') is not null drop view district
if object_id('customer') is not null drop view customer
if object_id('history') is not null drop view history
if object_id('stock') is not null drop view stock
if object_id('orders') is not null drop view orders
if object_id('order_line') is not null drop view order_line
if object_id('new_order') is not null drop view new_order
if object_id('item') is not null drop view item
go

```

```

--add views for partition 15
create view warehouse as
select * from /*rmt15.tpc.dbo.*/t_warehouse
union all
select * from rmt1.tpc.dbo.t_warehouse
union all

```

```

select * from rmt2.tpc.dbo.t_warehouse
union all
select * from rmt3.tpc.dbo.t_warehouse
union all
select * from rmt4.tpc.dbo.t_warehouse
union all
select * from rmt5.tpc.dbo.t_warehouse
union all
select * from rmt6.tpc.dbo.t_warehouse
union all
select * from rmt7.tpc.dbo.t_warehouse
union all
select * from rmt8.tpc.dbo.t_warehouse
union all
select * from rmt9.tpc.dbo.t_warehouse
union all
select * from rmt10.tpc.dbo.t_warehouse
union all
select * from rmt11.tpc.dbo.t_warehouse
union all
select * from rmt12.tpc.dbo.t_warehouse
union all
select * from rmt13.tpc.dbo.t_warehouse
union all
select * from rmt14.tpc.dbo.t_warehouse
union all
select * from rmt16.tpc.dbo.t_warehouse
go

```

```

create view district as
select * from /*rmt15.tpc.dbo.*/t_district
union all
select * from rmt1.tpc.dbo.t_district
union all
select * from rmt2.tpc.dbo.t_district
union all
select * from rmt3.tpc.dbo.t_district
union all
select * from rmt4.tpc.dbo.t_district
union all
select * from rmt5.tpc.dbo.t_district
union all
select * from rmt6.tpc.dbo.t_district
union all
select * from rmt7.tpc.dbo.t_district
union all
select * from rmt8.tpc.dbo.t_district
union all
select * from rmt9.tpc.dbo.t_district
union all
select * from rmt10.tpc.dbo.t_district
union all
select * from rmt11.tpc.dbo.t_district
union all
select * from rmt12.tpc.dbo.t_district
union all
select * from rmt13.tpc.dbo.t_district
union all
select * from rmt14.tpc.dbo.t_district
union all
select * from rmt16.tpc.dbo.t_district
go

```

```

create view customer as
select * from /*rmt15.tpc.dbo.*/t_customer
union all
select * from rmt1.tpc.dbo.t_customer
union all

```

```

select * from rmt2.tpcc.dbo.t_customer
union all
select * from rmt3.tpcc.dbo.t_customer
union all
select * from rmt4.tpcc.dbo.t_customer
union all
select * from rmt5.tpcc.dbo.t_customer
union all
select * from rmt6.tpcc.dbo.t_customer
union all
select * from rmt7.tpcc.dbo.t_customer
union all
select * from rmt8.tpcc.dbo.t_customer
union all
select * from rmt9.tpcc.dbo.t_customer
union all
select * from rmt10.tpcc.dbo.t_customer
union all
select * from rmt11.tpcc.dbo.t_customer
union all
select * from rmt12.tpcc.dbo.t_customer
union all
select * from rmt13.tpcc.dbo.t_customer
union all
select * from rmt14.tpcc.dbo.t_customer
union all
select * from rmt16.tpcc.dbo.t_customer
go

```

```

create view history as
select * from /*rmt15.tpcc.dbo.*/t_history
union all
select * from rmt1.tpcc.dbo.t_history
union all
select * from rmt2.tpcc.dbo.t_history
union all
select * from rmt3.tpcc.dbo.t_history
union all
select * from rmt4.tpcc.dbo.t_history
union all
select * from rmt5.tpcc.dbo.t_history
union all
select * from rmt6.tpcc.dbo.t_history
union all
select * from rmt7.tpcc.dbo.t_history
union all
select * from rmt8.tpcc.dbo.t_history
union all
select * from rmt9.tpcc.dbo.t_history
union all
select * from rmt10.tpcc.dbo.t_history
union all
select * from rmt11.tpcc.dbo.t_history
union all
select * from rmt12.tpcc.dbo.t_history
union all
select * from rmt13.tpcc.dbo.t_history
union all
select * from rmt14.tpcc.dbo.t_history
union all
select * from rmt16.tpcc.dbo.t_history
go

```

```

create view stock as
select * from /*rmt15.tpcc.dbo.*/t_stock
union all
select * from rmt1.tpcc.dbo.t_stock
union all

```

```

select * from rmt2.tpcc.dbo.t_stock
union all
select * from rmt3.tpcc.dbo.t_stock
union all
select * from rmt4.tpcc.dbo.t_stock
union all
select * from rmt5.tpcc.dbo.t_stock
union all
select * from rmt6.tpcc.dbo.t_stock
union all
select * from rmt7.tpcc.dbo.t_stock
union all
select * from rmt8.tpcc.dbo.t_stock
union all
select * from rmt9.tpcc.dbo.t_stock
union all
select * from rmt10.tpcc.dbo.t_stock
union all
select * from rmt11.tpcc.dbo.t_stock
union all
select * from rmt12.tpcc.dbo.t_stock
union all
select * from rmt13.tpcc.dbo.t_stock
union all
select * from rmt14.tpcc.dbo.t_stock
union all
select * from rmt16.tpcc.dbo.t_stock
go

```

```

create view orders as
select * from /*rmt15.tpcc.dbo.*/t_orders
union all
select * from rmt1.tpcc.dbo.t_orders
union all
select * from rmt2.tpcc.dbo.t_orders
union all
select * from rmt3.tpcc.dbo.t_orders
union all
select * from rmt4.tpcc.dbo.t_orders
union all
select * from rmt5.tpcc.dbo.t_orders
union all
select * from rmt6.tpcc.dbo.t_orders
union all
select * from rmt7.tpcc.dbo.t_orders
union all
select * from rmt8.tpcc.dbo.t_orders
union all
select * from rmt9.tpcc.dbo.t_orders
union all
select * from rmt10.tpcc.dbo.t_orders
union all
select * from rmt11.tpcc.dbo.t_orders
union all
select * from rmt12.tpcc.dbo.t_orders
union all
select * from rmt13.tpcc.dbo.t_orders
union all
select * from rmt14.tpcc.dbo.t_orders
union all
select * from rmt16.tpcc.dbo.t_orders
go

```

```

create view order_line as
select * from /*rmt15.tpcc.dbo.*/t_order_line
union all
select * from rmt1.tpcc.dbo.t_order_line
union all

```

```

select * from rmt2.tpc.dbo.t_order_line
union all
select * from rmt3.tpc.dbo.t_order_line
union all
select * from rmt4.tpc.dbo.t_order_line
union all
select * from rmt5.tpc.dbo.t_order_line
union all
select * from rmt6.tpc.dbo.t_order_line
union all
select * from rmt7.tpc.dbo.t_order_line
union all
select * from rmt8.tpc.dbo.t_order_line
union all
select * from rmt9.tpc.dbo.t_order_line
union all
select * from rmt10.tpc.dbo.t_order_line
union all
select * from rmt11.tpc.dbo.t_order_line
union all
select * from rmt12.tpc.dbo.t_order_line
union all
select * from rmt13.tpc.dbo.t_order_line
union all
select * from rmt14.tpc.dbo.t_order_line
union all
select * from rmt16.tpc.dbo.t_order_line
go

create view new_order as
select * from /*rmt15.tpc.dbo.*/t_new_order
union all
select * from rmt1.tpc.dbo.t_new_order
union all
select * from rmt2.tpc.dbo.t_new_order
union all
select * from rmt3.tpc.dbo.t_new_order
union all
select * from rmt4.tpc.dbo.t_new_order
union all
select * from rmt5.tpc.dbo.t_new_order
union all
select * from rmt6.tpc.dbo.t_new_order
union all
select * from rmt7.tpc.dbo.t_new_order
union all
select * from rmt8.tpc.dbo.t_new_order
union all
select * from rmt9.tpc.dbo.t_new_order
union all
select * from rmt10.tpc.dbo.t_new_order
union all
select * from rmt11.tpc.dbo.t_new_order
union all
select * from rmt12.tpc.dbo.t_new_order
union all
select * from rmt13.tpc.dbo.t_new_order
union all
select * from rmt14.tpc.dbo.t_new_order
union all
select * from rmt16.tpc.dbo.t_new_order
go

create view item as
select * from t_item
go

```

add_views16.sql

```

-- file 27001_to_28800\add_views.sql

set ansi_warnings on
set ansi_nulls on
go

use tpcc
go

-- drop any existing views
if object_id('warehouse') is not null drop view warehouse
if object_id('district') is not null drop view district
if object_id('customer') is not null drop view customer
if object_id('history') is not null drop view history
if object_id('stock') is not null drop view stock
if object_id('orders') is not null drop view orders
if object_id('order_line') is not null drop view order_line
if object_id('new_order') is not null drop view new_order
if object_id('item') is not null drop view item
go

--add views for partition 16
create view warehouse as
select * from /*rmt16.tpc.dbo.*/t_warehouse
union all
select * from rmt1.tpc.dbo.t_warehouse
union all
select * from rmt2.tpc.dbo.t_warehouse
union all
select * from rmt3.tpc.dbo.t_warehouse
union all
select * from rmt4.tpc.dbo.t_warehouse
union all
select * from rmt5.tpc.dbo.t_warehouse
union all
select * from rmt6.tpc.dbo.t_warehouse
union all
select * from rmt7.tpc.dbo.t_warehouse
union all
select * from rmt8.tpc.dbo.t_warehouse
union all
select * from rmt9.tpc.dbo.t_warehouse
union all
select * from rmt10.tpc.dbo.t_warehouse
union all
select * from rmt11.tpc.dbo.t_warehouse
union all
select * from rmt12.tpc.dbo.t_warehouse
union all
select * from rmt13.tpc.dbo.t_warehouse
union all
select * from rmt14.tpc.dbo.t_warehouse
union all
select * from rmt15.tpc.dbo.t_warehouse
go

create view district as
select * from /*rmt16.tpc.dbo.*/t_district
union all
select * from rmt1.tpc.dbo.t_district
union all
select * from rmt2.tpc.dbo.t_district
union all
select * from rmt3.tpc.dbo.t_district
union all

```

```

select * from rmt4.tpc.dbo.t_district
union all
select * from rmt5.tpc.dbo.t_district
union all
select * from rmt6.tpc.dbo.t_district
union all
select * from rmt7.tpc.dbo.t_district
union all
select * from rmt8.tpc.dbo.t_district
union all
select * from rmt9.tpc.dbo.t_district
union all
select * from rmt10.tpc.dbo.t_district
union all
select * from rmt11.tpc.dbo.t_district
union all
select * from rmt12.tpc.dbo.t_district
union all
select * from rmt13.tpc.dbo.t_district
union all
select * from rmt14.tpc.dbo.t_district
union all
select * from rmt15.tpc.dbo.t_district
go

```

```

create view customer as
select * from /*rmt16.tpc.dbo.*/t_customer
union all
select * from rmt1.tpc.dbo.t_customer
union all
select * from rmt2.tpc.dbo.t_customer
union all
select * from rmt3.tpc.dbo.t_customer
union all
select * from rmt4.tpc.dbo.t_customer
union all
select * from rmt5.tpc.dbo.t_customer
union all
select * from rmt6.tpc.dbo.t_customer
union all
select * from rmt7.tpc.dbo.t_customer
union all
select * from rmt8.tpc.dbo.t_customer
union all
select * from rmt9.tpc.dbo.t_customer
union all
select * from rmt10.tpc.dbo.t_customer
union all
select * from rmt11.tpc.dbo.t_customer
union all
select * from rmt12.tpc.dbo.t_customer
union all
select * from rmt13.tpc.dbo.t_customer
union all
select * from rmt14.tpc.dbo.t_customer
union all
select * from rmt15.tpc.dbo.t_customer
go

```

```

create view history as
select * from /*rmt16.tpc.dbo.*/t_history
union all
select * from rmt1.tpc.dbo.t_history
union all
select * from rmt2.tpc.dbo.t_history
union all
select * from rmt3.tpc.dbo.t_history
union all

```

```

select * from rmt4.tpc.dbo.t_history
union all
select * from rmt5.tpc.dbo.t_history
union all
select * from rmt6.tpc.dbo.t_history
union all
select * from rmt7.tpc.dbo.t_history
union all
select * from rmt8.tpc.dbo.t_history
union all
select * from rmt9.tpc.dbo.t_history
union all
select * from rmt10.tpc.dbo.t_history
union all
select * from rmt11.tpc.dbo.t_history
union all
select * from rmt12.tpc.dbo.t_history
union all
select * from rmt13.tpc.dbo.t_history
union all
select * from rmt14.tpc.dbo.t_history
union all
select * from rmt15.tpc.dbo.t_history
go

```

```

create view stock as
select * from /*rmt16.tpc.dbo.*/t_stock
union all
select * from rmt1.tpc.dbo.t_stock
union all
select * from rmt2.tpc.dbo.t_stock
union all
select * from rmt3.tpc.dbo.t_stock
union all
select * from rmt4.tpc.dbo.t_stock
union all
select * from rmt5.tpc.dbo.t_stock
union all
select * from rmt6.tpc.dbo.t_stock
union all
select * from rmt7.tpc.dbo.t_stock
union all
select * from rmt8.tpc.dbo.t_stock
union all
select * from rmt9.tpc.dbo.t_stock
union all
select * from rmt10.tpc.dbo.t_stock
union all
select * from rmt11.tpc.dbo.t_stock
union all
select * from rmt12.tpc.dbo.t_stock
union all
select * from rmt13.tpc.dbo.t_stock
union all
select * from rmt14.tpc.dbo.t_stock
union all
select * from rmt15.tpc.dbo.t_stock
go

```

```

create view orders as
select * from /*rmt16.tpc.dbo.*/t_orders
union all
select * from rmt1.tpc.dbo.t_orders
union all
select * from rmt2.tpc.dbo.t_orders
union all
select * from rmt3.tpc.dbo.t_orders
union all

```

```

select * from rmt4.tpcc.dbo.t_orders
union all
select * from rmt5.tpcc.dbo.t_orders
union all
select * from rmt6.tpcc.dbo.t_orders
union all
select * from rmt7.tpcc.dbo.t_orders
union all
select * from rmt8.tpcc.dbo.t_orders
union all
select * from rmt9.tpcc.dbo.t_orders
union all
select * from rmt10.tpcc.dbo.t_orders
union all
select * from rmt11.tpcc.dbo.t_orders
union all
select * from rmt12.tpcc.dbo.t_orders
union all
select * from rmt13.tpcc.dbo.t_orders
union all
select * from rmt14.tpcc.dbo.t_orders
union all
select * from rmt15.tpcc.dbo.t_orders
go

```

```

create view order_line as
select * from /*rmt16.tpcc.dbo.*/t_order_line
union all
select * from rmt1.tpcc.dbo.t_order_line
union all
select * from rmt2.tpcc.dbo.t_order_line
union all
select * from rmt3.tpcc.dbo.t_order_line
union all
select * from rmt4.tpcc.dbo.t_order_line
union all
select * from rmt5.tpcc.dbo.t_order_line
union all
select * from rmt6.tpcc.dbo.t_order_line
union all
select * from rmt7.tpcc.dbo.t_order_line
union all
select * from rmt8.tpcc.dbo.t_order_line
union all
select * from rmt9.tpcc.dbo.t_order_line
union all
select * from rmt10.tpcc.dbo.t_order_line
union all
select * from rmt11.tpcc.dbo.t_order_line
union all
select * from rmt12.tpcc.dbo.t_order_line
union all
select * from rmt13.tpcc.dbo.t_order_line
union all
select * from rmt14.tpcc.dbo.t_order_line
union all
select * from rmt15.tpcc.dbo.t_order_line
go

```

```

create view new_order as
select * from /*rmt16.tpcc.dbo.*/t_new_order
union all
select * from rmt1.tpcc.dbo.t_new_order
union all
select * from rmt2.tpcc.dbo.t_new_order
union all
select * from rmt3.tpcc.dbo.t_new_order
union all

```

```

select * from rmt4.tpcc.dbo.t_new_order
union all
select * from rmt5.tpcc.dbo.t_new_order
union all
select * from rmt6.tpcc.dbo.t_new_order
union all
select * from rmt7.tpcc.dbo.t_new_order
union all
select * from rmt8.tpcc.dbo.t_new_order
union all
select * from rmt9.tpcc.dbo.t_new_order
union all
select * from rmt10.tpcc.dbo.t_new_order
union all
select * from rmt11.tpcc.dbo.t_new_order
union all
select * from rmt12.tpcc.dbo.t_new_order
union all
select * from rmt13.tpcc.dbo.t_new_order
union all
select * from rmt14.tpcc.dbo.t_new_order
union all
select * from rmt15.tpcc.dbo.t_new_order
go

```

```

create view item as
select * from t_item
go

```

item_iot1.sql

```

-- file 1_to_1800\item_iot.sql
--add item instead-of-trigger for partition 1

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONS off
go

use tpcc
go

drop trigger iot_item
go

create trigger iot_item on item instead of update, insert, delete as
begin
--
-- local node (partition 1)
DELETE A FROM t_item A, DELETED D WHERE A.i_id = D.i_id
INSERT t_item SELECT * FROM INSERTED
--
-- remote node, partition 2
DELETE A FROM rmt2.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt2.tpcc.dbo.t_item SELECT * FROM INSERTED
--
-- remote node, partition 3
DELETE A FROM rmt3.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt3.tpcc.dbo.t_item SELECT * FROM INSERTED
--
-- remote node, partition 4

```

```

DELETE A FROM rmt4.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt4.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 5
DELETE A FROM rmt5.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt5.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 6
DELETE A FROM rmt6.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt6.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 7
DELETE A FROM rmt7.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt7.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 8
DELETE A FROM rmt8.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt8.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 9
DELETE A FROM rmt9.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt9.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 10
DELETE A FROM rmt10.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt10.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 11
DELETE A FROM rmt11.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt11.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 12
DELETE A FROM rmt12.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt12.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 13
DELETE A FROM rmt13.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt13.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 14
DELETE A FROM rmt14.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt14.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 15
DELETE A FROM rmt15.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id

```

```

INSERT rmt15.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 16
DELETE A FROM rmt16.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt16.tpcc.dbo.t_item SELECT * FROM INSERTED

end

```

item_iot2.sql

```

-- file 1801_to_3600\item_iot.sql

--add item instead-of-trigger for partition 2

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONS off
go

use tpcc
go

drop trigger iot_item
go

create trigger iot_item on item instead of update, insert, delete as
begin

--
-- remote node, partition 1
DELETE A FROM rmt1.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt1.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- local node (partition 2)
DELETE A FROM t_item A, DELETED D WHERE A.i_id = D.i_id
INSERT t_item SELECT * FROM INSERTED

--
-- remote node, partition 3
DELETE A FROM rmt3.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt3.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 4
DELETE A FROM rmt4.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt4.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 5
DELETE A FROM rmt5.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt5.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 6
DELETE A FROM rmt6.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt6.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 7

```

```

DELETE A FROM rmt7.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt7.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 8
DELETE A FROM rmt8.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt8.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 9
DELETE A FROM rmt9.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt9.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 10
DELETE A FROM rmt10.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt10.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 11
DELETE A FROM rmt11.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt11.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 12
DELETE A FROM rmt12.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt12.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 13
DELETE A FROM rmt13.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt13.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 14
DELETE A FROM rmt14.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt14.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 15
DELETE A FROM rmt15.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt15.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 16
DELETE A FROM rmt16.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt16.tpcc.dbo.t_item SELECT * FROM INSERTED

end

```

item_iot3.sql

```

-- file 3601_to_5400\item_iot.sql
--add item instead-of-trigger for partition 3

```

```

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONS off
go

use tpcc
go

drop trigger iot_item
go

create trigger iot_item on item instead of update, insert, delete as
begin
--
-- remote node, partition 1
DELETE A FROM rmt1.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt1.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 2
DELETE A FROM rmt2.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt2.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- local node (partition 3)
DELETE A FROM t_item A, DELETED D WHERE A.i_id = D.i_id
INSERT t_item SELECT * FROM INSERTED

--
-- remote node, partition 4
DELETE A FROM rmt4.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt4.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 5
DELETE A FROM rmt5.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt5.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 6
DELETE A FROM rmt6.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt6.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 7
DELETE A FROM rmt7.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt7.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 8
DELETE A FROM rmt8.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt8.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 9
DELETE A FROM rmt9.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt9.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 10

```



```

DELETE A FROM rmt10.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt10.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 11
DELETE A FROM rmt11.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt11.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 12
DELETE A FROM rmt12.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt12.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 13
DELETE A FROM rmt13.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt13.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 14
DELETE A FROM rmt14.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt14.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 15
DELETE A FROM rmt15.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt15.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 16
DELETE A FROM rmt16.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt16.tpcc.dbo.t_item SELECT * FROM INSERTED

end

```

item_iot4.sql

```

-- file 5401_to_7200\item_iot.sql

--add item instead-of-trigger for partition 4

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONS off
go

use tpcc
go

drop trigger iot_item
go

create trigger iot_item on item instead of update, insert, delete as
begin

--
-- remote node, partition 1
DELETE A FROM rmt1.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt1.tpcc.dbo.t_item SELECT * FROM INSERTED

```

```

--
-- remote node, partition 2
DELETE A FROM rmt2.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt2.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 3
DELETE A FROM rmt3.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt3.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- local node (partition 4)
DELETE A FROM t_item A, DELETED D WHERE A.i_id = D.i_id
INSERT t_item SELECT * FROM INSERTED

--
-- remote node, partition 5
DELETE A FROM rmt5.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt5.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 6
DELETE A FROM rmt6.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt6.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 7
DELETE A FROM rmt7.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt7.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 8
DELETE A FROM rmt8.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt8.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 9
DELETE A FROM rmt9.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt9.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 10
DELETE A FROM rmt10.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt10.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 11
DELETE A FROM rmt11.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt11.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 12
DELETE A FROM rmt12.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt12.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 13

```

```

DELETE A FROM rmt13.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt13.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 14
DELETE A FROM rmt14.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt14.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 15
DELETE A FROM rmt15.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt15.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 16
DELETE A FROM rmt16.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt16.tpcc.dbo.t_item SELECT * FROM INSERTED

end

```

item_iot5.sql

```

-- file 7201_to_9000\item_iot.sql

--add item instead-of-trigger for partition 5

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONSOFF
go

use tpcc
go

drop trigger iot_item
go

create trigger iot_item on item instead of update, insert, delete as
begin

--
-- remote node, partition 1
DELETE A FROM rmt1.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt1.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 2
DELETE A FROM rmt2.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt2.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 3
DELETE A FROM rmt3.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt3.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 4
DELETE A FROM rmt4.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt4.tpcc.dbo.t_item SELECT * FROM INSERTED

```

```

--
-- local node (partition 5)
DELETE A FROM t_item A, DELETED D WHERE A.i_id = D.i_id
INSERT t_item SELECT * FROM INSERTED

--
-- remote node, partition 6
DELETE A FROM rmt6.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt6.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 7
DELETE A FROM rmt7.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt7.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 8
DELETE A FROM rmt8.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt8.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 9
DELETE A FROM rmt9.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt9.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 10
DELETE A FROM rmt10.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt10.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 11
DELETE A FROM rmt11.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt11.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 12
DELETE A FROM rmt12.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt12.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 13
DELETE A FROM rmt13.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt13.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 14
DELETE A FROM rmt14.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt14.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 15
DELETE A FROM rmt15.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt15.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 16

```

```

DELETE A FROM rmt16.tpc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt16.tpc.dbo.t_item SELECT * FROM INSERTED

end

```

item_iot6.sql

```

-- file 9001_to_10800\item_iot.sql

--add item instead-of-trigger for partition 6

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONSOFF
go

use tpcc
go

drop trigger iot_item
go

create trigger iot_item on item instead of update, insert, delete as
begin

--
-- remote node, partition 1
DELETE A FROM rmt1.tpc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt1.tpc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 2
DELETE A FROM rmt2.tpc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt2.tpc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 3
DELETE A FROM rmt3.tpc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt3.tpc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 4
DELETE A FROM rmt4.tpc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt4.tpc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 5
DELETE A FROM rmt5.tpc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt5.tpc.dbo.t_item SELECT * FROM INSERTED

--
-- local node (partition 6)
DELETE A FROM t_item A, DELETED D WHERE A.i_id = D.i_id
INSERT t_item SELECT * FROM INSERTED

--
-- remote node, partition 7
DELETE A FROM rmt7.tpc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt7.tpc.dbo.t_item SELECT * FROM INSERTED

```

```

--
-- remote node, partition 8
DELETE A FROM rmt8.tpc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt8.tpc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 9
DELETE A FROM rmt9.tpc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt9.tpc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 10
DELETE A FROM rmt10.tpc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt10.tpc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 11
DELETE A FROM rmt11.tpc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt11.tpc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 12
DELETE A FROM rmt12.tpc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt12.tpc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 13
DELETE A FROM rmt13.tpc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt13.tpc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 14
DELETE A FROM rmt14.tpc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt14.tpc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 15
DELETE A FROM rmt15.tpc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt15.tpc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 16
DELETE A FROM rmt16.tpc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt16.tpc.dbo.t_item SELECT * FROM INSERTED

end

```

item_iot7.sql

```

-- file 10801_to_12600\item_iot.sql

--add item instead-of-trigger for partition 7

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONSOFF
go

```

```

use tpcc
go

drop trigger iot_item
go

create trigger iot_item on item instead of update, insert, delete as
begin

--
-- remote node, partition 1
DELETE A FROM rmt1.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt1.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 2
DELETE A FROM rmt2.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt2.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 3
DELETE A FROM rmt3.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt3.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 4
DELETE A FROM rmt4.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt4.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 5
DELETE A FROM rmt5.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt5.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 6
DELETE A FROM rmt6.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt6.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- local node (partition 7)
DELETE A FROM t_item A, DELETED D WHERE A.i_id = D.i_id
INSERT t_item SELECT * FROM INSERTED

--
-- remote node, partition 8
DELETE A FROM rmt8.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt8.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 9
DELETE A FROM rmt9.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt9.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 10
DELETE A FROM rmt10.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt10.tpcc.dbo.t_item SELECT * FROM INSERTED

```

```

--
-- remote node, partition 11
DELETE A FROM rmt11.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt11.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 12
DELETE A FROM rmt12.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt12.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 13
DELETE A FROM rmt13.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt13.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 14
DELETE A FROM rmt14.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt14.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 15
DELETE A FROM rmt15.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt15.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 16
DELETE A FROM rmt16.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt16.tpcc.dbo.t_item SELECT * FROM INSERTED

end

```

item_iot8.sql

```

-- file 12601_to_14400\item_iot.sql
--add item instead-of-trigger for partition 8

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONS off
go

use tpcc
go

drop trigger iot_item
go

create trigger iot_item on item instead of update, insert, delete as
begin

--
-- remote node, partition 1
DELETE A FROM rmt1.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt1.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 2

```

```

DELETE A FROM rmt2.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt2.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 3
DELETE A FROM rmt3.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt3.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 4
DELETE A FROM rmt4.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt4.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 5
DELETE A FROM rmt5.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt5.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 6
DELETE A FROM rmt6.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt6.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 7
DELETE A FROM rmt7.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt7.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- local node (partition 8)
DELETE A FROM t_item A, DELETED D WHERE A.i_id = D.i_id
INSERT t_item SELECT * FROM INSERTED

--
-- remote node, partition 9
DELETE A FROM rmt9.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt9.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 10
DELETE A FROM rmt10.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt10.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 11
DELETE A FROM rmt11.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt11.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 12
DELETE A FROM rmt12.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt12.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 13
DELETE A FROM rmt13.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt13.tpcc.dbo.t_item SELECT * FROM INSERTED

```

```

--
-- remote node, partition 14
DELETE A FROM rmt14.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt14.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 15
DELETE A FROM rmt15.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt15.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 16
DELETE A FROM rmt16.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt16.tpcc.dbo.t_item SELECT * FROM INSERTED

end

```

item_iot9.sql

```

-- file 14401_to_16200\item_iot.sql
--add item instead-of-trigger for partition 9

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONS off
go

use tpcc
go

drop trigger iot_item
go

create trigger iot_item on item instead of update, insert, delete as
begin

--
-- remote node, partition 1
DELETE A FROM rmt1.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt1.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 2
DELETE A FROM rmt2.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt2.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 3
DELETE A FROM rmt3.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt3.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 4
DELETE A FROM rmt4.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt4.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 5

```

```

DELETE A FROM rmt5.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt5.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 6
DELETE A FROM rmt6.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt6.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 7
DELETE A FROM rmt7.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt7.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 8
DELETE A FROM rmt8.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt8.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- local node (partition 9)
DELETE A FROM t_item A, DELETED D WHERE A.i_id = D.i_id
INSERT t_item SELECT * FROM INSERTED

--
-- remote node, partition 10
DELETE A FROM rmt10.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt10.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 11
DELETE A FROM rmt11.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt11.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 12
DELETE A FROM rmt12.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt12.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 13
DELETE A FROM rmt13.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt13.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 14
DELETE A FROM rmt14.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt14.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 15
DELETE A FROM rmt15.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt15.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 16
DELETE A FROM rmt16.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt16.tpcc.dbo.t_item SELECT * FROM INSERTED

```

```
end
```

item_iod10.sql

```

-- file 16201_to_18000\item_iod.sql

--add item instead-of-trigger for partition 10

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONS off
go

use tpcc
go

drop trigger iod_item
go

create trigger iod_item on item instead of update, insert, delete as
begin

--
-- remote node, partition 1
DELETE A FROM rmt1.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt1.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 2
DELETE A FROM rmt2.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt2.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 3
DELETE A FROM rmt3.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt3.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 4
DELETE A FROM rmt4.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt4.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 5
DELETE A FROM rmt5.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt5.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 6
DELETE A FROM rmt6.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt6.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 7
DELETE A FROM rmt7.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt7.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 8

```

```

DELETE A FROM rmt8.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt8.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 9
DELETE A FROM rmt9.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt9.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- local node (partition 10)
DELETE A FROM t_item A, DELETED D WHERE A.i_id = D.i_id
INSERT t_item SELECT * FROM INSERTED

--
-- remote node, partition 11
DELETE A FROM rmt11.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt11.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 12
DELETE A FROM rmt12.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt12.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 13
DELETE A FROM rmt13.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt13.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 14
DELETE A FROM rmt14.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt14.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 15
DELETE A FROM rmt15.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt15.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 16
DELETE A FROM rmt16.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt16.tpcc.dbo.t_item SELECT * FROM INSERTED

end

```

item_iot11.sql

```

-- file 18001_to_19800\item_iot.sql

--add item instead-of-trigger for partition 11

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONS off
go

use tpcc
go

```

```

drop trigger iot_item
go

create trigger iot_item on item instead of update, insert, delete as
begin

--
-- remote node, partition 1
DELETE A FROM rmt1.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt1.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 2
DELETE A FROM rmt2.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt2.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 3
DELETE A FROM rmt3.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt3.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 4
DELETE A FROM rmt4.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt4.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 5
DELETE A FROM rmt5.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt5.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 6
DELETE A FROM rmt6.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt6.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 7
DELETE A FROM rmt7.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt7.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 8
DELETE A FROM rmt8.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt8.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 9
DELETE A FROM rmt9.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt9.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 10
DELETE A FROM rmt10.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt10.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- local node (partition 11)

```

```

DELETE A FROM t_item A, DELETED D WHERE A.i_id = D.i_id
INSERT t_item SELECT * FROM INSERTED

--
-- remote node, partition 12
DELETE A FROM rmt12.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt12.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 13
DELETE A FROM rmt13.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt13.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 14
DELETE A FROM rmt14.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt14.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 15
DELETE A FROM rmt15.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt15.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 16
DELETE A FROM rmt16.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt16.tpcc.dbo.t_item SELECT * FROM INSERTED

end

```

item_iot12.sql

```

-- file 19801_to_21600\item_iot.sql

--add item instead-of-trigger for partition 12

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONS off
go

use tpcc
go

drop trigger iot_item
go

create trigger iot_item on item instead of update, insert, delete as
begin

--
-- remote node, partition 1
DELETE A FROM rmt1.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt1.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 2
DELETE A FROM rmt2.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt2.tpcc.dbo.t_item SELECT * FROM INSERTED

```

```

--
-- remote node, partition 3
DELETE A FROM rmt3.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt3.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 4
DELETE A FROM rmt4.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt4.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 5
DELETE A FROM rmt5.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt5.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 6
DELETE A FROM rmt6.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt6.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 7
DELETE A FROM rmt7.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt7.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 8
DELETE A FROM rmt8.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt8.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 9
DELETE A FROM rmt9.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt9.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 10
DELETE A FROM rmt10.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt10.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 11
DELETE A FROM rmt11.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt11.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- local node (partition 12)
DELETE A FROM t_item A, DELETED D WHERE A.i_id = D.i_id
INSERT t_item SELECT * FROM INSERTED

--
-- remote node, partition 13
DELETE A FROM rmt13.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt13.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 14

```



```

DELETE A FROM rmt14.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt14.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 15
DELETE A FROM rmt15.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt15.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 16
DELETE A FROM rmt16.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt16.tpcc.dbo.t_item SELECT * FROM INSERTED

end

```

item_iot13.sql

```

-- file 21601_to_23400\item_iot.sql

--add item instead-of-trigger for partition 13

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONSOFF
go

use tpcc
go

drop trigger iot_item
go

create trigger iot_item on item instead of update, insert, delete as
begin

--
-- remote node, partition 1
DELETE A FROM rmt1.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt1.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 2
DELETE A FROM rmt2.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt2.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 3
DELETE A FROM rmt3.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt3.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 4
DELETE A FROM rmt4.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt4.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 5
DELETE A FROM rmt5.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt5.tpcc.dbo.t_item SELECT * FROM INSERTED

```

```

--
-- remote node, partition 6
DELETE A FROM rmt6.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt6.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 7
DELETE A FROM rmt7.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt7.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 8
DELETE A FROM rmt8.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt8.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 9
DELETE A FROM rmt9.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt9.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 10
DELETE A FROM rmt10.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt10.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 11
DELETE A FROM rmt11.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt11.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 12
DELETE A FROM rmt12.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt12.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- local node (partition 13)
DELETE A FROM t_item A, DELETED D WHERE A.i_id = D.i_id
INSERT t_item SELECT * FROM INSERTED

--
-- remote node, partition 14
DELETE A FROM rmt14.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt14.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 15
DELETE A FROM rmt15.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt15.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 16
DELETE A FROM rmt16.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt16.tpcc.dbo.t_item SELECT * FROM INSERTED

end

```

item_iot14.sql

```
-- file 23401_to_25200\item_iot.sql

--add item instead-of-trigger for partition 14

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONSOFF
go

use tpcc
go

drop trigger iot_item
go

create trigger iot_item on item instead of update, insert, delete as
begin

--
-- remote node, partition 1
DELETE A FROM rmt1.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt1.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 2
DELETE A FROM rmt2.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt2.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 3
DELETE A FROM rmt3.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt3.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 4
DELETE A FROM rmt4.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt4.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 5
DELETE A FROM rmt5.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt5.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 6
DELETE A FROM rmt6.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt6.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 7
DELETE A FROM rmt7.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt7.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 8
DELETE A FROM rmt8.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt8.tpcc.dbo.t_item SELECT * FROM INSERTED
```

```
--
-- remote node, partition 9
DELETE A FROM rmt9.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt9.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 10
DELETE A FROM rmt10.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt10.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 11
DELETE A FROM rmt11.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt11.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 12
DELETE A FROM rmt12.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt12.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 13
DELETE A FROM rmt13.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt13.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- local node (partition 14)
DELETE A FROM t_item A, DELETED D WHERE A.i_id = D.i_id
INSERT t_item SELECT * FROM INSERTED

--
-- remote node, partition 15
DELETE A FROM rmt15.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt15.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 16
DELETE A FROM rmt16.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt16.tpcc.dbo.t_item SELECT * FROM INSERTED

end
```

item_iot15.sql

```
-- file 25201_to_27000\item_iot.sql

--add item instead-of-trigger for partition 15

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONSOFF
go

use tpcc
go

drop trigger iot_item
go
```

```

create trigger iot_item on item instead of update, insert, delete as
begin

--
-- remote node, partition 1
DELETE A FROM rmt1.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt1.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 2
DELETE A FROM rmt2.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt2.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 3
DELETE A FROM rmt3.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt3.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 4
DELETE A FROM rmt4.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt4.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 5
DELETE A FROM rmt5.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt5.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 6
DELETE A FROM rmt6.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt6.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 7
DELETE A FROM rmt7.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt7.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 8
DELETE A FROM rmt8.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt8.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 9
DELETE A FROM rmt9.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt9.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 10
DELETE A FROM rmt10.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt10.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 11
DELETE A FROM rmt11.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt11.tpcc.dbo.t_item SELECT * FROM INSERTED

```

```

--
-- remote node, partition 12
DELETE A FROM rmt12.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt12.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 13
DELETE A FROM rmt13.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt13.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 14
DELETE A FROM rmt14.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt14.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- local node (partition 15)
DELETE A FROM t_item A, DELETED D WHERE A.i_id = D.i_id
INSERT t_item SELECT * FROM INSERTED

--
-- remote node, partition 16
DELETE A FROM rmt16.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt16.tpcc.dbo.t_item SELECT * FROM INSERTED

end

```

item_iot16.sql

```

-- file 27001_to_28800\item_iot.sql

--add item instead-of-trigger for partition 16

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONS off
go

use tpcc
go

drop trigger iot_item
go

create trigger iot_item on item instead of update, insert, delete as
begin

--
-- remote node, partition 1
DELETE A FROM rmt1.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt1.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 2
DELETE A FROM rmt2.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt2.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 3

```

```

DELETE A FROM rmt3.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt3.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 4
DELETE A FROM rmt4.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt4.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 5
DELETE A FROM rmt5.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt5.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 6
DELETE A FROM rmt6.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt6.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 7
DELETE A FROM rmt7.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt7.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 8
DELETE A FROM rmt8.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt8.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 9
DELETE A FROM rmt9.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt9.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 10
DELETE A FROM rmt10.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt10.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 11
DELETE A FROM rmt11.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt11.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 12
DELETE A FROM rmt12.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt12.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 13
DELETE A FROM rmt13.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt13.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 14
DELETE A FROM rmt14.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id

```

```

INSERT rmt14.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- remote node, partition 15
DELETE A FROM rmt15.tpcc.dbo.t_item A, DELETED D WHERE
A.i_id = D.i_id
INSERT rmt15.tpcc.dbo.t_item SELECT * FROM INSERTED

--
-- local node (partition 16)
DELETE A FROM t_item A, DELETED D WHERE A.i_id = D.i_id
INSERT t_item SELECT * FROM INSERTED

end

```

backup.sql

```

-- File: BACKUP.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.30
-- Copyright Microsoft, 2000
-- Purpose: Creates backup of tpcc database

declare @startdate datetime
declare @enddate datetime
select @startdate = getdate()
select 'Start date:', convert(varchar(30),@startdate,9)

dump database tpcc to tpccback1, tpccback2, tpccback3, tpccback4
with init, stats = 1, blocksize = 4096

select @enddate = getdate()
select 'End date: ', convert(varchar(30),@enddate,9)
select 'Elapsed time (in seconds): ', datediff(second, @startdate,
@enddate)

go

```

backupdev.sql

```

-- File: BACKUPDEVB.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.30
-- Copyright Microsoft, 2000
-- Purpose: Creates tpcc database Backup Devices

use master
go

-- create backup devices

exec sp_addumpdevice 'disk','tpccback1','f:\tpccback1.dmp'
exec sp_addumpdevice 'disk','tpccback2','g:\tpccback2.dmp'
exec sp_addumpdevice 'disk','tpccback3','h:\tpccback3.dmp'
exec sp_addumpdevice 'disk','tpccback4','k:\tpccback4.dmp'
go

```

createdb.sql

```

-- File: CREATEDB.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.30
-- Copyright Microsoft, 2000
-- Purpose: Creates tpcc database and backup files for 10 warehouses

```

```

use master
go

-- Create temporary table for timing

if exists ( select name from sysobjects where name = 'tpcc_timer' )
drop table tpcc_timer
go

create table tpcc_timer
(
    start_date          char(30),
    end_date            char(30)
)

insert into tpcc_timer values (0,0)
go

-- Store starting time

update tpcc_timer
set start_date = (select convert(char(30), getdate(),9))
go

-- create main database files

CREATE DATABASE tpcc
ON PRIMARY
(
    NAME = MSSQL80_tpcc_root,
    FILENAME = "j:\MSSQL80_tpcc_root.mdf",
    SIZE = 8MB,
    FILEGROWTH = 0),
FILEGROUP misc_fg
(
    NAME = MSSQL80_tpcc_misc_1,
    FILENAME = "C:\devjp\misc_files_1\",
    SIZE = 4000MB,
    FILEGROWTH = 0),
(
    NAME = MSSQL80_tpcc_misc_2,
    FILENAME = "C:\devjp\misc_files_2\",
    SIZE = 4000MB,
    FILEGROWTH = 0),
(
    NAME = MSSQL80_tpcc_misc_3,
    FILENAME = "C:\devjp\misc_files_3\",
    SIZE = 4000MB,
    FILEGROWTH = 0),
(
    NAME = MSSQL80_tpcc_misc_4,
    FILENAME = "C:\devjp\misc_files_4\",
    SIZE = 4000MB,
    FILEGROWTH = 0),
(
    NAME = MSSQL80_tpcc_misc_5,
    FILENAME = "C:\devjp\misc_files_5\",
    SIZE = 4000MB,
    FILEGROWTH = 0),
(
    NAME = MSSQL80_tpcc_misc_6,
    FILENAME = "C:\devjp\misc_files_6\",
    SIZE = 4000MB,
    FILEGROWTH = 0),
(
    NAME = MSSQL80_tpcc_misc_7,
    FILENAME = "C:\devjp\misc_files_7\",
    SIZE = 4000MB,
    FILEGROWTH = 0),
(
    NAME = MSSQL80_tpcc_misc_8,
    FILENAME = "C:\devjp\misc_files_8\",
    SIZE = 4000MB,
    FILEGROWTH = 0),
(
    NAME = MSSQL80_tpcc_misc_9,
    FILENAME = "C:\devjp\misc_files_9\",

```

```

    SIZE = 4000MB,
    FILEGROWTH = 0),
    NAME = MSSQL80_tpcc_misc_10,
    FILENAME = "C:\devjp\misc_files_10\",
    SIZE = 4000MB,
    FILEGROWTH = 0),
    NAME = MSSQL80_tpcc_misc_11,
    FILENAME = "C:\devjp\misc_files_11\",
    SIZE = 4000MB,
    FILEGROWTH = 0),
    NAME = MSSQL80_tpcc_misc_12,
    FILENAME = "C:\devjp\misc_files_12\",
    SIZE = 4000MB,
    FILEGROWTH = 0),
    NAME = MSSQL80_tpcc_misc_13,
    FILENAME = "C:\devjp\misc_files_13\",
    SIZE = 4000MB,
    FILEGROWTH = 0),
    NAME = MSSQL80_tpcc_misc_14,
    FILENAME = "C:\devjp\misc_files_14\",
    SIZE = 4000MB,
    FILEGROWTH = 0),
FILEGROUP big_fg
(
    NAME = MSSQL80_tpcc_big_1,
    FILENAME = "C:\devjp\big_files_1\",
    SIZE = 8500MB,
    FILEGROWTH = 0),
(
    NAME = MSSQL80_tpcc_big_2,
    FILENAME = "C:\devjp\big_files_2\",
    SIZE = 8500MB,
    FILEGROWTH = 0),
(
    NAME = MSSQL80_tpcc_big_3,
    FILENAME = "C:\devjp\big_files_3\",
    SIZE = 8500MB,
    FILEGROWTH = 0),
(
    NAME = MSSQL80_tpcc_big_4,
    FILENAME = "C:\devjp\big_files_4\",
    SIZE = 8500MB,
    FILEGROWTH = 0),
(
    NAME = MSSQL80_tpcc_big_5,
    FILENAME = "C:\devjp\big_files_5\",
    SIZE = 8500MB,
    FILEGROWTH = 0),
(
    NAME = MSSQL80_tpcc_big_6,
    FILENAME = "C:\devjp\big_files_6\",
    SIZE = 8500MB,
    FILEGROWTH = 0),
(
    NAME = MSSQL80_tpcc_big_7,
    FILENAME = "C:\devjp\big_files_7\",
    SIZE = 8500MB,
    FILEGROWTH = 0),
(
    NAME = MSSQL80_tpcc_big_8,
    FILENAME = "C:\devjp\big_files_8\",
    SIZE = 8500MB,
    FILEGROWTH = 0),
(
    NAME = MSSQL80_tpcc_big_9,
    FILENAME = "C:\devjp\big_files_9\",
    SIZE = 8500MB,
    FILEGROWTH = 0),
(
    NAME = MSSQL80_tpcc_big_10,
    FILENAME = "C:\devjp\big_files_10\",
    SIZE = 8500MB,
    FILEGROWTH = 0),
(
    NAME = MSSQL80_tpcc_big_11,
    FILENAME = "C:\devjp\big_files_11\",
    SIZE = 8500MB,
    FILEGROWTH = 0),
(
    NAME = MSSQL80_tpcc_big_12,

```

```

        FILENAME      = "C:\devjp\big_files_12\",
        SIZE          = 8500MB,
        FILEGROWTH    = 0),
(
    NAME            = MSSQL80_tpcc_big_13,
    FILENAME        = "C:\devjp\big_files_13\",
    SIZE            = 8500MB,
    FILEGROWTH      = 0),
(
    NAME            = MSSQL80_tpcc_big_14,
    FILENAME        = "C:\devjp\big_files_14\",
    SIZE            = 8500MB,
    FILEGROWTH      = 0)
LOG ON
(
    NAME            = MSSQL80_tpcc_log,
    FILENAME        = "E:",
    SIZE            = 61000MB,
    FILEGROWTH      = 0)

go

-- Store ending time
update tpcc_timer
set end_date = (select convert(char(30), getdate(),9))
go

select "Elapsed time (in seconds): ", datediff(second,(select start_date
from tpcc_timer),(select end_date from tpcc_timer))

-- remove temporary table

if exists ( select name from sysobjects where name = 'tpcc_timer' )
drop table tpcc_timer
go

```

tables.sql

```

-- File: TABLES.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.30
-- Copyright Microsoft, 2000
-- Purpose: Creates TPC-C tables

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONS off
set QUOTED_IDENTIFIER off
SET ANSI_NULL_DFLT_ON off

go

use tpcc
go

-- Remove all existing TPC-C tables, views, etc.
--

if objectproperty(object_id('warehouse'), 'IsView') is not null
drop view warehouse
go
if objectproperty(object_id('district'), 'IsView') is not null
drop view district
go
if objectproperty(object_id('customer'), 'IsView') is not null
drop view customer
go
if objectproperty(object_id('history'), 'IsView') is not null
drop view history
go

```

```

if objectproperty(object_id('stock'), 'IsView') is not null
drop view stock
go
if objectproperty(object_id('orders'), 'IsView') is not null
drop view orders
go
if objectproperty(object_id('order_line'), 'IsView') is not null
drop view order_line
go
if objectproperty(object_id('new_order'), 'IsView') is not null
drop view new_order
go
if objectproperty(object_id('item'), 'IsView') is not null
drop view item
go

if objectproperty(object_id('t_warehouse'), 'IsTable') is not null
drop table t_warehouse
go
if objectproperty(object_id('t_district'), 'IsTable') is not null
drop table t_district
go
if objectproperty(object_id('t_customer'), 'IsTable') is not null
drop table t_customer
go
if objectproperty(object_id('t_history'), 'IsTable') is not null
drop table t_history
go
if objectproperty(object_id('t_new_order'), 'IsTable') is not null
drop table t_new_order
go
if objectproperty(object_id('t_orders'), 'IsTable') is not null
drop table t_orders
go
if objectproperty(object_id('t_order_line'), 'IsTable') is not null
drop table t_order_line
go
if objectproperty(object_id('t_item'), 'IsTable') is not null
drop table t_item
go
if objectproperty(object_id('t_stock'), 'IsTable') is not null
drop table t_stock
go

--
-- Create new tables
--

create table t_warehouse
(
    w_id int,
    w_name char(10),
    w_street_1 char(20),
    w_street_2 char(20),
    w_city char(20),
    w_state char(2),
    w_zip char(9),
    w_tax numeric(4,4),
    w_ytd numeric(12,2)
) on misc_fg
go

create table t_district
(
    d_id tinyint,

```

```

        d_w_id                int,
        d_name                char(10),
        d_street_1            char(20),
        d_street_2            char(20),
        d_city                char(20),
        d_state                char(2),
        d_zip                 char(9),
        d_tax                 numeric(4,4),
        d_ytd                 numeric(12,2),
        d_next_o_id          int
    ) on misc_fg
go

create table t_customer
(
    c_id                int,
    c_d_id              tinyint,
    c_w_id              int,
    c_first             char(16),
    c_middle            char(2),
    c_last              char(16),
    c_street_1         char(20),
    c_street_2         char(20),
    c_city              char(20),
    c_state             char(2),
    c_zip              char(9),
    c_phone             char(16),
    c_since             datetime,
    c_credit            char(2),
    c_credit_lim        numeric(12,2),
    c_discount          numeric(4,4),
    c_balance           numeric(12,2),
    c_ytd_payment       numeric(12,2),
    c_payment_cnt       smallint,
    c_delivery_cnt      smallint,
    c_data              char(500)
) on big_fg
go

create table t_history
(
    h_c_id                int,
    h_c_d_id              tinyint,
    h_c_w_id              int,
    h_d_id                tinyint,
    h_w_id                int,
    h_date                datetime,
    h_amount              numeric(6,2),
    h_data                char(24),
    constraint history_c1PK primary key clustered (h_w_id,
h_d_id, h_c_id, h_date)
) on misc_fg
go

create table t_new_order
(
    no_o_id                int,
    no_d_id                tinyint,
    no_w_id                int
) on misc_fg
go

create table t_orders
(
    o_id                int,
    o_d_id                tinyint,
    o_w_id                int,
    o_c_id                int,
    o_entry_d            datetime,
    o_carrier_id         tinyint,
    o_ol_cnt              tinyint,
    o_all_local           tinyint
) on misc_fg
go

create table t_order_line
(
    ol_o_id                int,
    ol_d_id                tinyint,
    ol_w_id                int,
    ol_number              tinyint,
    ol_i_id                int,
    ol_supply_w_id         int,
    ol_delivery_d          datetime,
    ol_quantity            smallint,
    ol_amount              numeric(6,2),
    ol_dist_info          char(24)
) on misc_fg
go

create table t_item
(
    i_id                int,
    i_im_id              int,
    i_name                char(24),
    i_price               numeric(5,2),
    i_data                char(50)
) on misc_fg
go

create table t_stock
(
    s_i_id                int,
    s_w_id                int,
    s_quantity            smallint,
    s_dist_01             char(24),
    s_dist_02             char(24),
    s_dist_03             char(24),
    s_dist_04             char(24),
    s_dist_05             char(24),
    s_dist_06             char(24),
    s_dist_07             char(24),
    s_dist_08             char(24),
    s_dist_09             char(24),
    s_dist_10             char(24),
    s_ytd                 int,
    s_order_cnt           smallint,
    s_remote_cnt          smallint,
    s_data                char(50)
) on big_fg
go

dbopt1.sql

-- File: DBOPT1.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.30
-- Copyright Microsoft, 2000
-- Purpose: Sets database options for data load

use master
go

exec sp_dboption tpcc,'select into/bulkcopy',true

```

```
exec sp_dboption tpcc,'trunc. log on chkpt.',true
exec sp_dboption tpcc,'torn page detection',false
go
```

```
use tpcc
go
```

```
checkpoint
go
```

dbopt2.sql

```
-- File:  DBOPT2_SHILOH.SQL
--      Microsoft TPC-C Benchmark Kit Ver. 4.30
--      Copyright Microsoft, 2000
-- Purpose: Resets database options after data load
```

```
use master
go
```

```
sp_dboption tpcc,'select ',false
go
```

```
sp_dboption tpcc,'trunc. ',false
go
```

```
use tpcc
go
```

```
checkpoint
go
```

```
sp_configure allow,1
go
```

```
reconfigure with override
go
```

```
/*          */
/* Set option values for user-defined indexes */
/*          */
```

```
sp_indexoption 't_customer', 'DisallowPageLocks', TRUE
go
sp_indexoption 't_district', 'DisallowPageLocks',TRUE
go
sp_indexoption 't_warehouse', 'DisallowPageLocks',
TRUE
go
sp_indexoption 't_stock', 'DisallowPageLocks',TRUE
go
sp_indexoption 't_order_line', 'DisallowRowLocks',
TRUE
go
sp_indexoption 't_orders', 'DisallowRowLocks',TRUE
go
sp_indexoption 't_new_order', 'DisallowRowLocks',
TRUE
go
sp_indexoption 't_item', 'DisallowRowLocks',TRUE
go
sp_indexoption 't_item', 'DisallowPageLocks',TRUE
go
```

```
Print ''
```

```
Print '*****'
Print 'Pre-specified Locking Hierarchy:'
Print ' Lockflag = 0 ==> No pre-specified hierarchy'
Print ' Lockflag = 1 ==> Lock at Page-level then Table-level'
Print ' Lockflag = 2 ==> Lock at Row-level then Table-level'
Print ' Lockflag = 3 ==> Lock at Table-level'
Print ''
```

```
select name,lockflags
from sysindexes
where object_id("t_warehouse") = id or
      object_id("t_district") = id or
      object_id("t_customer") = id or
      object_id("t_stock") = id or
      object_id("t_orders") = id or
      object_id("t_order_line") = id or
      object_id("t_history") = id or
      object_id("t_new_order") = id or
      object_id("t_item") = id
order by lockflags asc
go
```

```
sp_configure allow,0
go
```

```
reconfigure with override
go
```

```
exec sp_dboption tpcc, 'auto update statistics',
FALSE
exec sp_dboption tpcc, 'auto create statistics', FALSE
go
```

```
exec sp_tableoption "t_district", "pintable",true
exec sp_tableoption "t_warehouse", "pintable",true
exec sp_tableoption "t_new_order", "pintable",true
exec sp_tableoption "t_item", "pintable",true
go
```

idxcuscl.sql

```
-- File:  IDXCUSCL.SQL
--      Microsoft TPC-C Benchmark Kit Ver. 4.30
--      Copyright Microsoft, 2000
-- Purpose: Creates clustered index on customer table
```

```
set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONS off
set QUOTED_IDENTIFIER off
go
```

```
use tpcc
go
```

```
declare @startdate datetime
declare @enddate datetime
select @startdate = getdate()
select "Start date:", convert(varchar(30),@startdate,9)
```

```
if exists (select o.name from sysconstraints c, sysobjects o
where c.id = object_id('t_customer') and c.constid = o.id and o.name =
'customer_c1PK')
alter table t_customer drop constraint customer_c1PK
```

```
alter table t_customer add constraint customer_c1PK primary key
clustered (c_w_id, c_d_id, c_id)
```



```

        on big_fg

select @enddate = getdate()
select "End date: ", convert(varchar(30),@enddate,9)
select "Elapsed time (in seconds): ", datediff(second, @startdate,
@enddate)

go

```

idxcusnc.sql

```

-- File:  IDXCUSNC.SQL
--      Microsoft TPC-C Benchmark Kit Ver. 4.30
--      Copyright Microsoft, 2000
-- Purpose:  Creates non-clustered index on customer table

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONSOFF
set QUOTED_IDENTIFIER off
go

use tpcc
go

declare @startdate datetime
declare @enddate datetime
select @startdate = getdate()
select "Start date:", convert(varchar(30),@startdate,9)

if exists ( select name from sysindexes where name = 'customer_nc1' )
    drop index t_customer.customer_nc1

create unique nonclustered index customer_nc1 on t_customer(c_w_id,
c_d_id, c_last, c_first, c_id)
    on big_fg

select @enddate = getdate()
select "End date: ", convert(varchar(30),@enddate,9)
select "Elapsed time (in seconds): ", datediff(second, @startdate,
@enddate)

go

```

idxdiscl.sql

```

-- File:  IDXDISCL.SQL
--      Microsoft TPC-C Benchmark Kit Ver. 4.20
--      Copyright Microsoft, 2000
-- Purpose:  Creates clustered index on district table

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONSOFF
set QUOTED_IDENTIFIER off
go

use tpcc
go

declare @startdate datetime
declare @enddate datetime
select @startdate = getdate()
select "Start date:", convert(varchar(30),@startdate,9)

if exists (select o.name from sysconstraints c, sysobjects o

```

```

where c.id = object_id('t_district') and c.constid = o.id and o.name =
'district_c1PK' )
    alter table t_district drop constraint district_c1PK

alter table t_district add constraint district_c1PK primary key clustered
(d_w_id, d_id)
    with fillfactor=100
    on misc_fg

select @enddate = getdate()
select "End date: ", convert(varchar(30),@enddate,9)
select "Elapsed time (in seconds): ", datediff(second, @startdate,
@enddate)

```

go

idxitmcl.sql

```

-- File:  IDXITMCL.SQL
--      Microsoft TPC-C Benchmark Kit Ver. 4.30
--      Copyright Microsoft, 2000
-- Purpose:  Creates clustered index on item table

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONSOFF
set QUOTED_IDENTIFIER off
go

use tpcc
go

declare @startdate datetime
declare @enddate datetime
select @startdate = getdate()
select "Start date:", convert(varchar(30),@startdate,9)

if exists ( select name from sysindexes where name = 'item_c1' )
    drop index t_item.item_c1

create unique clustered index item_c1 on t_item(i_id)
    on misc_fg

select @enddate = getdate()
select "End date: ", convert(varchar(30),@enddate,9)
select "Elapsed time (in seconds): ", datediff(second, @startdate,
@enddate)

```

go

idxnodcl.sql

```

-- File:  IDXNODCL.SQL
--      Microsoft TPC-C Benchmark Kit Ver. 4.30
--      Copyright Microsoft, 2000
-- Purpose:  Creates clustered index on new_order table

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONSOFF
set QUOTED_IDENTIFIER off
go

use tpcc
go

declare @startdate datetime
declare @enddate datetime

```

```

select @startdate = getdate()
select "Start date:", convert(varchar(30),@startdate,9)

if exists (select o.name from sysconstraints c, sysobjects o
where c.id = object_id('t_new_order') and c.constid = o.id and o.name = 't_new_order_c1PK')
    alter table t_new_order drop constraint new_order_c1PK

alter table t_new_order add constraint new_order_c1PK primary key
clustered (no_w_id, no_d_id, no_o_id)
    on misc_fg

select @enddate = getdate()
select "End date: ", convert(varchar(30),@enddate,9)
select "Elapsed time (in seconds): ", datediff(second, @startdate, @enddate)

go

```

idxodlcl.sql

```

-- File:  IDXODLCL.SQL
--      Microsoft TPC-C Benchmark Kit Ver. 4.30
--      Copyright Microsoft, 2000
-- Purpose:  Creates clustered index on order_line table

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONSOFF
set QUOTED_IDENTIFIER off
go

use tpcc
go

declare @startdate datetime
declare @enddate datetime
select @startdate = getdate()
select "Start date:", convert(varchar(30),@startdate,9)

if exists (select o.name from sysconstraints c, sysobjects o
where c.id = object_id('t_order_line') and c.constid = o.id and o.name = 'order_line_c1PK')
    alter table t_order_line drop constraint order_line_c1PK

alter table t_order_line add constraint order_line_c1PK primary key
clustered (ol_w_id, ol_d_id, ol_o_id, ol_number)
    on misc_fg

select @enddate = getdate()
select "End date: ", convert(varchar(30),@enddate,9)
select "Elapsed time (in seconds): ", datediff(second, @startdate, @enddate)

go

```

idxordcl.sql

```

-- File:  IDXORDCL.SQL
--      Microsoft TPC-C Benchmark Kit Ver. 4.30
--      Copyright Microsoft, 2000
-- Purpose:  Creates clustered index on orders table

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONSOFF
set QUOTED_IDENTIFIER off
go

```

```

use tpcc
go

declare @startdate datetime
declare @enddate datetime
select @startdate = getdate()
select "Start date:", convert(varchar(30),@startdate,9)

if exists (select o.name from sysconstraints c, sysobjects o
where c.id = object_id('t_orders') and c.constid = o.id and o.name = 'orders_c1PK')
    alter table t_orders drop constraint orders_c1PK

alter table t_orders add constraint orders_c1PK primary key clustered
(o_w_id, o_d_id, o_id)
    on misc_fg

select @enddate = getdate()
select "End date: ", convert(varchar(30),@enddate,9)
select "Elapsed time (in seconds): ", datediff(second, @startdate, @enddate)

go

```

idxordnc.sql

```

-- File:  IDXORDNC.SQL
--      Microsoft TPC-C Benchmark Kit Ver. 4.30
--      Copyright Microsoft, 2000
-- Purpose:  Creates non-clustered index on orders table

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONSOFF
set QUOTED_IDENTIFIER off
go

use tpcc
go

declare @startdate datetime
declare @enddate datetime
select @startdate = getdate()
select "Start date:", convert(varchar(30),@startdate,9)

if exists ( select name from sysindexes where name = 'orders_nc1' )
    drop index t_orders.orders_nc1

create index orders_nc1 on t_orders(o_w_id, o_d_id, o_c_id, o_id)
    on misc_fg

select @enddate = getdate()
select "End date: ", convert(varchar(30),@enddate,9)
select "Elapsed time (in seconds): ", datediff(second, @startdate, @enddate)

go

```

idxstkcl.sql

```

-- File:  IDXSTKCL.SQL
--      Microsoft TPC-C Benchmark Kit Ver. 4.30
--      Copyright Microsoft, 2000
-- Purpose:  Creates clustered index on stock table

```

```

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONS off
set QUOTED_IDENTIFIER off
go

use tpcc
go

declare @startdate datetime
declare @enddate datetime
select @startdate = getdate()
select "Start date:", convert(varchar(30),@startdate,9)

if exists (select o.name from sysconstraints c, sysobjects o
where c.id = object_id('t_stock') and c.constid = o.id and o.name =
'stock_c1PK' )
    alter table t_stock drop constraint stock_c1PK

alter table t_stock add constraint stock_c1PK primary key clustered
(s_i_id, s_w_id)
on big_fg

select @enddate = getdate()
select "End date: ", convert(varchar(30),@enddate,9)
select "Elapsed time (in seconds): ", datediff(second, @startdate,
@enddate)

go

```

idxwarcl.sql

```

-- File:  IDXWARCL.SQL
--      Microsoft TPC-C Benchmark Kit Ver. 4.30
--      Copyright Microsoft, 2000
-- Purpose:  Creates clustered index on warehouse table

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONS off
set QUOTED_IDENTIFIER off
go

use tpcc
go

declare @startdate datetime
declare @enddate datetime
select @startdate = getdate()
select "Start date:", convert(varchar(30),@startdate,9)

if exists (select o.name from sysconstraints c, sysobjects o
where c.id = object_id('t_warehouse') and c.constid = o.id and o.name =
'warehouse_c1PK' )
    alter table t_warehouse drop constraint warehouse_c1PK

alter table t_warehouse add constraint warehouse_c1PK primary key
clustered (w_id)
with fillfactor=100
on misc_fg

select @enddate = getdate()
select "End date: ", convert(varchar(30),@enddate,9)
select "Elapsed time (in seconds): ", datediff(second, @startdate,
@enddate)

go

```

version.sql

```

-- File:  VERSION.SQL
--      Microsoft TPC-C Benchmark Kit Ver. 4.10
--      Copyright Microsoft, 1999
-- Purpose:  Returns version level of TPC-C stored procs
-- Note:  Always update the return value of this proc for
--      any interface changes or 'must have' bug fixes.

set ANSI_DEFAULTS on
set IMPLICIT_TRANSACTIONS off
go

use tpcc
go

if exists ( select name from sysobjects where name = 'tpcc_version' )
    drop procedure tpcc_version
go

create proc tpcc_version
as
declare @version char(8)

begin
    select @version = '4.10.000'
    select @version as 'Version'

end

go

```

Load Source Code

getargs.c

```

//      File:          GETARGS.C
//                      Microsoft TPC-C Kit Ver.
4.30
//                      Copyright Microsoft,
1996, 1997, 1998, 1999, 2000
//      Purpose:  Source file for command line processing

// Includes
#include "tpcc.h"

//=====
//
// Function name: GetArgsLoader
//
//=====

void GetArgsLoader(int argc, char **argv, TPCCLDR_ARGS *pargs)
{
    int      i;
    char    *ptr;

#ifdef DEBUG
    printf("[%ld]DBG: Entering GetArgsLoader()\n", (int)
GetCurrentThreadId());
#endif
}

```

```

/* init args struct with some useful values */
pargs->server = SERVER;
pargs->user = USER;
pargs->password = PASSWORD;
pargs->database = DATABASE;
pargs->batch = BATCH;
pargs->num_warehouses = UNDEF;
pargs->tables_all = TRUE;
pargs->table_item = FALSE;
pargs->table_warehouse = FALSE;
pargs->table_customer = FALSE;
pargs->table_orders = FALSE;
pargs->loader_res_file =
LOADER_RES_FILE;
pargs->pack_size = DEFLDPACKSIZE;
pargs->starting_warehouse =
DEF_STARTING_WAREHOUSE;
pargs->build_index =
BUILD_INDEX;
pargs->index_order =
INDEX_ORDER;
pargs->index_script_path =
INDEX_SCRIPT_PATH;
pargs->scale_down =
SCALE_DOWN;

/* check for zero command line args */
if ( argc == 1 )
    GetArgsLoaderUsage();

for ( i = 1; i < argc; ++i )
{
    if ( argv[i][0] != '-' && argv[i][0] != '/' )
    {
        printf("\nUnrecognized command");
        GetArgsLoaderUsage();
        exit(1);
    }

    ptr = argv[i];

    switch ( ptr[1] )
    {
        case 'h': /* Fall through */
        case 'H':
            GetArgsLoaderUsage();
            break;

        case 'D':
            pargs->database = ptr+2;
            break;

        case 'P':
            pargs->password = ptr+2;
            break;

        case 'S':
            pargs->server = ptr+2;
            break;

        case 'U':
            pargs->user = ptr+2;
            break;

        case 'b':
            pargs->batch =
atol(ptr+2);

```

```

break;
        case 'W':
            pargs->num_warehouses
            = atol(ptr+2);
            break;
        case 's':
            pargs->starting_warehouse = atol(ptr+2);
            break;
        case 't':
            {
                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 '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
            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");
    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) DEFLDPAKSIZE);
    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.
// 4.30
//                      Copyright Microsoft,
// 1996, 1997, 1998, 1999, 2000
// Purpose: Random number generation routines for
database loader

// Includes
#include "tpcc.h"
#include "math.h"

// Defines
#define A 16807
#define M 2147483647
#define Q 127773 /* M div A */
#define R 2836 /* M mod A */
#define Thread __declspec(thread)

// Globals
long Thread Seed = 0; /* thread local seed */

/*****
*****
* random -
* Implements a GOOD pseudo random number generator. This
generator *
* will/should? run the complete period before repeating. *
* Copied from:
* Random Numbers Generators: Good Ones Are Hard to Find.
* Communications of the ACM - October 1988 Volume 31
Number 10 *
* Machine Dependencies:
* long must be 2 ^ 31 - 1 or greater.
*
*

```

```

*****
*****/

/*****
*****
* seed - load the Seed value used in irand and drand. Should be used
before *
* first call to irand or drand. *
*****
*****/

void seed(long val)
{

#ifdef DEBUG
printf("[%ld]DBG: Entering seed(...\n", (int)
GetCurrentThreadId());
printf("Old Seed %ld New Seed %ld\n",Seed, val);
#endif

if ( val < 0 )
val = abs(val);

Seed = val;

}

/*****
*****
*
* irand - returns a 32 bit integer pseudo random number with a period
of *
* 1 to 2 ^ 32 - 1. *
*
* parameters: *
* none. *
*
* returns: *
* 32 bit integer - defined as long ( see above ). *
*
* side effects: *
* seed get recomputed. *
*****
*****/

long irand()
{
register long s; /* copy of seed */
register long test; /* test flag */
register long hi; /* tmp value for speed */
register long lo; /* tmp value for speed */

#ifdef DEBUG
printf("[%ld]DBG: Entering irand(...\n", (int)
GetCurrentThreadId());
#endif

s = Seed;
hi = s / Q;
lo = s % Q;

test = A * lo - R * hi;
if ( test > 0 )
Seed = test;
else
Seed = test + M;

return( Seed );
}

/*****
*****
*
* drand - returns a double pseudo random number between 0.0 and 1.0.
*
* See irand. *
*****
*****/

double drand()
{
#ifdef DEBUG
printf("[%ld]DBG: Entering drand(...\n", (int)
GetCurrentThreadId());
#endif

return( (double)irand() / 2147483647.0);

}

//=====
// Function : RandomNumber
//
// Description:
//=====

long RandomNumber(long lower, long upper)
{
long rand_num;

#ifdef DEBUG
printf("[%ld]DBG: Entering RandomNumber(...\n", (int)
GetCurrentThreadId());
#endif

if ( upper == lower ) /* pgd 08-13-96 perf enhancement */
return lower;

upper++;

if ( upper <= lower )
rand_num = upper;
else
rand_num = lower + irand() % (upper - lower);
/* pgd 08-13-96 perf enhancement */

#ifdef DEBUG
printf("[%ld]DBG: RandomNumber between %ld & %ld ==>
%ld\n",
(int)
GetCurrentThreadId(), lower, upper, rand_num);
#endif

return rand_num;
}

#if 0
//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.
4.30

```

```

// Copyright Microsoft,
1996, 1997, 1998, 1999, 2000
// Purpose: Source file for database loader string functions

// Includes
#include "tpcc.h"
#include <string.h>
#include <ctype.h>

//=====
//
// Function name: MakeAddress
//
//=====

void MakeAddress(char *street_1,
                char *street_2,
                char *city,
                char *state,
                char *zip)
{
#ifdef DEBUG
    printf("[%ld]DBG: Entering MakeAddress()\n", (int)
GetCurrentThreadId());
#endif

    MakeAlphaString (10, 20, ADDRESS_LEN, street_1);
    MakeAlphaString (10, 20, ADDRESS_LEN, street_2);
    MakeAlphaString (10, 20, ADDRESS_LEN, city);
    MakeAlphaString ( 2,  2, STATE_LEN, state);
    MakeZipNumberString( 9, 9, ZIP_LEN, zip);

#ifdef DEBUG
    printf("[%ld]DBG: MakeAddress: street_1: %s, street_2: %s, city:
%s, state: %s, zip: %s\n",
          (int) GetCurrentThreadId(), street_1,
street_2, city, state, zip);
#endif

    return;
}

//=====
//
// Function name: LastName
//
//=====

void LastName(int num,
             char *name)
{
    static char *n[] =
    {
        "BAR", "OUGHT", "ABLE", "PRI", "PRES",
        "ESE", "ANTI", "CALLY", "ATION", "EING"
    };

#ifdef DEBUG
    printf("[%ld]DBG: Entering LastName()\n", (int)
GetCurrentThreadId());

```

```

#endif

if ((num >= 0) && (num < 1000))
{
    strcpy(name, n[(num/100)%10]);
    strcat(name, n[(num/10)%10]);
    strcat(name, n[(num/1)%10]);

    if (strlen(name) < LAST_NAME_LEN)
    {
        PaddString(LAST_NAME_LEN,
name);
    }
    else
    {
        printf("\nError in LastName()... num <%ld> out
of range (0,999)\n", num);
        exit(-1);
    }

#ifdef DEBUG
    printf("[%ld]DBG: LastName: num = [%d] ==> [%d][%d][%d]\n",
(int) GetCurrentThreadId(), num,
num/100, (num/10)%10, num%10);
    printf("[%ld]DBG: LastName: String = %s\n", (int)
GetCurrentThreadId(), name);
#endif

    return;
}

//=====
//
// Function name: MakeAlphaString
//
//=====

//philipdu 08/13/96 Changed MakeAlphaString to use A-Z, a-z, and
0-9 in
//accordance with spec see below:
//The spec says:
//4.3.2.2 The notation random a-string [x .. y]
//(respectively, n-string [x .. y]) represents a string of random
alphanumeric
//(respectively, numeric) characters of a random length of minimum x,
maximum y,
//and mean (y+x)/2. Alphanumerics are A..Z, a..z, and 0..9. The only
other
//requirement is that the character set used "must be able to represent a
minimum
//of 128 different characters". We are using 8-bit chars, so this is a non
issue.
//It is completely unreasonable to stuff non-printing chars into the text
fields.
//-CLevine 08/13/96

int MakeAlphaString( int x, int y, int z, char *str)
{
    int len;
    int i;
    char cc = 'a';

```

```

        static char chArray[] =
"0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz";
        static int chArrayMax = 61;

#ifdef DEBUG
    printf("[%ld]DBG: Entering MakeAlphaString()\n", (int)
GetCurrentThreadId());
#endif

    len= RandomNumber(x, y);

    for (i=0; i<len; i++)
    {
        cc = chArray[RandomNumber(0, chArrayMax)];
        str[i] = cc;
    }

    if ( len < z )
        memset(str+len, ' ', z - len);
    str[z] = 0;

    return len;
}

//=====
//
// Function name: MakeOriginalAlphaString
//
//=====

int MakeOriginalAlphaString(int x,

int y,

int z,

char *str,

int

percent)
{
    int len;
    int val;
    int start;

#ifdef DEBUG
    printf("[%ld]DBG: Entering MakeOriginalAlphaString()\n", (int)
GetCurrentThreadId());
#endif

    // verify percentage is valid
    if ((percent < 0) || (percent > 100))
    {
        printf("MakeOriginalAlphaString: Invalid
percentage: %d\n", percent);
        exit(-1);
    }

    // verify string is at least 8 chars in length
    if ((x + y) <= 8)
    {
        printf("MakeOriginalAlphaString: string length
must be >= 8\n");
        exit(-1);
    }

```



```

// Make Alpha String
len = MakeAlphaString(x,y, z, str);

val = RandomNumber(1,100);
if (val <= percent)
{
    start = RandomNumber(0, len - 8);
    strncpy(str + start, "ORIGINAL", 8);
}

#ifdef DEBUG
printf("[%ld]DBG: MakeOriginalAlphaString: : %s\n",
      (int) GetCurrentThreadId(), str);
#endif

return strlen(str);
}

//=====
//
// Function name: MakeNumberString
//
//=====
int MakeNumberString(int x, int y, int z, char *str)
{
    char tmp[16];

    //MakeNumberString is always called
    MakeZipNumberString(16, 16, 16, string)

    memset(str, '0', 16);
    itoa(RandomNumber(0, 99999999), tmp, 10);
    memcpy(str, tmp, strlen(tmp));

    itoa(RandomNumber(0, 99999999), tmp, 10);
    memcpy(str+8, tmp, strlen(tmp));

    str[16] = 0;

return 16;
}

//=====
//
// Function name: MakeZipNumberString
//
//=====
int MakeZipNumberString(int x, int y, int z, char *str)
{
    char tmp[16];

    //MakeZipNumberString is always called
    MakeZipNumberString(9, 9, 9, string)

    strcpy(str, "000011111");

    itoa(RandomNumber(0, 9999), tmp, 10);
    memcpy(str, tmp, strlen(tmp));

return 9;
}

```

```

//=====
//
// Function name: InitString
//
//=====
void InitString(char *str, int len)
{
#ifdef DEBUG
printf("[%ld]DBG: Entering InitString()\n", (int)
GetCurrentThreadId());
#endif

    memset(str, ' ', len);
    str[len] = 0;
}

//=====
//
// Function name: InitAddress
//
// Description:
//
//=====
void InitAddress(char *street_1, char *street_2, char *city, char *state,
char *zip)
{
    memset(street_1, ' ', 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.
// 4.30
// Copyright Microsoft,
// 1996, 1997, 1998, 1999,2000
// Purpose: Source file for time functions
```

```
// Includes
#include "tpcc.h"
```

```
// Globals
static long start_sec;
```

```
=====
=====
//
```

```
// Function name: TimeNow
```

```
//
=====
=====
```

```
long TimeNow()
```

```
{
    long time_now;
    struct _timeb el_time;
```

```
#ifdef DEBUG
    printf("[%ld]DBG: Entering TimeNow()\n", (int)
GetCurrentThreadId());
#endif
```

```
    _ftime(&el_time);
```

```
    time_now = ((el_time.time - start_sec) * 1000) + el_time.millitm;
```

```
    return time_now;
```

```
}
```

tpcc.h

```
// File: TPCC.H
// Microsoft TPC-C Kit Ver.
// 4.30
// Copyright Microsoft,
// 1996, 1997, 1998, 1999, 2000
// Purpose: Header file for TPC-C database loader
```

```
// Build number of TPC Benchmark Kit
#define TPCKIT_VER "4.30"
```

```
// General headers
#include <windows.h>
#include <winbase.h>
#include <stdlib.h>
#include <stdio.h>
#include <process.h>
#include <stddef.h>
#include <stdarg.h>
#include <string.h>
```

```
#include <time.h>
#include <sys\timeb.h>
#include <sys\types.h>
```

```
// ODBC headers
#include <sql.h>
#include <sqlext.h>
#include <odbcss.h>
```

```
// General constants
#define MILLI 1000
#define FALSE 0
#define TRUE 1
#define UNDEF -1
#define MINPRINTASCII 32
#define MAXPRINTASCII 126
```

```
// Default environment constants
#define SERVER ""
#define DATABASE "tpcc"
#define USER "sa"
#define PASSWORD ""
```

```
// Default loader arguments
#define BATCH 10000
#define DEFLOADPACKSIZE 32768
#define LOADER_RES_FILE "logs\\load.out"
#define LOADER_NURAND_C 123
#define DEF_STARTING_WAREHOUSE 1
#define BUILD_INDEX 1
// build both data and indexes
#define INDEX_ORDER 1
// build indexes before load
#define SCALE_DOWN 0
// build a normal scale database
#define INDEX_SCRIPT_PATH "scripts"
```

```
typedef struct
```

```
{
    char *server;
    char *database;
    char *user;
    char *password;
    BOOL tables_all;
    // set if loading all tables
    BOOL table_item;
    // set if loading ITEM table specifically
    BOOL 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;
    long char loader_res_file;
    char *synch_servername;
    long case_sensitivity;
    long starting_warehouse;
    long build_index;
```

```

        long                index_order;
        long                scale_down;
        char
*index_script_path;
} 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   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
#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.
4.30
//                                Copyright Microsoft,
1996, 1997, 1998, 1999, 2000
//      Purpose:    Source file for TPC-C database loader

// Includes
#include "tpcc.h"
#include "search.h"

// Defines
#define MAXITEMS            100000
#define MAXITEMS_SCALE_DOWN        100
#define CUSTOMERS_PER_DISTRICT 3000
#define CUSTOMERS_SCALE_DOWN 30
#define DISTRICT_PER_WAREHOUSE 10
#define ORDERS_PER_DISTRICT 3000
#define ORDERS_SCALE_DOWN 30
#define MAX_CUSTOMER_THREADS 2
#define MAX_ORDER_THREADS 3
#define MAX_MAIN_THREADS 4

// Functions declarations

void HandleErrorDBC (SQLHDBC hdbc1);

void CheckSQL();
void CheckDataBase();

long NURand();
void LoadItem();
void LoadWarehouse();

void Stock();
void District();

void LoadCustomer();
void CustomerBufInit();
void CustomerBufLoad();
void LoadCustomerTable();
void LoadHistoryTable();

void LoadOrders();
void OrdersBufInit();
void OrdersBufLoad();
void LoadOrdersTable();
void LoadNewOrderTable();
void LoadOrderLineTable();
void GetPermutation();
void CheckForCommit();
void OpenConnections();
void BuildIndex();
void FormatDate ();

// Shared memory structures
typedef struct
{
    long        ol;
    long        ol_i_id;
    long        ol_supply_w_id;
    short       ol_quantity;
    double      ol_amount;
}

```

```

char    ol_dist_info[DIST_INFO_LEN+1];
char
ol_delivery_d[OL_DELIVERY_D_LEN+1];
} ORDER_LINE_STRUCT;

typedef struct
{
    long    o_id;
    short   o_d_id;
    long    o_w_id;
    long    o_c_id;
    short   o_carrier_id;
    short   o_ol_cnt;
    short   o_all_local;
    ORDER_LINE_STRUCT  o_ol[15];
} ORDERS_STRUCT;

typedef struct
{
    long                c_id;
    short               c_d_id;
    long                c_w_id;
    char
c_first[FIRST_NAME_LEN+1];
    char
c_middle[MIDDLE_NAME_LEN+1];
    char
c_last[LAST_NAME_LEN+1];
    char
c_street_1[ADDRESS_LEN+1];
    char
c_street_2[ADDRESS_LEN+1];
    char
c_city[ADDRESS_LEN+1];
    char                c_state[STATE_LEN+1];
    char                c_zip[ZIP_LEN+1];
    char
c_phone[PHONE_LEN+1];
    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
    short              c_ytd_payment;
    short
c_payment_cnt;
    short              c_delivery_cnt;
    char
c_data[C_DATA_LEN+1];
    double              h_amount;
    char
h_data[H_DATA_LEN+1];
} CUSTOMER_STRUCT;

typedef struct
{
    char
c_last[LAST_NAME_LEN+1];
    char
c_first[FIRST_NAME_LEN+1];
    long                c_id;
} CUSTOMER_SORT_STRUCT;

typedef struct
{
    long    time_start;
} LOADER_TIME_STRUCT;

// Global variables

char    szLastError[300];

HENV    henv;

HDBC    v_hdbc;
// for SQL Server version verification
HDBC    i_hdbc1;
// for ITEM table
HDBC    w_hdbc1;
// for WAREHOUSE, DISTRICT, STOCK
HDBC    c_hdbc1;
// for CUSTOMER
HDBC    c_hdbc2;
// for HISTORY
HDBC    o_hdbc1;
// for ORDERS
HDBC    o_hdbc2;
// for NEW-ORDER
HDBC    o_hdbc3;
// for ORDER-LINE
HSTMT    v_hstmt;
// for SQL Server version verification
HSTMT    i_hstmt1;
HSTMT    w_hstmt1;
HSTMT    c_hstmt1, c_hstmt2;
HSTMT    o_hstmt1, o_hstmt2, o_hstmt3;

ORDERS_STRUCT  orders_buf[ORDERS_PER_DISTRICT];
CUSTOMER_STRUCT
customer_buf[CUSTOMERS_PER_DISTRICT];
long    orders_rows_loaded;
long    new_order_rows_loaded;
long    order_line_rows_loaded;
long    history_rows_loaded;
long    customer_rows_loaded;
long    stock_rows_loaded;
long    district_rows_loaded;
long    item_rows_loaded;
long    warehouse_rows_loaded;
long    main_time_start;
long    main_time_end;
long    max_items;
long    customers_per_district;
long    orders_per_district;
long    first_new_order;
long    last_new_order;

TPCCLDR_ARGS  *aptr, args;

// support for log directory determined by environment
#define LOGBASE_MAXLEN 128
int    UseAltLogDir = 0;
char    LogBase[LOGBASE_MAXLEN];
char    LogFile[2*LOGBASE_MAXLEN];
#define LOGFILE_CREATE(name)
\
strcpy(LogFile, LogBase);
\
strcat(LogFile, name);

```

```

//=====
//
// Function name: main
//
//=====
=====

int main(int argc, char **argv)
{
    DWORD      dwThreadID[MAX_MAIN_THREADS];
    HANDLE      hThread[MAX_MAIN_THREADS];
    FILE        *fLoader;
    char        buffer[255];
    int         i;

    for (i=0; i<MAX_MAIN_THREADS;i++)
        hThread[i] = NULL;

    printf("\n*****
*");
    printf("\n*
*");
    printf("\n* Microsoft SQL Server
*");
    printf("\n*
*");
    printf("\n* TPC-C BENCHMARK KIT: Database loader
*");
    printf("\n* Version %s
*");
    TPCKIT_VER);
    printf("\n*
*");

    printf("\n*****
*\n");

    // process command line arguments

    aptr = &args;
    GetArgsLoader(argc, argv, aptr);

    // process environment variables
    if ( GetEnvironmentVariable("LOGBASE", LogBase,
LOGBASE_MAXLEN))
    {
        UseAltLogDir = 1;
        strcat(LogBase, "\\logs");
        printf("Will use %s for log files\n", LogBase);
    }
    else
        strcpy(LogBase, "logs");

    // verify database and tables exist before attempting to load

    CheckDataBase();

    printf("Build interface is ODBC.\n");

    if (aptr->build_index == 0)
        printf("Data load only - no index creation.\n");
    else
        printf("Data load and index creation.\n");

    if (aptr->index_order == 0)
        printf("Clustered indexes will be created after
bulk load.\n");
    else

```

```

        printf("Clustered indexes will be created before
bulk load.\n");

        // set database scale values
        if (aptr->scale_down == 1)
        {
            printf("*** Scaled Down Database ***\n");
            max_items = MAXITEMS_SCALE_DOWN;
            customers_per_district =
CUSTOMERS_SCALE_DOWN;
            orders_per_district =
ORDERS_SCALE_DOWN;
            first_new_order = 0;
            last_new_order = 30;
        }
        else
        {
            max_items = MAXITEMS;
            customers_per_district =
CUSTOMERS_PER_DISTRICT;
            orders_per_district =
ORDERS_PER_DISTRICT;
            first_new_order = 2100;
            last_new_order = 3000;
        }

        // open connections to SQL Server
        OpenConnections();

        // open file for loader results
        fLoader = fopen(aptr->loader_res_file, "w");

        if (fLoader == NULL)
        {
            printf("Error, loader result file open failed.");
            exit(-1);
        }

        // start loading data

        sprintf(buffer, "TPC-C load started for %ld
warehouses.\n", aptr->num_warehouses);

        printf("%s", buffer);
        fprintf(fLoader, "%s", buffer);

        main_time_start = (TimeNow() / MILLI);

        // start parallel load threads

        if (aptr->tables_all || aptr->table_item)
        {
            fprintf(fLoader, "\nStarting loader threads for:
item\n");

            hThread[0] = CreateThread(NULL,
0,
(LPTHREAD_START_ROUTINE)LoadItem,
NULL,
0,

```

```

        &dwThreadID[0]);
        if (hThread[0] == NULL)
        {
            printf("Error, failed in creating
creating thread = 0.\n");
            exit(-1);
        }
    }

    if (aptr->tables_all || aptr->table_warehouse)
    {
        fprintf(fLoader, "Starting loader threads for:
warehouse\n");

        hThread[1] = CreateThread(NULL,

0,
(LPTHREAD_START_ROUTINE)LoadWarehouse,
NULL,
0,
&dwThreadID[1]);

        if (hThread[1] == NULL)
        {
            printf("Error, failed in creating
creating thread = 1.\n");
            exit(-1);
        }
    }

    if (aptr->tables_all || aptr->table_customer)
    {
        fprintf(fLoader, "Starting loader threads for:
customer\n");

        hThread[2] = CreateThread(NULL,

0,
(LPTHREAD_START_ROUTINE)LoadCustomer,
NULL,
0,
&dwThreadID[2]);

        if (hThread[2] == NULL)
        {
            printf("Error, failed in creating
creating main thread = 2.\n");
            exit(-1);
        }
    }

    if (aptr->tables_all || aptr->table_orders)
    {
        fprintf(fLoader, "Starting loader threads for:
orders\n");

        hThread[3] = CreateThread(NULL,

```

```

0,
(LPTHREAD_START_ROUTINE)LoadOrders,
NULL,
0,
&dwThreadID[3]);

        if (hThread[3] == NULL)
        {
            printf("Error, failed in creating
creating main thread = 3.\n");
            exit(-1);
        }
    }

    // Wait for threads to finish...
    for (i=0; i<MAX_MAIN_THREADS; i++)
    {
        if (hThread[i] != NULL)
        {
            WaitForSingleObject( hThread[i],
INFINITE );
            CloseHandle(hThread[i]);
            hThread[i] = NULL;
        }
    }

    main_time_end = (TimeNow() / MILLD);

    sprintf(buffer, "\nTPC-C load completed successfully in %ld
minutes.\n",
            (main_time_end -
main_time_start)/60);

    printf("%s",buffer);
    fprintf(fLoader, "%s", buffer);

    fclose(fLoader);

    SQLFreeEnv(henv);

    exit(0);

    return 0;
}

//=====
//
// Function name: LoadItem
//
//=====

void LoadItem()
{
    long        i_id;
                long        i_im_id;
    char        i_name[I_NAME_LEN+1];
    double      i_price;
    char        i_data[I_DATA_LEN+1];
                char        name[20];
                long        time_start;

```

```

RETCODE          rc;
DBINT            rcint;
char             bcphint[128];

// Seed with unique number
seed(1);

printf("Loading item table...\n");

// if build index before load
if ((aptr->build_index == 1) && (aptr->index_order == 1))
    BuildIndex("idxitmcl");

InitString(i_name, I_NAME_LEN+1);
InitString(i_data, I_DATA_LEN+1);

sprintf(name, "%s..%s", aptr->database, "t_item");

LOGFILE_CREATE("\\item.err")
rc = bcp_init(i_hdbc1, name, NULL, LogFile, DB_IN);

if (rc != SUCCEEDED)
    HandleErrorDBC(i_hdbc1);

if ((aptr->build_index == 1) && (aptr->index_order == 1))
{
    sprintf(bcphint, "tablock, order (i_id),
ROWS_PER_BATCH = 100000, CHECK_CONSTRAINTS");
    rc = bcp_control(i_hdbc1, BCPHINTS, (void*)
bcphint);
    if (rc != 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);
    i_name);
    MakeAlphaString(14, 24, I_NAME_LEN,
    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()
{
    long    w_id;
    char    w_name[W_NAME_LEN+1];
    char    w_street_1[ADDRESS_LEN+1];
    char    w_street_2[ADDRESS_LEN+1];
    char    w_city[ADDRESS_LEN+1];
    char    w_state[STATE_LEN+1];
    char    w_zip[ZIP_LEN+1];
    double  w_tax;
    double  w_ytd;
    char    name[20];
    long    time_start;
    RETCODE rc;
    DBINT   rcint;
    char    bcphint[128];

    // Seed with unique number
    seed(aptr->starting_warehouse + 1);

    printf("Loading warehouse table...\n");

```

```

// if build index before load...
if ((aptr->build_index == 1) && (aptr->index_order == 1))
    BuildIndex("idxwarcl");

InitString(w_name, W_NAME_LEN+1);
InitAddress(w_street_1, w_street_2, w_city, w_state,
w_zip);

sprintf(name, "%s.%s", aptr->database, "t_warehouse");

LOGFILE_CREATE("\\whouse.err")
rc = bcp_init(w_hdbc1, name, NULL, LogFile, DB_IN);

if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

if ((aptr->build_index == 1) && (aptr->index_order == 1))
{
    sprintf(bcpint, "tablock, order (w_id),
ROWS_PER_BATCH = %d, CHECK_CONSTRAINTS",
aptr->num_warehouses);
    rc = bcp_control(w_hdbc1, BCPHINTS, (void*)
bcpint);
    if (rc != SUCCEED)
        HandleErrorDBC(w_hdbc1);
}

rc = bcp_bind(w_hdbc1, (BYTE *) &w_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 1);
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

rc = bcp_bind(w_hdbc1, (BYTE *) w_name, 0,
W_NAME_LEN, NULL, 0, 0, 2);
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

rc = bcp_bind(w_hdbc1, (BYTE *) w_street_1, 0,
ADDRESS_LEN, NULL, 0, 0, 3);
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

rc = bcp_bind(w_hdbc1, (BYTE *) w_street_2, 0,
ADDRESS_LEN, NULL, 0, 0, 4);
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

rc = bcp_bind(w_hdbc1, (BYTE *) w_city, 0,
ADDRESS_LEN, NULL, 0, 0, 5);
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

rc = bcp_bind(w_hdbc1, (BYTE *) w_state, 0,
STATE_LEN, NULL, 0, 0, 6);
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

rc = bcp_bind(w_hdbc1, (BYTE *) w_zip, 0, ZIP_LEN,
NULL, 0, 0, 7);
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

rc = bcp_bind(w_hdbc1, (BYTE *) &w_tax, 0,
SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 8);
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

```

```

rc = bcp_bind(w_hdbc1, (BYTE *) &w_ytd, 0,
SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 9);
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

time_start = (TimeNow() / MILLI);

warehouse_rows_loaded = 0;

for (w_id = (long)aptr->starting_warehouse; w_id <=
aptr->num_warehouses; w_id++)
{
    MakeAlphaString(6,10, W_NAME_LEN,
w_name);

    MakeAddress(w_street_1, w_street_2, w_city,
w_state, w_zip);

    w_tax = ((float)
RandomNumber(0L,2000L))/10000.00;

    w_ytd = 300000.00;

    rc = bcp_sendrow(w_hdbc1);
    if (rc != SUCCEED)
        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("idxwarcl");

stock_rows_loaded = 0;
district_rows_loaded = 0;

District();
Stock();
}

//=====
//
// Function : District
//
//=====

void District()
{
    short d_id;
    long d_w_id;
    char d_name[D_NAME_LEN+1];
    char d_street_1[ADDRESS_LEN+1];
    char d_street_2[ADDRESS_LEN+1];
    char d_city[ADDRESS_LEN+1];
    char d_state[STATE_LEN+1];

```



```

char    d_zip[ZIP_LEN+1];
double  d_tax;
double  d_ytd;
char    name[20];
long    d_next_o_id;
long    time_start;
int     w_id;
int     RETCODE    rc;
DBINT   rcint;
char    bcp hint[128];

// Seed with unique number
seed(aptr->starting_warehouse + 2);

printf("Loading district table...\n");

// build index before load
if ((aptr->build_index == 1) && (aptr->index_order == 1))
    BuildIndex("idxdiscl");

InitString(d_name, D_NAME_LEN+1);
InitAddress(d_street_1, d_street_2, d_city, d_state, d_zip);

sprintf(name, "%s..%s", aptr->database, "t_district");

LOGFILE_CREATE("\\district.err")
rc = bcp_init(w_hdbc1, name, NULL, LogFile, DB_IN);

if (rc != SUCCEED)
    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, CHECK_CONSTRAINTS",
(aptr->num_warehouses * 10));
    rc = bcp_control(w_hdbc1, BCPHINTS, (void*)
bcp hint);
    if (rc != SUCCEED)
        HandleErrorDBC(w_hdbc1);
}

rc = bcp_bind(w_hdbc1, (BYTE *) &d_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 1);
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

rc = bcp_bind(w_hdbc1, (BYTE *) &d_w_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 2);
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

rc = bcp_bind(w_hdbc1, (BYTE *) d_name, 0,
D_NAME_LEN, NULL, 0, 0, 3);
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

rc = bcp_bind(w_hdbc1, (BYTE *) d_street_1, 0,
ADDRESS_LEN, NULL, 0, 0, 4);
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

rc = bcp_bind(w_hdbc1, (BYTE *) d_street_2, 0,
ADDRESS_LEN, NULL, 0, 0, 5);
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

```

```

rc = bcp_bind(w_hdbc1, (BYTE *) d_city, 0,
ADDRESS_LEN, NULL, 0, 0, 6);
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

rc = bcp_bind(w_hdbc1, (BYTE *) d_state, 0,
STATE_LEN, NULL, 0, 0, 7);
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

rc = bcp_bind(w_hdbc1, (BYTE *) d_zip, 0, ZIP_LEN,
NULL, 0, 0, 8);
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

rc = bcp_bind(w_hdbc1, (BYTE *) &d_tax, 0,
SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 9);
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

rc = bcp_bind(w_hdbc1, (BYTE *) &d_ytd, 0,
SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 10);
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

rc = bcp_bind(w_hdbc1, (BYTE *) &d_next_o_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 11);
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

d_ytd = 30000.0;

d_next_o_id = orders_per_district+1;

time_start = (TimeNow()/ MILLI);

for (w_id = 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 != SUCCEED)
    HandleErrorDBC(w_hdbc1);

        district_rows_loaded++;
        CheckForCommit(w_hdbc1,
w_hstmt1, district_rows_loaded, "district", &time_start);
    }
}

rcint = bcp_done(w_hdbc1);
if (rcint < 0)
    HandleErrorDBC(w_hdbc1);

```

```

printf("Finished loading district table.\n");

// if build index after load...
if ((aptr->build_index == 1) && (aptr->index_order == 0))
    BuildIndex("idxdiscl");

return;
}

//=====
//
// Function : Stock
//
//=====

void Stock()
{
    long    s_i_id;
    long    s_w_id;
    short   s_quantity;
    char    s_dist_01[S_DIST_LEN+1];
    char    s_dist_02[S_DIST_LEN+1];
    char    s_dist_03[S_DIST_LEN+1];
    char    s_dist_04[S_DIST_LEN+1];
    char    s_dist_05[S_DIST_LEN+1];
    char    s_dist_06[S_DIST_LEN+1];
    char    s_dist_07[S_DIST_LEN+1];
    char    s_dist_08[S_DIST_LEN+1];
    char    s_dist_09[S_DIST_LEN+1];
    char    s_dist_10[S_DIST_LEN+1];
    long    s_ytd;
    short   s_order_cnt;
    short   s_remote_cnt;
    char    s_data[S_DATA_LEN+1];
    short   len;
    char    name[20];
    long    time_start;
    RETCODE rc;
    DBINT   rcint;
    char    bcphint[128];

    // Seed with unique number
    seed(aptr->starting_warehouse + 3);

    // if build index before load...
    if ((aptr->build_index == 1) && (aptr->index_order == 1))
        BuildIndex("idxstckl");

    sprintf(name, "%s.%s", aptr->database, "t_stock");

    LOGFILE_CREATE("\\stock.err")
    rc = bcp_init(w_hdbc1, name, NULL, LogFile, DB_IN);
    if (rc != SUCCEEDED)
        HandleErrorDBC(w_hdbc1);

    if ((aptr->build_index == 1) && (aptr->index_order == 1))
    {
        sprintf(bcphint, "tablock, order (s_i_id, s_w_id),
ROWS_PER_BATCH = %u, CHECK_CONSTRAINTS",
(aptr->num_warehouses * 100000));
        rc = bcp_control(w_hdbc1, BCPHINTS, (void*)
bcphint);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);
    }
}

```

```

        rc = bcp_bind(w_hdbc1, (BYTE *) &s_i_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 1);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        bcp_bind(w_hdbc1, (BYTE *) &s_w_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 2);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) &s_quantity, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 3);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_01, 0,
S_DIST_LEN, NULL, 0, 0, 4);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_02, 0,
S_DIST_LEN, NULL, 0, 0, 5);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_03, 0,
S_DIST_LEN, NULL, 0, 0, 6);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_04, 0,
S_DIST_LEN, NULL, 0, 0, 7);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_05, 0,
S_DIST_LEN, NULL, 0, 0, 8);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_06, 0,
S_DIST_LEN, NULL, 0, 0, 9);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_07, 0,
S_DIST_LEN, NULL, 0, 0, 10);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_08, 0,
S_DIST_LEN, NULL, 0, 0, 11);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_09, 0,
S_DIST_LEN, NULL, 0, 0, 12);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_10, 0,
S_DIST_LEN, NULL, 0, 0, 13);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) &s_ytd, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 14);

```

```

        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) &s_order_cnt, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 15);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) &s_remote_cnt, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 16);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) s_data, 0,
S_DATA_LEN, NULL, 0, 0, 17);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        s_ytd = s_order_cnt = s_remote_cnt = 0;

        time_start = (TimeNow() / MILLI);

        printf("...Loading stock table\n");

        for (s_i_id=1; s_i_id <= max_items; s_i_id++)
        {

                for (s_w_id = (long)aptr->starting_warehouse;
s_w_id <= aptr->num_warehouses; s_w_id++)
                {

                        s_quantity =
(short)RandomNumber(10L,100L);
                        len =
MakeAlphaString(24,24,S_DIST_LEN, s_dist_01);
                        len =
MakeAlphaString(24,24,S_DIST_LEN, s_dist_02);
                        len =
MakeAlphaString(24,24,S_DIST_LEN, s_dist_03);
                        len =
MakeAlphaString(24,24,S_DIST_LEN, s_dist_04);
                        len =
MakeAlphaString(24,24,S_DIST_LEN, s_dist_05);
                        len =
MakeAlphaString(24,24,S_DIST_LEN, s_dist_06);
                        len =
MakeAlphaString(24,24,S_DIST_LEN, s_dist_07);
                        len =
MakeAlphaString(24,24,S_DIST_LEN, s_dist_08);
                        len =
MakeAlphaString(24,24,S_DIST_LEN, s_dist_09);
                        len =
MakeAlphaString(24,24,S_DIST_LEN, s_dist_10);

                                len =
MakeOriginalAlphaString(26,50, S_DATA_LEN, s_data,10);

                                        rc = bcp_sendrow(w_hdbc1);
                                        if (rc != SUCCEEDED)

HandleErrorDBC(w_hdbc1);

                                                stock_rows_loaded++;
                                                CheckForCommit(w_hdbc1,
w_hstmt1, stock_rows_loaded, "stock", &time_start);

                }

        }

```

```

        rcint = bcp_done(w_hdbc1);
        if (rcint < 0)
            HandleErrorDBC(w_hdbc1);

        printf("Finished loading stock table.\n");

        SQLFreeStmt(w_hstmt1, SQL_DROP);
        SQLDisconnect(w_hdbc1);
        SQLFreeConnect(w_hdbc1);

        // if build index after load...
        if ((aptr->build_index == 1) && (aptr->index_order == 0))
            BuildIndex("idxstkcl");

        return;
    }

//=====
//
// Function : LoadCustomer
//
//=====

void LoadCustomer()
{
        LOADER_TIME_STRUCT customer_time_start;
        LOADER_TIME_STRUCT history_time_start;
        long w_id;

        short d_id;

        DWORD
dwThreadID[MAX_CUSTOMER_THREADS];
        HANDLE
hThread[MAX_CUSTOMER_THREADS];
        char name[20];

        RETCODE
rc;

        DBINT
rcint;

        char
bcphint[128];
        char
cmd[256];
        // SQLRETURN
rc_1;
        // SQLSMALLINT
reccum, MsgLen;
        // SQLCHAR
SqlState[6], Msg[SQL_MAX_MESSAGE_LENGTH];
        // SQLINTEGER
NativeError;

        // Seed with unique number
        seed(aptr->starting_warehouse + 4);

        printf("Loading customer and history tables...\n");

        // if build index before load...
        if ((aptr->build_index == 1) && (aptr->index_order == 1))
            BuildIndex("idxcuscl");

        // Initialize bulk copy
        sprintf(name, "%s.%s", aptr->database, "t_customer");

        LOGFILE_CREATE("\\customer.err")

```

```

rc = bcp_init(c_hdbc1, name, NULL, LogFile, DB_IN);
if (rc != 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, CHECK_CONSTRAINTS",
(aptr->num_warehouses * 30000));
    rc = bcp_control(c_hdbc1, BCPHINTS, (void*)
bcphint);
    if (rc != SUCCEED)
        HandleErrorDBC(c_hdbc1);
}

sprintf(name, "%s.%s", aptr->database, "t_history");

LOGFILE_CREATE("\\history.err")
rc = bcp_init(c_hdbc2, name, NULL, LogFile, DB_IN);
if (rc != SUCCEED)
    HandleErrorDBC(c_hdbc2);

sprintf(bcphint, "tablock, order (h_w_id, h_d_id, h_c_id,
h_date), ROWS_PER_BATCH = %u, CHECK_CONSTRAINTS",
(aptr->num_warehouses * 30000));
rc = bcp_control(c_hdbc2, BCPHINTS, (void*) bcphint);
if (rc != 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 = (long)aptr->starting_warehouse; w_id <=
aptr->num_warehouses; w_id++)
{
    for (d_id = 1; d_id <=
DISTRICT_PER_WAREHOUSE; d_id++)
    {
        CustomerBufLoad(d_id, w_id);

        // Start parallel loading threads here...

        // Start customer table thread

        printf("...Loading customer table for:
d_id = %d, w_id = %d\n", d_id, w_id);

        hThread[0] = CreateThread(NULL,

0,

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,

LoadHistoryTable,

&history_time_start,

0,

&dwThreadID[1]);

if (hThread[1] == NULL)
{
    printf("Error, failed in
creating creating thread = 1.\n");
    exit(-1);
}

WaitForSingleObject( hThread[0],
INFINITE );
WaitForSingleObject( hThread[1],
INFINITE );

if (CloseHandle(hThread[0]) ==
FALSE)
{
    printf("Error, failed in
closing customer thread handle with errno: %d\n", GetLastError());
}

if (CloseHandle(hThread[1]) ==
FALSE)
{
    printf("Error, failed in
closing history thread handle with errno: %d\n", GetLastError());
}

}

// flush the bulk connection
rcint = bcp_done(c_hdbc1);
if (rcint < 0)
    HandleErrorDBC(c_hdbc1);

rcint = bcp_done(c_hdbc2);
if (rcint < 0)
    HandleErrorDBC(c_hdbc2);

printf("Finished loading customer table.\n");

// if build index after load...
if ((aptr->build_index == 1) && (aptr->index_order == 0))

```

```

        BuildIndex("idxcuscl");

// build non-clustered index
if (aptr->build_index == 1)
    BuildIndex("idxcusnc");

// Output the NURAND used for the loader into C_FIRST
for C_ID = 1,
    // C_W_ID = 1, and C_D_ID = 1
    sprintf(cmd, "isql -S%s -U%s -P%s -d%s -e -Q\"update
t_customer set c_first = 'C_LOAD = %d' where c_id = 1 and c_w_id =
1 and c_d_id = 1\" > %s\\nurand_load.log",
        aptr->server,
        aptr->user,
        aptr->password,
        aptr->database,
        LOADER_NURAND_C,
        LogBase);

system(cmd);

SQLFreeStmt(c_hstmt1, SQL_DROP);
SQLDisconnect(c_hdbc1);
SQLFreeConnect(c_hdbc1);

SQLFreeStmt(c_hstmt2, SQL_DROP);
SQLDisconnect(c_hdbc2);
SQLFreeConnect(c_hdbc2);

return;
}

//=====
//
// Function : CustomerBufInit
//
//=====

void CustomerBufInit()
{
    int i;

    for (i=0;i<customers_per_district;i++)
    {
        customer_buf[i].c_id = 0;
        customer_buf[i].c_d_id = 0;
        customer_buf[i].c_w_id = 0;

        strcpy(customer_buf[i].c_first,"");
        strcpy(customer_buf[i].c_middle,"");
        strcpy(customer_buf[i].c_last,"");
        strcpy(customer_buf[i].c_street_1,"");
        strcpy(customer_buf[i].c_street_2,"");
        strcpy(customer_buf[i].c_city,"");
        strcpy(customer_buf[i].c_state,"");
        strcpy(customer_buf[i].c_zip,"");
        strcpy(customer_buf[i].c_phone,"");
        strcpy(customer_buf[i].c_credit,"");

        customer_buf[i].c_credit_lim = 0;
        customer_buf[i].c_discount = (float) 0;

```

```

// fix to avoid ODBC float to numeric
conversion problem.
// customer_buf[i].c_balance = 0;
strcpy(customer_buf[i].c_balance,"");

customer_buf[i].c_ytd_payment = 0;
customer_buf[i].c_payment_cnt = 0;
customer_buf[i].c_delivery_cnt = 0;

strcpy(customer_buf[i].c_data,"");

customer_buf[i].h_amount = 0;

strcpy(customer_buf[i].h_data,"");
    }
}

//=====
//
// Function : CustomerBufLoad
//
// Fills shared buffer for HISTORY and CUSTOMER
//=====

void CustomerBufLoad(int d_id, int w_id)
{
    long i;
    CUSTOMER_SORT_STRUCT
c[CUSTOMERS_PER_DISTRICT];

    for (i=0;i<customers_per_district;i++)
    {
        if (i < 1000)
            LastName(i, c[i].c_last);
        else
            LastName(NURand(255,0,999,LOADER_NURAND_C), c[i].c_last);

            MakeAlphaString(8,16,FIRST_NAME_LEN,
c[i].c_first);

            c[i].c_id = i+1;
    }

    printf("...Loading customer buffer for: d_id = %d, w_id =
%d\n",
        d_id, w_id);

    for (i=0;i<customers_per_district;i++)
    {
        customer_buf[i].c_d_id = d_id;
        customer_buf[i].c_w_id = w_id;
        customer_buf[i].h_amount = 10.0;

        customer_buf[i].c_ytd_payment = 10.0;

        customer_buf[i].c_payment_cnt = 1;
        customer_buf[i].c_delivery_cnt = 0;

```

```

// Generate CUSTOMER and HISTORY data

customer_buf[i].c_id = c[i].c_id;

strcpy(customer_buf[i].c_first, c[i].c_first);
strcpy(customer_buf[i].c_last, c[i].c_last);

customer_buf[i].c_middle[0] = 'O';
customer_buf[i].c_middle[1] = 'E';

MakeAddress(customer_buf[i].c_street_1,
customer_buf[i].c_street_2,
customer_buf[i].c_city,
customer_buf[i].c_state,
customer_buf[i].c_zip);

MakeNumberString(16, 16, PHONE_LEN,
customer_buf[i].c_phone);

if (RandomNumber(1L, 100L) > 10)
    customer_buf[i].c_credit[0] = 'G';
else
    customer_buf[i].c_credit[0] = 'B';
customer_buf[i].c_credit[1] = 'C';

customer_buf[i].c_credit_lim = 50000.0;
customer_buf[i].c_discount = ((float)
RandomNumber(0L, 5000L)) / 10000.0;

// fix to avoid ODBC float to numeric
conversion problem.
// customer_buf[i].c_balance = -10.0;
strcpy(customer_buf[i].c_balance, "-10.0");

MakeAlphaString(300, 500, C_DATA_LEN,
customer_buf[i].c_data);

// Generate HISTORY data
MakeAlphaString(12, 24, H_DATA_LEN,
customer_buf[i].h_data);
}
}

//=====
//
// Function : LoadCustomerTable
//
//=====

void LoadCustomerTable(LOADER_TIME_STRUCT
*customer_time_start)
{
    int i;
    long c_id;
    short c_d_id;
    long c_w_id;
    char c_first[FIRST_NAME_LEN+1];
    char c_middle[MIDDLE_NAME_LEN+1];
    char c_last[LAST_NAME_LEN+1];
    char c_street_1[ADDRESS_LEN+1];
    char c_street_2[ADDRESS_LEN+1];
    char c_city[ADDRESS_LEN+1];
    char c_state[STATE_LEN+1];

```

```

char c_zip[ZIP_LEN+1];
char c_phone[PHONE_LEN+1];
char c_credit[CREDIT_LEN+1];
double c_credit_lim;
double c_discount;

// fix to avoid ODBC float to numeric conversion problem.

// double c_balance;
char c_balance[6];

double c_ytd_payment;
short c_payment_cnt;
short c_delivery_cnt;
char c_data[C_DATA_LEN+1];
char c_since[C_SINCE_LEN+1];
RETCODE rc;

rc = bcp_bind(c_hdbc1, (BYTE *) &c_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 1);
if (rc != SUCCEED)
    HandleErrorDBC(c_hdbc1);

rc = bcp_bind(c_hdbc1, (BYTE *) &c_d_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 2);
if (rc != SUCCEED)
    HandleErrorDBC(c_hdbc1);

rc = bcp_bind(c_hdbc1, (BYTE *) &c_w_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 3);
if (rc != SUCCEED)
    HandleErrorDBC(c_hdbc1);

rc = bcp_bind(c_hdbc1, (BYTE *) c_first, 0, FIRST_NAME_LEN,
NULL, 0, 0, 4);
if (rc != SUCCEED)
    HandleErrorDBC(c_hdbc1);

rc = bcp_bind(c_hdbc1, (BYTE *) c_middle, 0,
MIDDLE_NAME_LEN, NULL, 0, 0, 5);
if (rc != SUCCEED)
    HandleErrorDBC(c_hdbc1);

rc = bcp_bind(c_hdbc1, (BYTE *) c_last, 0, LAST_NAME_LEN,
NULL, 0, 0, 6);
if (rc != SUCCEED)
    HandleErrorDBC(c_hdbc1);

rc = bcp_bind(c_hdbc1, (BYTE *) c_street_1, 0, ADDRESS_LEN,
NULL, 0, 0, 7);
if (rc != SUCCEED)
    HandleErrorDBC(c_hdbc1);

rc = bcp_bind(c_hdbc1, (BYTE *) c_street_2, 0, ADDRESS_LEN,
NULL, 0, 0, 8);
if (rc != SUCCEED)
    HandleErrorDBC(c_hdbc1);

rc = bcp_bind(c_hdbc1, (BYTE *) c_city, 0, ADDRESS_LEN,
NULL, 0, 0, 9);
if (rc != SUCCEED)
    HandleErrorDBC(c_hdbc1);

rc = bcp_bind(c_hdbc1, (BYTE *) c_state, 0, STATE_LEN, NULL,
0, 0, 10);
if (rc != SUCCEED)
    HandleErrorDBC(c_hdbc1);

```

```

rc = bcp_bind(c_hdbc1, (BYTE *) c_zip, 0, ZIP_LEN, NULL, 0, 0, 11);
    if (rc != SUCCEEDED)
        HandleErrorDBC(c_hdbc1);

rc = bcp_bind(c_hdbc1, (BYTE *) c_phone, 0, PHONE_LEN, NULL, 0, 0, 12);
    if (rc != SUCCEEDED)
        HandleErrorDBC(c_hdbc1);

rc = bcp_bind(c_hdbc1, (BYTE *) &c_since, 0, C_SINCE_LEN, NULL, 0, SQLCHARACTER, 13);
    if (rc != SUCCEEDED)
        HandleErrorDBC(c_hdbc1);

rc = bcp_bind(c_hdbc1, (BYTE *) c_credit, 0, CREDIT_LEN, NULL, 0, 0, 14);
    if (rc != SUCCEEDED)
        HandleErrorDBC(c_hdbc1);

rc = bcp_bind(c_hdbc1, (BYTE *) &c_credit_lim, 0, SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 15);
    if (rc != SUCCEEDED)
        HandleErrorDBC(c_hdbc1);

rc = bcp_bind(c_hdbc1, (BYTE *) &c_discount, 0, SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 16);
    if (rc != SUCCEEDED)
        HandleErrorDBC(c_hdbc1);

// fix to avoid ODBC float to numeric conversion problem.

// rc = bcp_bind(c_hdbc1, (BYTE *) &c_balance, 0, SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 17);
// if (rc != SUCCEEDED)
//     HandleErrorDBC(c_hdbc1);
rc = bcp_bind(c_hdbc1, (BYTE *) c_balance, 0, 5, NULL, 0, SQLCHARACTER, 17);
    if (rc != SUCCEEDED)
        HandleErrorDBC(c_hdbc1);

rc = bcp_bind(c_hdbc1, (BYTE *) &c_ytd_payment, 0, SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 18);
    if (rc != SUCCEEDED)
        HandleErrorDBC(c_hdbc1);

rc = bcp_bind(c_hdbc1, (BYTE *) &c_payment_cnt, 0, SQL_VARLEN_DATA, NULL, 0, SQLINT2, 19);
    if (rc != SUCCEEDED)
        HandleErrorDBC(c_hdbc1);

rc = bcp_bind(c_hdbc1, (BYTE *) &c_delivery_cnt, 0, SQL_VARLEN_DATA, NULL, 0, SQLINT2, 20);
    if (rc != SUCCEEDED)
        HandleErrorDBC(c_hdbc1);

rc = bcp_bind(c_hdbc1, (BYTE *) c_data, 0, 500, NULL, 0, 0, 21);
    if (rc != SUCCEEDED)
        HandleErrorDBC(c_hdbc1);

for (i = 0; i < customers_per_district; i++)
{
    c_id = customer_buf[i].c_id;
    c_d_id = customer_buf[i].c_d_id;
    c_w_id = customer_buf[i].c_w_id;

```

```

strcpy(c_first, customer_buf[i].c_first);
strcpy(c_middle, customer_buf[i].c_middle);
strcpy(c_last, customer_buf[i].c_last);
strcpy(c_street_1, customer_buf[i].c_street_1);
strcpy(c_street_2, customer_buf[i].c_street_2);
strcpy(c_city, customer_buf[i].c_city);
strcpy(c_state, customer_buf[i].c_state);
strcpy(c_zip, customer_buf[i].c_zip);
strcpy(c_phone, customer_buf[i].c_phone);
strcpy(c_credit, customer_buf[i].c_credit);

FormatDate(&c_since);

c_credit_lim = customer_buf[i].c_credit_lim;
c_discount = customer_buf[i].c_discount;

// fix to avoid ODBC float to numeric
conversion problem.
// c_balance = customer_buf[i].c_balance;
strcpy(c_balance, customer_buf[i].c_balance);

c_ytd_payment =
customer_buf[i].c_ytd_payment;
c_payment_cnt =
customer_buf[i].c_payment_cnt;
c_delivery_cnt =
customer_buf[i].c_delivery_cnt;

strcpy(c_data, customer_buf[i].c_data);

// Send data to server
rc = bcp_sendrow(c_hdbc1);
if (rc != SUCCEEDED)
    HandleErrorDBC(c_hdbc1);

customer_rows_loaded++;
CheckForCommit(c_hdbc1, c_hstmt1,
customer_rows_loaded, "customer",
&customer_time_start->time_start);
}

}

//=====
//
// Function : LoadHistoryTable
//
//=====

void LoadHistoryTable(LOADER_TIME_STRUCT
*history_time_start)
{
    int i;
    long c_id;
    short c_d_id;
    long c_w_id;
    double h_amount;
    char h_data[H_DATA_LEN+1];
    char h_date[H_DATE_LEN+1];
    RETCODE rc;

rc = bcp_bind(c_hdbc2, (BYTE *) &c_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 1);
    if (rc != SUCCEEDED)

```

```

        HandleErrorDBC(c_hdbc2);

    rc = bcp_bind(c_hdbc2, (BYTE *) &c_d_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 2);
    if (rc != SUCCEED)
        HandleErrorDBC(c_hdbc2);

    rc = bcp_bind(c_hdbc2, (BYTE *) &c_w_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 3);
    if (rc != SUCCEED)
        HandleErrorDBC(c_hdbc2);

    rc = bcp_bind(c_hdbc2, (BYTE *) &c_d_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 4);
    if (rc != SUCCEED)
        HandleErrorDBC(c_hdbc2);

    rc = bcp_bind(c_hdbc2, (BYTE *) &c_w_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 5);
    if (rc != SUCCEED)
        HandleErrorDBC(c_hdbc2);

    rc = bcp_bind(c_hdbc2, (BYTE *) &h_date, 0,
H_DATE_LEN, NULL, 0, SQLCHARACTER, 6);
    if (rc != SUCCEED)
        HandleErrorDBC(c_hdbc2);

    rc = bcp_bind(c_hdbc2, (BYTE *) &h_amount, 0,
SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 7);
    if (rc != SUCCEED)
        HandleErrorDBC(c_hdbc2);

    rc = bcp_bind(c_hdbc2, (BYTE *) h_data, 0, H_DATA_LEN,
NULL, 0, 0, 8);
    if (rc != SUCCEED)
        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 != SUCCEED)
            HandleErrorDBC(c_hdbc2);

        history_rows_loaded++;
        CheckForCommit(c_hdbc2, c_hstmt2,
history_rows_loaded, "history", &history_time_start->time_start);
    }
}

//=====
//
// Function : LoadOrders
//
//=====

```

```

void LoadOrders()
{
    LOADER_TIME_STRUCT  orders_time_start;
    LOADER_TIME_STRUCT  new_order_time_start;
    LOADER_TIME_STRUCT  order_line_time_start;
    long                 w_id;

    short                d_id;
    DWORD
dwThreadID[MAX_ORDER_THREADS];
    HANDLE
hThread[MAX_ORDER_THREADS];
    char                 name[20];
    RETCODE
rc;
    char
bcphint[128];

    // seed with unique number
    seed(aptr->starting_warehouse + 5);

    printf("Loading orders...\n");

    // if build index before load...
    if ((aptr->build_index == 1) && (aptr->index_order == 1))
    {
        BuildIndex("idxordcl");
        BuildIndex("idxnodcl");
        BuildIndex("idxodcl");
    }

    // initialize bulk copy
    sprintf(name, "%s..%s", aptr->database, "t_orders");

    LOGFILE_CREATE("\\orders.err")
    rc = bcp_init(o_hdbc1, name, NULL, LogFile, DB_IN);
    if (rc != SUCCEED)
        HandleErrorDBC(o_hdbc1);

    if ((aptr->build_index == 1) && (aptr->index_order == 1))
    {
        sprintf(bcphint, "tablock, order (o_w_id, o_d_id,
o_id), ROWS_PER_BATCH = %u, CHECK_CONSTRAINTS",
(aptr->num_warehouses * 30000));
        rc = bcp_control(o_hdbc1, BCPHINTS, (void*)
bcphint);
        if (rc != SUCCEED)
            HandleErrorDBC(o_hdbc1);
    }

    sprintf(name, "%s..%s", aptr->database, "t_new_order");

    LOGFILE_CREATE("\\neword.err")
    rc = bcp_init(o_hdbc2, name, NULL, LogFile, DB_IN);
    if (rc != SUCCEED)
        HandleErrorDBC(o_hdbc2);

    if ((aptr->build_index == 1) && (aptr->index_order == 1))
    {
        sprintf(bcphint, "tablock, order (no_w_id,
no_d_id, no_o_id), ROWS_PER_BATCH = %u,
CHECK_CONSTRAINTS", (aptr->num_warehouses * 9000));
        rc = bcp_control(o_hdbc2, BCPHINTS, (void*)
bcphint);
        if (rc != SUCCEED)
            HandleErrorDBC(o_hdbc2);
    }

    sprintf(name, "%s..%s", aptr->database, "t_order_line");

```



```

LOGFILE_CREATE("\\ordline.err")
rc = bcp_init(o_hdbc3, name, NULL, LogFile, DB_IN);
if (rc != SUCCEED)
    HandleErrorDBC(o_hdbc3);

if ((aptr->build_index == 1) && (aptr->index_order == 1))
{
    sprintf(bcp_hint, "tablock, order (ol_w_id,
ol_d_id, ol_o_id, ol_number), ROWS_PER_BATCH = %u,
CHECK_CONSTRAINTS", (aptr->num_warehouses * 300000));
rc = bcp_control(o_hdbc3, BCPHINTS, (void*)
bcp_hint);
    if (rc != SUCCEED)
        HandleErrorDBC(o_hdbc3);
}

orders_rows_loaded = 0;
new_order_rows_loaded = 0;
order_line_rows_loaded = 0;

OrdersBufInit();

orders_time_start.time_start = (TimeNow() / MILLI);
new_order_time_start.time_start = (TimeNow() / MILLI);
order_line_time_start.time_start = (TimeNow() / MILLI);

for (w_id = (long)aptr->starting_warehouse; w_id <=
aptr->num_warehouses; w_id++)
{
    for (d_id = 1; d_id <=
DISTRICT_PER_WAREHOUSE; d_id++)
    {
        OrdersBufLoad(d_id, w_id);

        // start parallel loading threads here...
        // start Orders table thread
        printf("...Loading Order Table for:
d_id = %d, w_id = %d\n", d_id, w_id);

        hThread[0] = CreateThread(NULL,
0,
(LPTHREAD_START_ROUTINE)
LoadOrdersTable,
&orders_time_start,
0,
&dwThreadID[0]);

        if (hThread[0] == NULL)
        {
            printf("Error, failed in
creating creating thread = 0.\n");
            exit(-1);
        }

        // start NewOrder table thread
        printf("...Loading New-Order Table
for: d_id = %d, w_id = %d\n", d_id, w_id);

```

```

        hThread[1] = CreateThread(NULL,
0,
(LPTHREAD_START_ROUTINE)
LoadNewOrderTable,
&new_order_time_start,
0,
&dwThreadID[1]);

        if (hThread[1] == NULL)
        {
            printf("Error, failed in
creating creating thread = 1.\n");
            exit(-1);
        }

        // start Order-Line table thread
        printf("...Loading Order-Line Table
for: d_id = %d, w_id = %d\n", d_id, w_id);

        hThread[2] = CreateThread(NULL,
0,
(LPTHREAD_START_ROUTINE)
LoadOrderLineTable,
&order_line_time_start,
0,
&dwThreadID[2]);

        if (hThread[2] == NULL)
        {
            printf("Error, failed in
creating creating thread = 2.\n");
            exit(-1);
        }

        WaitForSingleObject( hThread[0],
INFINITE );
        WaitForSingleObject( hThread[1],
INFINITE );
        WaitForSingleObject( hThread[2],
INFINITE );

        if (CloseHandle(hThread[0]) ==
FALSE)
        {
            printf("Error, failed in
closing Orders thread handle with errno: %d\n", GetLastError());
        }

        if (CloseHandle(hThread[1]) ==
FALSE)
        {
            printf("Error, failed in
closing NewOrder thread handle with errno: %d\n", GetLastError());
        }

        if (CloseHandle(hThread[2]) ==
FALSE)

```

```

        {
            printf("Error, failed in
closing OrderLine thread handle with errno: %d\n", GetLastError());
        }
    }
}

printf("Finished loading orders.\n");

return;
}

//=====
//
// Function : OrdersBufInit
//
// Clears shared buffer for ORDERS, NEWORDER, and ORDERLINE
//
//=====
void OrdersBufInit()
{
    int i;
    int j;

    for (i=0;i<orders_per_district;i++)
    {
        orders_buf[i].o_id = 0;
        orders_buf[i].o_d_id = 0;
        orders_buf[i].o_w_id = 0;
        orders_buf[i].o_c_id = 0;
        orders_buf[i].o_carrier_id = 0;
        orders_buf[i].o_ol_cnt = 0;
        orders_buf[i].o_all_local = 0;

        for (j=0;j<=14;j++)
        {
            orders_buf[i].o_ol[j].ol = 0;
            orders_buf[i].o_ol[j].ol_i_id = 0;

            orders_buf[i].o_ol[j].ol_supply_w_id
= 0;
            orders_buf[i].o_ol[j].ol_quantity = 0;
            orders_buf[i].o_ol[j].ol_amount = 0;

            strcpy(orders_buf[i].o_ol[j].ol_dist_info,"");
        }
    }

//=====
//
// Function : OrdersBufLoad
//
// Fills shared buffer for ORDERS, NEWORDER, and ORDERLINE
//
//=====

```

```

void OrdersBufLoad(int d_id, int w_id)
{
    int cust[ORDERS_PER_DISTRICT+1];
    long o_id;
    short ol;

    printf("...Loading Order Buffer for: d_id = %d, w_id =
%d\n",
        d_id, w_id);

    GetPermutation(cust, orders_per_district);

    for (o_id=0;o_id<orders_per_district;o_id++)
    {
        // Generate ORDER and NEW-ORDER data

        orders_buf[o_id].o_d_id = d_id;
        orders_buf[o_id].o_w_id = w_id;
        orders_buf[o_id].o_id = o_id+1;
        orders_buf[o_id].o_c_id = cust[o_id+1];
        orders_buf[o_id].o_ol_cnt =
(short)RandomNumber(5L, 15L);

        if (o_id < first_new_order)
        {
            orders_buf[o_id].o_carrier_id =
(short)RandomNumber(1L, 10L);
            orders_buf[o_id].o_all_local = 1;
        }
        else
        {
            orders_buf[o_id].o_carrier_id = 0;
            orders_buf[o_id].o_all_local = 1;
        }

        for (ol=0; ol<orders_buf[o_id].o_ol_cnt; ol++)
        {
            orders_buf[o_id].o_ol[ol].ol = ol+1;
            orders_buf[o_id].o_ol[ol].ol_i_id =
RandomNumber(1L, max_items);
            orders_buf[o_id].o_ol[ol].ol_supply_w_id = w_id;
            orders_buf[o_id].o_ol[ol].ol_quantity
= 5;
            MakeAlphaString(24, 24,
OL_DIST_INFO_LEN, &orders_buf[o_id].o_ol[ol].ol_dist_info);

            // Generate ORDER-LINE data
            if (o_id < first_new_order)
            {

orders_buf[o_id].o_ol[ol].ol_amount = 0;
// Added to insure
ol_delivery_d set properly during load

FormatDate(&orders_buf[o_id].o_ol[ol].ol_delivery_d);

            }
            else
            {

```

```

orders_buf[o_id].o_ol[ol].ol_amount =
RandomNumber(1,999999)/100.0;
// Added to insure
ol_delivery_d set properly during load

// odbc datetime format
strcpy(orders_buf[o_id].o_ol[ol].ol_delivery_d,"1899-12-31
00:00:00.000");
}
}
}

//=====
//
// Function : LoadOrdersTable
//
//=====

void LoadOrdersTable(LOADER_TIME_STRUCT*orders_time_start)
{
    int i;
    long o_id;
    short o_d_id;
    long o_w_id;
    long o_c_id;
    short o_carrier_id;
    short o_ol_cnt;
    short o_all_local;
    char o_entry_d[O_ENTRY_D_LEN+1];
    RETCODE rc;
    DBINT rcint;

    // bind ORDER data
    rc = bcp_bind(o_hdbc1, (BYTE *) &o_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 1);
    if (rc != SUCCEED)
        HandleErrorDBC(o_hdbc1);

    rc = bcp_bind(o_hdbc1, (BYTE *) &o_d_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 2);
    if (rc != SUCCEED)
        HandleErrorDBC(o_hdbc1);

    rc = bcp_bind(o_hdbc1, (BYTE *) &o_w_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 3);
    if (rc != SUCCEED)
        HandleErrorDBC(o_hdbc1);

    rc = bcp_bind(o_hdbc1, (BYTE *) &o_c_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 4);
    if (rc != SUCCEED)
        HandleErrorDBC(o_hdbc1);

    rc = bcp_bind(o_hdbc1, (BYTE *) &o_entry_d, 0,
O_ENTRY_D_LEN, NULL, 0, SQLCHARACTER, 5);
    if (rc != SUCCEED)
        HandleErrorDBC(o_hdbc1);

    rc = bcp_bind(o_hdbc1, (BYTE *) &o_carrier_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 6);
    if (rc != SUCCEED)

```

```

        HandleErrorDBC(o_hdbc1);

    rc = bcp_bind(o_hdbc1, (BYTE *) &o_ol_cnt, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 7);
    if (rc != SUCCEED)
        HandleErrorDBC(o_hdbc1);

    rc = bcp_bind(o_hdbc1, (BYTE *) &o_all_local, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 8);
    if (rc != SUCCEED)
        HandleErrorDBC(o_hdbc1);

    for (i = 0; i < orders_per_district; i++)
    {
        o_id = orders_buf[i].o_id;
        o_d_id = orders_buf[i].o_d_id;
        o_w_id = orders_buf[i].o_w_id;
        o_c_id = orders_buf[i].o_c_id;
        o_carrier_id = orders_buf[i].o_carrier_id;
        o_ol_cnt = orders_buf[i].o_ol_cnt;
        o_all_local = orders_buf[i].o_all_local;

        FormatDate(&o_entry_d);

        // send data to server
        rc = bcp_sendrow(o_hdbc1);
        if (rc != SUCCEED)
            HandleErrorDBC(o_hdbc1);

        orders_rows_loaded++;
        CheckForCommit(o_hdbc1, o_hstmt1,
orders_rows_loaded, "orders", &orders_time_start->time_start);
    }

    // rcint = bcp_batch(o_hdbc1);
    // if (rcint < 0)
    //     HandleErrorDBC(o_hdbc1);

    if ((o_w_id == aptr->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 ((aptr->build_index == 1) &&
(aptr->index_order == 0))
            BuildIndex("idxordc1");

        // build non-clustered index
        if (aptr->build_index == 1)
            BuildIndex("idxordnc");
    }
}

//=====
//
// Function : LoadNewOrderTable
//

```

```

//=====
//=====
void LoadNewOrderTable(LOADER_TIME_STRUCT
*new_order_time_start)
{
    int    i;
    long   o_id;
    short  o_d_id;
    long   o_w_id;
    RETCODE rc;
    DBINT  rcint;

    // Bind NEW-ORDER data

    rc = bcp_bind(o_hdbc2, (BYTE *) &o_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 1);
    if (rc != 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, SQLINT4, 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    i,j;
    long   o_id;
    short  o_d_id;
    long   o_w_id;
    long   ol;
    long   ol_i_id;
    long   ol_supply_w_id;
    short  ol_quantity;
    double ol_amount;
    char   ol_dist_info[DIST_INFO_LEN+1];
    char   ol_delivery_d[OL_DELIVERY_D_LEN+1];
    RETCODE rc;
    DBINT  rcint;

    // bind ORDER-LINE data
    rc = bcp_bind(o_hdbc3, (BYTE *) &o_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 1);
    if (rc != 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, SQLINT4, 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, SQLINT4, 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 != SUCCEEDED)

                    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("idxodcl1");

```

```

        }
    }

//=====
//
// Function : GetPermutation
//
//=====

void GetPermutation(int perm[], int n)
{
    int i, r, t;

    for (i=1;i<=n;i++)
        perm[i] = i;

    for (i=1;i<=n;i++)
    {
        r = RandomNumber(i,n);
        t = perm[i];
        perm[i] = perm[r];
        perm[r] = t;
    }
}

//=====
//
// Function : CheckForCommit
//
//=====

void CheckForCommit(HDBC hdbc,
                    HSTMT hstmt,
                    long rows_loaded,
                    char
*table_name,
                    long *time_start)
{
    long time_end, time_diff;
    // DBINT rcint;

    if ( !(rows_loaded % aptr->batch) )
    {
        // rcint = bcp_batch(hdbc);
        // if (rcint < 0)
        //     HandleErrorDBC(hdbc);

        time_end = (TimeNow() / MILLI);
        time_diff = time_end - *time_start;

        printf("-> Loaded %ld rows into %s in %ld sec -
Total = %d (%.2f rps)\n",
                aptr->batch,
                table_name,
                time_diff,
                rows_loaded,

```

```

(float) aptr->batch /
(time_diff ? time_diff : 1L));

        *time_start = time_end;
    }

    return;
}

//=====
//
// Function : OpenConnections
//
//=====

void OpenConnections()
{
    RETCODE    rc;

    char        szDriverString[300];
    char        szDriverStringOut[1024];
    SQLSMALLINT cbDriverStringOut;

    SQLAllocHandle(SQL_HANDLE_ENV,
SQL_NULL_HANDLE, &henv );

    SQLSetEnvAttr(henv, SQL_ATTR_ODBC_VERSION,
(void*)SQL_OV_ODBC3, 0 );

    SQLAllocHandle(SQL_HANDLE_DBC, henv , &i_hdbc1);
    SQLAllocHandle(SQL_HANDLE_DBC, henv ,
&w_hdbc1);
    SQLAllocHandle(SQL_HANDLE_DBC, henv ,
&c_hdbc1);
    SQLAllocHandle(SQL_HANDLE_DBC, henv ,
&c_hdbc2);
    SQLAllocHandle(SQL_HANDLE_DBC, henv ,
&o_hdbc1);
    SQLAllocHandle(SQL_HANDLE_DBC, henv ,
&o_hdbc2);
    SQLAllocHandle(SQL_HANDLE_DBC, henv ,
&o_hdbc3);

    SQLSetConnectAttr(i_hdbc1, SQL_COPT_SS_BCP, (void
*)SQL_BCP_ON, SQL_IS_INTEGER);
    SQLSetConnectAttr(w_hdbc1, SQL_COPT_SS_BCP, (void
*)SQL_BCP_ON, SQL_IS_INTEGER);
    SQLSetConnectAttr(c_hdbc1, SQL_COPT_SS_BCP, (void
*)SQL_BCP_ON, SQL_IS_INTEGER);
    SQLSetConnectAttr(c_hdbc2, SQL_COPT_SS_BCP, (void
*)SQL_BCP_ON, SQL_IS_INTEGER);
    SQLSetConnectAttr(o_hdbc1, SQL_COPT_SS_BCP, (void
*)SQL_BCP_ON, SQL_IS_INTEGER);
    SQLSetConnectAttr(o_hdbc2, SQL_COPT_SS_BCP, (void
*)SQL_BCP_ON, SQL_IS_INTEGER);
    SQLSetConnectAttr(o_hdbc3, SQL_COPT_SS_BCP, (void
*)SQL_BCP_ON, SQL_IS_INTEGER);

    // Open connections to SQL Server

    // Connection 1

```

```

    sprintf( szDriverString , "DRIVER={SQL
Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s",
aptr->server,
aptr->user,
aptr->password,
aptr->database );

    rc = SQLSetConnectOption (i_hdbc1,
SQL_PACKET_SIZE, aptr->pack_size);
    if (rc != SUCCEED)
        HandleErrorDBC(i_hdbc1);

    rc = SQLDriverConnect ( i_hdbc1,
NULL,
(SQLCHAR*)&szDriverString[0],
SQL_NTS,
(SQLCHAR*)&szDriverStringOut[0],
sizeof(szDriverStringOut),
&cbDriverStringOut,
SQL_DRIVER_NOPROMPT);
    if (rc != SUCCEED)
        HandleErrorDBC(i_hdbc1);

    // Connection 2

    sprintf( szDriverString , "DRIVER={SQL
Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s",
aptr->server,
aptr->user,
aptr->password,
aptr->database );

    rc = SQLSetConnectOption (w_hdbc1,
SQL_PACKET_SIZE, aptr->pack_size);
    if (rc != SUCCEED)
        HandleErrorDBC(w_hdbc1);

    rc = SQLDriverConnect ( w_hdbc1,
NULL,
(SQLCHAR*)&szDriverString[0],
SQL_NTS,
(SQLCHAR*)&szDriverStringOut[0],
sizeof(szDriverStringOut),
&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->server,
aptr->user,
aptr->password,
aptr->database );

    rc = SQLSetConnectOption (c_hdbc1,
SQL_PACKET_SIZE, aptr->pack_size);
    if (rc != SUCCEEDED)
        HandleErrorDBC(c_hdbc1);

    rc = SQLDriverConnect ( c_hdbc1,
NULL,
(SQLCHAR*)&szDriverString[0],
SQL_NTS,
(SQLCHAR*)&szDriverStringOut[0],
sizeof(szDriverStringOut),
&cbDriverStringOut,
SQL_DRIVER_NOPROMPT );
    if (rc != SUCCEEDED)
        HandleErrorDBC(c_hdbc1);

// Connection 4
    sprintf( szDriverString , "DRIVER={SQL
Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s",
aptr->server,
aptr->user,
aptr->password,
aptr->database );

    rc = SQLSetConnectOption (c_hdbc2,
SQL_PACKET_SIZE, aptr->pack_size);
    if (rc != SUCCEEDED)
        HandleErrorDBC(c_hdbc2);

    rc = SQLDriverConnect ( c_hdbc2,
NULL,
(SQLCHAR*)&szDriverString[0],
SQL_NTS,
(SQLCHAR*)&szDriverStringOut[0],
sizeof(szDriverStringOut),
&cbDriverStringOut,

```

```

SQL_DRIVER_NOPROMPT);
    if (rc != SUCCEEDED)
        HandleErrorDBC(c_hdbc2);

// Connection 5
    sprintf( szDriverString , "DRIVER={SQL
Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s",
aptr->server,
aptr->user,
aptr->password,
aptr->database );

    rc = SQLSetConnectOption (o_hdbc1,
SQL_PACKET_SIZE, aptr->pack_size);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc1);

    rc = SQLDriverConnect ( o_hdbc1,
NULL,
(SQLCHAR*)&szDriverString[0],
SQL_NTS,
(SQLCHAR*)&szDriverStringOut[0],
sizeof(szDriverStringOut),
&cbDriverStringOut,
SQL_DRIVER_NOPROMPT );
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc1);

// Connection 6
    sprintf( szDriverString , "DRIVER={SQL
Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s",
aptr->server,
aptr->user,
aptr->password,
aptr->database );

    rc = SQLSetConnectOption (o_hdbc2,
SQL_PACKET_SIZE, aptr->pack_size);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc2);

    rc = SQLDriverConnect ( o_hdbc2,
NULL,
(SQLCHAR*)&szDriverString[0],
SQL_NTS,
(SQLCHAR*)&szDriverStringOut[0],

```

```

sizeof(szDriverStringOut),

&cbDriverStringOut,

SQL_DRIVER_NOPROMPT);
    if (rc != 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, "isql -S%s -U%s -P%s -e -i%s\\%s.sql >
%s\\%s.log",

aptr->server,
aptr->user,
aptr->password,
aptr->index_script_path,
index_script,

```

```

LogBase,
index_script);

system(cmd);

printf("Finished index creation: %s\n",index_script);
}

void HandleErrorDBC (SQLHDBC hdbc1)
{
    SQLCHAR                SqlState[6],
Msg[SQL_MAX_MESSAGE_LENGTH];
    SQLINTEGER NativeError;
    SQLSMALLINT i, MsgLen;
    SQLRETURN rc2;
    char                timebuf[128];
    char                datebuf[128];
    FILE                *fp1;

    i = 1;
    while (( rc2 = SQLGetDiagRec(SQL_HANDLE_DBC,
hdbc1, i, SqlState , &NativeError,

Msg, sizeof(Msg) ,

&MsgLen )) != SQL_NO_DATA )
    {

        sprintf( szLastError , "%s" , Msg );

        _strtime(timebuf);
        _strdate(datebuf);

        printf( "[%s : %s] %s\n" , datebuf, timebuf,
szLastError);

        LOGFILE_CREATE("\\tpccldr.err")
        fp1 = fopen(LogFile,"w");
        if (fp1 == NULL)
            printf("ERROR: Unable to open
errorlog file.\n");
        else
        {
            fprintf(fp1, "[%s : %s] %s\n" ,
datebuf, timebuf, szLastError);
            fclose(fp1);
        }

        i++;
    }
}

void HandleErrorSTMT (HSTMT hstmt1)
{
    SQLCHAR                SqlState[6],
Msg[SQL_MAX_MESSAGE_LENGTH];
    SQLINTEGER NativeError;
    SQLSMALLINT i, MsgLen;
    SQLRETURN rc2;
    char                timebuf[128];
    char                datebuf[128];
    FILE                *fp1;

    i = 1;
    while (( rc2 = SQLGetDiagRec(SQL_HANDLE_STMT,
hstmt1, i, SqlState , &NativeError,

```



```

        Msg, sizeof(Msg),
&MsgLen)) != SQL_NO_DATA)
    {
        sprintf( szLastError, "%s", Msg);

        _strtime(timebuf);
        _strdate(datebuf);

        printf( "[%s : %s] %s\n", datebuf, timebuf,
szLastError);

        LOGFILE_CREATE("\\tpccldr.err")
        fp1 = fopen(LogFile,"w");
        if (fp1 == NULL)
            printf("ERROR: Unable to open
errorlog file.\n");
        else
        {
            fprintf(fp1, "[%s : %s] %s\n",
datebuf, timebuf, szLastError);
            fclose(fp1);
        }
        i++;
    }
}

void FormatDate ( char* szTimeCOutput )
{
    struct tm when;
    time_t now;

    time( &now );
    when = *localtime( &now );

    mktime( &when );

    // odbc datetime format
    strftime( szTimeCOutput , 30 , "%Y-%m-%d
%H:%M:%S.000", &when );

    return;
}

//=====
//
// Function : CheckDataBase
//
//=====

void CheckDataBase()
{
    RETCODE    rc;

    char        szDriverString[300];
    char        szDriverStringOut[1024];
    char        TablesBitMap[9] =
{"000000000"};
    int         i, ExitFlag;

```

```

    SQLSMALLINT    cbDriverStringOut;
    SQLCHAR        TabName[10];
    SQLINTEGER     TabNameInd, TabCount,
    TabCountInd;

    ExitFlag = 0;

    SQLAllocHandle(SQL_HANDLE_ENV,
SQL_NULL_HANDLE,&henv );

    SQLSetEnvAttr(henv, SQL_ATTR_ODBC_VERSION,
(void*)SQL_OV_ODBC3, 0 );

    SQLAllocHandle(SQL_HANDLE_DBC,henv , &v_hdbc);

    SQLSetConnectAttr(v_hdbc, SQL_COPT_SS_BCP, (void
*)SQL_BCP_ON, SQL_IS_INTEGER );

    // Open connection to SQL Server

    sprintf( szDriverString , "DRIVER={SQL
Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s",
aptr->server,
aptr->user,
aptr->password,
aptr->database );

    rc = SQLSetConnectAttr( v_hdbc,
SQL_ATTR_PACKET_SIZE, (SQLPOINTER)aptr->pack_size,
SQL_IS_UINTEGER );
    if (rc != SQL_SUCCESS)
        HandleErrorDBC(v_hdbc);

    rc = SQLDriverConnect ( v_hdbc,
NULL,
(SQLCHAR*)&szDriverString[0],
SQL_NTS,
(SQLCHAR*)&szDriverStringOut[0],
sizeof(szDriverStringOut),
&cbDriverStringOut,
SQL_DRIVER_NOPROMPT );

    // if the rc is SQL_ERROR, the the TPCC database
probably does not exist
    if (rc == SQL_ERROR)
    {
        printf("The database TPCC does not appear to
exist!\n");
        printf("\nCheck %s\ directory for database
creation errors.\n", LogBase);

        // cleanup database connections and handles
        SQLFreeHandle(SQL_HANDLE_STMT,
v_hstmt);
        SQLDisconnect(v_hdbc);
        SQLFreeHandle(SQL_HANDLE_DBC,
v_hdbc);

```

```

// since there is not a database, exit back to
SETUP.CMD
    exit(1);
}

if ( SQLAllocHandle(SQL_HANDLE_STMT,v_hdbc ,
&v_hstmt) != SQL_SUCCESS )
    HandleErrorDBC(v_hdbc);

if ( SQLBindCol(v_hstmt, 1, SQL_C_ULONG,
&TabCount, 0, &TabCountInd) != SQL_SUCCESS )
    HandleErrorSTMT(v_hstmt);

// count the number of user tables from sysobjects
rc = SQLExecDirect(v_hstmt, "select count(*) from
sysobjects where xtype = 'U'", SQL_NTS);
if ((rc != SQL_SUCCESS) && (rc !=
SQL_SUCCESS_WITH_INFO))
    HandleErrorSTMT(v_hstmt);

if ( SQLFetch(v_hstmt) != SQL_SUCCESS )
    HandleErrorSTMT(v_hstmt);

// if the number of tables is less than 9, select all the user
tables in TPCC
if (TabCount != 9)
{
    SQLFreeHandle(SQL_HANDLE_STMT,
v_hstmt);

    SQLAllocHandle(SQL_HANDLE_STMT,
v_hdbc , &v_hstmt);

    if ( SQLBindCol(v_hstmt, 1, SQL_C_CHAR,
&TabName, sizeof(TabName), &TabNameInd) != SQL_SUCCESS )
        HandleErrorSTMT(v_hstmt);

    // select the list of user tables into a result set
rc = SQLExecDirect(v_hstmt, "select * from
sysobjects where xtype = 'U'", SQL_NTS);
if ((rc != SQL_SUCCESS) && (rc !=
SQL_SUCCESS_WITH_INFO))
    HandleErrorSTMT(v_hstmt);

    // go through the result set and set the bitmap for
each found table
    // set the bitmap to '1' if the table name is found

    while ((rc = SQLFetch(v_hstmt)) !=
SQL_NO_DATA)
    {
        switch( TabName[0] )
        {
            case 'w':
                TablesBitMap[0] = '1';
                break;
            case 'd':
                TablesBitMap[1] = '1';
                break;
            case 'c':
                TablesBitMap[2] = '1';
                break;
            case 'h':
                TablesBitMap[3] = '1';
                break;
            case 'n':
                TablesBitMap[4] = '1';

```

```

                break;
            case 'o':
                if (TabName[5] = 's')
                    break;
                if (TabName[5] = '_')
                    break;
            case 'i':
                TablesBitMap[7] = '1';
                break;
            case 's':
                TablesBitMap[8] = '1';
                break;
        }
    }

    // a '0' ExitFlag means do NOT exit the loader
early, a '1' means exit the loader early
    ExitFlag = 0;

    // iterate through the bitmap to display which
table(s) is actually missing
    for (i = 0; i <= 8; i++)
    {
        switch(i)
        {
            case 0:
                if (TablesBitMap[i] == '0')
                {
                    printf("The
Warehouse table is missing or damaged.\n");
                    ExitFlag = 1;
                }
                break;
            case 1:
                if (TablesBitMap[i] == '0')
                {
                    printf("The
District table is missing or damaged.\n");
                    ExitFlag = 1;
                }
                break;
            case 2:
                if (TablesBitMap[i] == '0')
                {
                    printf("The
Customer table is missing or damaged.\n");
                    ExitFlag = 1;
                }
                break;
            case 3:
                if (TablesBitMap[i] == '0')
                {
                    printf("The
History table is missing or damaged.\n");
                    ExitFlag = 1;
                }
                break;
            case 4:
                if (TablesBitMap[i] == '0')
                {
                    printf("The
New_Order table is missing or damaged.\n");
                    ExitFlag = 1;
                }
                break;

```

```

case 5:
    if (TablesBitMap[i] == '0')
    {
        printf("The
Orders table is missing or damaged.\n");
        ExitFlag = 1;
    }
    break;
case 6:
    if (TablesBitMap[i] == '0')
    {
        printf("The
Order_Line table is missing or damaged.\n");
        ExitFlag = 1;
    }
    break;
case 7:
    if (TablesBitMap[i] == '0')
    {
        printf("The
Item table is missing or damaged.\n");
        ExitFlag = 1;
    }
    break;
case 8:
    if (TablesBitMap[i] == '0')
    {
        printf("The
Stock table is missing or damaged.\n");
        ExitFlag = 1;
    }
    break;
    }

// if one or more tables are missing, display
message and exit the loader
if (ExitFlag = 1)
{
    printf("\nExiting TPC-C Loader!\n");
    printf("\nCheck %s\ directory for
database\n", LogBase);
    printf("or table creation errors.\n");

    // cleanup database connections and
handles
SQLFreeHandle(SQL_HANDLE_STMT, v_hstmt);
SQLDisconnect(v_hdbc);

SQLFreeHandle(SQL_HANDLE_DBC, v_hdbc);

    exit(1);
}

// cleanup database connections and handles
SQLFreeHandle(SQL_HANDLE_STMT, v_hstmt);
SQLDisconnect(v_hdbc);
SQLFreeHandle(SQL_HANDLE_DBC, v_hdbc);

return;
}

tpccldr.mak

```

```

# Microsoft Developer Studio Generated NMAKE File, Format
Version 4.10
# ** DO NOT EDIT **

# TARGETTYPE "Win32 (x86) Console Application" 0x0103

!IF "$(CFG)" == ""
CFG=tpccldr - Win32 Debug
!MESSAGE No configuration specified. Defaulting to tpccldr - Win32
Debug.
!ENDIF

!IF "$(CFG)" != "tpccldr - Win32 Release" && "$(CFG)" !=\
"tpccldr - Win32 Debug"
!MESSAGE Invalid configuration "$(CFG)" specified.
!MESSAGE You can specify a configuration when running NMAKE
on this makefile
!MESSAGE by defining the macro CFG on the command line. For
example:
!MESSAGE
!MESSAGE NMAKE /f "tpccldr.mak" CFG="tpccldr - Win32 Debug"
!MESSAGE
!MESSAGE Possible choices for configuration are:
!MESSAGE "tpccldr - Win32 Release" (based on "Win32 (x86)
Console Application")
!MESSAGE "tpccldr - Win32 Debug" (based on "Win32 (x86)
Console Application")
!MESSAGE
!ERROR An invalid configuration is specified.
!ENDIF

!IF "$(OS)" == "Windows_NT"
NULL=
!ELSE
NULL=nul
!ENDIF

#####
#####
# Begin Project
# PROP Target_Last_Scanned "tpccldr - Win32 Debug"
RSC=rc.exe
CPP=cl.exe

!IF "$(CFG)" == "tpccldr - Win32 Release"

# PROP BASE Use_MFC 0
# PROP BASE Use_Debug_Libraries 0
# PROP BASE Output_Dir "Release"
# PROP BASE Intermediate_Dir "Release"
# PROP BASE Target_Dir ""
# PROP Use_MFC 0
# PROP Use_Debug_Libraries 0
# PROP Output_Dir "bin"
# PROP Intermediate_Dir "objects"
# PROP Target_Dir ""
OUTDIR=.\bin
INTDIR=.\objects

ALL : "$(OUTDIR)\tpccldr.exe"

CLEAN :
    -@erase "$(INTDIR)\getargs.obj"
    -@erase "$(INTDIR)\random.obj"
    -@erase "$(INTDIR)\strings.obj"
    -@erase "$(INTDIR)\time.obj"
    -@erase "$(INTDIR)\tpccldr.obj"
    -@erase "$(OUTDIR)\tpccldr.exe"

```

```

$(OUTDIR) :
  if not exist "$(OUTDIR)/$(NULL)" mkdir "$(OUTDIR)"

$(INTDIR) :
  if not exist "$(INTDIR)/$(NULL)" mkdir "$(INTDIR)"

# ADD BASE CPP /nologo /W3 /GX /O2 /D "WIN32" /D "NDEBUG"
/D "_CONSOLE" /YX /c
# ADD CPP /nologo /MT /W3 /GX /O2 /I "c:\mssql\dblib\include" /D
"NDEBUG" /D "WIN32" /D "_CONSOLE" /D "DBNTWIN32" /c
# SUBTRACT CPP /YX
CPP_PROJ=/nologo /MT /W3 /GX /O2 /I "c:\mssql\dblib\include" /D
"NDEBUG" /D\
"WIN32" /D "_CONSOLE" /D "DBNTWIN32" /Fo "$(INTDIR)"/" /c
CPP_OBJS=. \objects\
CPP_SBRS=. \.
# ADD BASE RSC /I 0x409 /d "NDEBUG"
# ADD RSC /I 0x409 /d "NDEBUG"
BSC32=bscmake.exe
# ADD BASE BSC32 /nologo
# ADD BSC32 /nologo
BSC32_FLAGS=/nologo /o "$(OUTDIR)/tpccldr.bsc"
BSC32_SBRS= \

LINK32=link.exe
# ADD BASE LINK32 kernel32.lib user32.lib gdi32.lib winspool.lib
comdlg32.lib advapi32.lib shell32.lib ole32.lib oleaut32.lib uuid.lib
odbc32.lib odbccp32.lib /nologo /subsystem:console /machine:I386
# ADD LINK32 c:\mssql\dblib\lib\ntwdblib.lib kernel32.lib user32.lib
gdi32.lib winspool.lib comdlg32.lib advapi32.lib shell32.lib ole32.lib
oleaut32.lib uuid.lib odbc32.lib odbccp32.lib /nologo
/subsystem:console /pdb:none /machine:I386
LINK32_FLAGS=c:\mssql\dblib\lib\ntwdblib.lib kernel32.lib
user32.lib gdi32.lib\
winspool.lib comdlg32.lib advapi32.lib shell32.lib ole32.lib
oleaut32.lib\
uuid.lib odbc32.lib odbccp32.lib /nologo /subsystem:console
/pdb:none\
/machine:I386 /out:"$(OUTDIR)/tpccldr.exe"
LINK32_OBJS= \
    "$(INTDIR)\getargs.obj" \
    "$(INTDIR)\random.obj" \
    "$(INTDIR)\strings.obj" \
    "$(INTDIR)\time.obj" \
    "$(INTDIR)\tpccldr.obj"

$(OUTDIR)\tpccldr.exe : "$(OUTDIR)" $(DEF_FILE)
$(LINK32_OBJS)
    $(LINK32) @<<
    $(LINK32_FLAGS)$(LINK32_OBJS)
<<

!ELSEIF "$(CFG)" == "tpccldr - Win32 Debug"

# PROP BASE Use_MFC 0
# PROP BASE Use_Debug_Libraries 1
# PROP BASE Output_Dir "Debug"
# PROP BASE Intermediate_Dir "Debug"
# PROP BASE Target_Dir ""
# PROP Use_MFC 0
# PROP Use_Debug_Libraries 1
# PROP Output_Dir "bin"
# PROP Intermediate_Dir "objects"
# PROP Target_Dir ""
OUTDIR=. \bin
INTDIR=. \objects

```

```

ALL : "$(OUTDIR)\tpccldr.exe"

CLEAN :
    -@erase "$(INTDIR)\getargs.obj"
    -@erase "$(INTDIR)\random.obj"
    -@erase "$(INTDIR)\strings.obj"
    -@erase "$(INTDIR)\time.obj"
    -@erase "$(INTDIR)\tpccldr.obj"
    -@erase "$(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 "$(INTDIR)"/" /c
CPP_OBJS=. \objects\
CPP_SBRS=. \.
# ADD BASE RSC /I 0x409 /d "_DEBUG"
# ADD RSC /I 0x409 /d "_DEBUG"
BSC32=bscmake.exe
# ADD BASE BSC32 /nologo
# ADD BSC32 /nologo
BSC32_FLAGS=/nologo /o "$(OUTDIR)/tpccldr.bsc"
BSC32_SBRS= \

LINK32=link.exe
# ADD BASE LINK32 kernel32.lib user32.lib gdi32.lib winspool.lib
comdlg32.lib advapi32.lib shell32.lib ole32.lib oleaut32.lib uuid.lib
odbc32.lib odbccp32.lib /nologo /subsystem:console /debug
/machine:I386
# ADD LINK32 c:\mssql\dblib\lib\ntwdblib.lib kernel32.lib user32.lib
gdi32.lib winspool.lib comdlg32.lib advapi32.lib shell32.lib ole32.lib
oleaut32.lib uuid.lib odbc32.lib odbccp32.lib /nologo
/subsystem:console /pdb:none /debug /machine:I386
LINK32_FLAGS=c:\mssql\dblib\lib\ntwdblib.lib kernel32.lib
user32.lib gdi32.lib\
winspool.lib comdlg32.lib advapi32.lib shell32.lib ole32.lib
oleaut32.lib\
uuid.lib odbc32.lib odbccp32.lib /nologo /subsystem:console
/pdb:none /debug\
/machine:I386 /out:"$(OUTDIR)/tpccldr.exe"
LINK32_OBJS= \
    "$(INTDIR)\getargs.obj" \
    "$(INTDIR)\random.obj" \
    "$(INTDIR)\strings.obj" \
    "$(INTDIR)\time.obj" \
    "$(INTDIR)\tpccldr.obj"

$(OUTDIR)\tpccldr.exe : "$(OUTDIR)" $(DEF_FILE)
$(LINK32_OBJS)
    $(LINK32) @<<
    $(LINK32_FLAGS)$(LINK32_OBJS)
<<

```

```

!ENDIF

.c{$(CPP_OBJS)}.obj:
  $(CPP) $(CPP_PROJ) $<

.cpp{$(CPP_OBJS)}.obj:
  $(CPP) $(CPP_PROJ) $<

.cxx{$(CPP_OBJS)}.obj:
  $(CPP) $(CPP_PROJ) $<

.c{$(CPP_SBRS)}.sbr:
  $(CPP) $(CPP_PROJ) $<

.cpp{$(CPP_SBRS)}.sbr:
  $(CPP) $(CPP_PROJ) $<

.cxx{$(CPP_SBRS)}.sbr:
  $(CPP) $(CPP_PROJ) $<

#####
#####
# Begin Target

# Name "tpccldr - Win32 Release"
# Name "tpccldr - Win32 Debug"

!IF "$(CFG)" == "tpccldr - Win32 Release"

!ELSEIF "$(CFG)" == "tpccldr - Win32 Debug"

!ENDIF

#####
#####
# Begin Source File

SOURCE=. \src\random.c
DEP_CPP_RAND0=\
  ".\src\tpcc.h" \
  "\mssql\dblib\include\sqldb.h" \
  "\mssql\dblib\include\sqlfront.h" \

"$$(INTDIR)\random.obj" : $(SOURCE) $(DEP_CPP_RAND0)
"$$(INTDIR)"
  $(CPP) $(CPP_PROJ) $(SOURCE)

# End Source File
#####
#####
# Begin Source File

SOURCE=. \src\strings.c
DEP_CPP_STRIN=\
  ".\src\tpcc.h" \
  "\mssql\dblib\include\sqldb.h" \
  "\mssql\dblib\include\sqlfront.h" \

"$$(INTDIR)\strings.obj" : $(SOURCE) $(DEP_CPP_STRIN)
"$$(INTDIR)"
  $(CPP) $(CPP_PROJ) $(SOURCE)

# End Source File

#####
#####
# Begin Source File

SOURCE=. \src\time.c
DEP_CPP_TIME_=\
  ".\src\tpcc.h" \
  "\mssql\dblib\include\sqldb.h" \
  "\mssql\dblib\include\sqlfront.h" \

"$$(INTDIR)\time.obj" : $(SOURCE) $(DEP_CPP_TIME_)
"$$(INTDIR)"
  $(CPP) $(CPP_PROJ) $(SOURCE)

# End Source File
#####
#####
# Begin Source File

SOURCE=. \src\tpccldr.c
DEP_CPP_TPCCL=\
  ".\src\tpcc.h" \
  "\mssql\dblib\include\sqldb.h" \
  "\mssql\dblib\include\sqlfront.h" \

"$$(INTDIR)\tpccldr.obj" : $(SOURCE) $(DEP_CPP_TPCCL)
"$$(INTDIR)"
  $(CPP) $(CPP_PROJ) $(SOURCE)

# End Source File
#####
#####
# Begin Source File

SOURCE=. \src\getargs.c
DEP_CPP_GETAR=\
  ".\src\tpcc.h" \
  "\mssql\dblib\include\sqldb.h" \
  "\mssql\dblib\include\sqlfront.h" \

"$$(INTDIR)\getargs.obj" : $(SOURCE) $(DEP_CPP_GETAR)
"$$(INTDIR)"
  $(CPP) $(CPP_PROJ) $(SOURCE)

# End Source File
# End Target
# End Project
#####
#####

```

Appendix C: Tunable Parameters

Database Startup Parameters

Microsoft SQL Server 2000 Startup Parameters

```
c:\Program Files\Microsoft SQL Server\MSSQL\Binn\sqlservr.exe -c -x
-t3502 -g1024
```

Where:

```
-c Start SQL Server independent of the Service Control Manager
-x Disable the keeping of CPU time and cache hit ratio statistics
-t3502 Writes a message to the SQL Server Errorlog showing the
beginning and ending time of each checkpoint
-g1024 Specifies the amount of memory that is set aside for allocations
not from the buffer pool
```

Database Information

```
1> 2> 3> 4> 5> 6> 7> 8> 9> 10> 11>
-- File:  VERSION.SQL
--      Microsoft TPC-C Benchmark Kit Ver. 4.30
--      Copyright Microsoft, 2000
-- Purpose: Returns SQL Server version string
```

```
print ''
select convert(char(30), getdate(),21)
print ''
```

```
-----
2001-03-05 18:21:24.117
```

(1 row affected)

```
1> 2> 3> 4> 5> Server Name:
```

```
print 'Server Name:'
select @@servername
print ''
```

```
-----
RTNODE01
```

(1 row affected)

```
1> 2> 3> 4> 5> Server Version:
```

```
print 'Server Version:'
select @@version
print ''
```

```
-----
```

```
-----
-----
Microsoft SQL Server 2000 - 8.00.208 (Intel X86)
Sep 27 2000 14:06:38
Cop
yright (c) 1988-2000 Microsoft Corporation
Enterprise Edition on Windo
ws NT 5.0 (Build 2195: Service Pack 1)
```

(1 row affected)

```
1> 2>
1> 2> 3> 4> 5> 6> 7> 8> 9> 10>
-- File:  CONFIG.SQL
--      Microsoft TPC-C Benchmark Kit Ver. 4.30
--      Copyright Microsoft, 2000
-- Purpose: Collects SQL Server configuration parameters
```

```
print ''
select convert(char(30), getdate(),21)
print ''
```

```
-----
2001-03-05 18:21:24.583
```

(1 row affected)

```
1> 2> 3> DBCC execution completed. If DBCC printed error
messages, contact your system administrator.
Configuration option 'show advanced options' changed from 1 to 1.
Run the RECONFIGURE statement to install.
```

```
sp_configure 'show advanced',1
1> 2> reconfigure with override
```

```
1> 2> sp_configure
```

name	minimum	maximum	config_value	run_value
affinity mask	0	2147483647	255	255
allow updates	0	1	0	0
awe enabled	0	1	1	1
c2 audit mode	0	1	0	0
cost threshold for parallelism	0	32767	5	5
cursor threshold	-1	2147483647	-1	-1
default full-text language	0	2147483647	1033	1033
default language	0	9999	0	0
fill factor (%)	0	100	0	0
index create memory (KB)	0	704 2147483647	0	0
lightweight pooling	0	1	0	0
locks	5000	2147483647	0	0
max degree of parallelism	0	32	0	0
max server memory (MB)	4	2147483647	2147483647	2147483647
max text repl size (B)	0	2147483647	65536	65536
max worker threads	32	32767	750	750
media retention	0	365	0	0
min memory per query (KB)	1024	2147483647	1024	1024
min server memory (MB)	0	2147483647	0	0

```

nested triggers          0      1      1      1
network packet size (B) 512    65536   4096
4096
open objects            0 2147483647      0      0
priority boost          0      1      1      1
query governor cost limit      0 2147483647      0      0

query wait (s)          -1 2147483647     -1     -1
recovery interval (min)  0    32767     36     36
remote access           0      1      1      1
remote login timeout (s)  0 2147483647      0      0

remote proc trans       0      1      0      0
remote query timeout (s)  0 2147483647      0      0

scan for startup proc   0      1      0      0
set working set size    0      1      0      0
show advanced options   0      1      1      1
two digit year cutoff   1753   9999     2049   2049

user connections        0    32767      0      0
user options            0    32767      0      0

```

1>

Software Update

Software Update

SQL Server QFE 208 was applied to all 16 server nodes and is available.

Windows 2000 SP1 was applied to all 16 server nodes and is available.

Microsoft Windows 2000 Datacenter Configuration

boot.ini

```

[boot loader]
timeout=5
default=multi(0)disk(0)rdisk(0)partition(1)\WINNT
[operating systems]
multi(0)disk(0)rdisk(0)partition(1)\WINNT="Microsoft Windows
2000 Datacenter Server /3GB /PAE" /fastdetect /3GB /PAE
multi(0)disk(0)rdisk(0)partition(1)\WINNT="Microsoft Windows
2000 Datacenter Server" /fastdetect

```

Enabling AWE

The following procedure was used on each node to enable SQL Server access to AWE memory:

- 1) Ran gpedit.msc to open the Group Policy Editor window.
- 2) Selected Computer Configuration -> Windows Settings -> Security Settings -> Local Policies -> User Rights Assignment.
- 3) Double-clicked on Lock Pages in Memory in the right hand pane of the editor.
- 4) Clicked Add.
- 5) Selected the Administrators group.
- 6) Click OK to set the local security policy setting.
- 7) Click OK to return to the Group Policy Editor.

8) Close the editor. Reboot the node.

Enabling VIA

The following procedure was used on each node to enable VIA support for SQL Server:

- 1) Open the SQL Client Network Utility.
- 2) Enable VIA as a protocol.
- 3) Select the Alias page. Press the Add button.
- 4) Check the VIA radio button.
- 5) Create a server alias.
- 6) Close the SQL Client Network Utility.
- 7) Open the SQL Server Network Utility.
- 8) Add VIA to the list of enabled protocols.
- 9) Close the SQL Server Network Utility.

Specifying Remote DTC

One remote DTC coordinator was specified.

- Nodes 1-16 specified DTCSERV1 as the remote DTC coordinator.

System Information Report for Software Configuration

This software configuration inventory was obtained from 1 of 16 nodes.

All 16 nodes are identically configured.

System Information report written at: 03/05/2001 06:23:34 PM
[System Summary]

Item	Value
OS Name	Microsoft Windows 2000 Datacenter Server
Version	5.0.2195 Service Pack 1 Build 2195
OS Manufacturer	Microsoft Corporation
System Name	RTNODE01
System Manufacturer	IBM
System Model	IBM Server -[8681]-
System Type	X86-based PC
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
BIOS Version	IBM BIOS Ver 6.0
Windows Directory	C:\WINNT
System Directory	C:\WINNT\System32
Boot Device	\Device\Harddisk0\Partition 1
Locale	United States
User Name	RTNODE01\tpcc
Time Zone	Eastern Standard Time
Total Physical Memory	16,514,360 KB
Available Physical Memory	16,280,660 KB
Total Virtual Memory	34,983,416 KB
Available Virtual Memory	34,635,660 KB
Page File Space	18,469,056 KB
Page File	C:\pagefile.sys

```

1> 2> 3> 4> 5> 6> 7> 8> 9> 10> 11>
-- File:  VERSION.SQL
--      Microsoft TPC-C Benchmark Kit Ver. 4.30
--      Copyright Microsoft, 2000
-- Purpose: Returns SQL Server version string

```

```

print ''
select convert(char(30), getdate(),21)
print ''

```

```

-----
2001-03-05 18:21:24.117

```

(1 row affected)

```

1> 2> 3> 4> 5> Server Name:

```

```

print 'Server Name:'
select @@servername
print ''

```

```

-----
RTNODE01

```

(1 row affected)

```

1> 2> 3> 4> 5> Server Version:

```

```

print 'Server Version:'
select @@version
print ''

```

```

-----
-----
-----
-----
-----
Microsoft SQL Server 2000 - 8.00.208 (Intel X86)
Sep 27 2000 14:06:38
Cop
yright (c) 1988-2000 Microsoft Corporation
Enterprise Edition on Windo
ws NT 5.0 (Build 2195: Service Pack 1)

```

System Information Report for Hardware Configuration

This hardware configuration inventory was obtained from 1 of 16 nodes.
All 16 nodes are identically configured.

System Information report written at: 03/05/2001 06:05:01 PM
[System Information]

[Following are sub-categories of this main category]

[System Summary]

Item	Value
OS Name	Microsoft Windows 2000 Datacenter Server
Version	5.0.2195 Service Pack 1 Build 2195
OS Manufacturer	Microsoft Corporation
System Name	RTNODE01
System Manufacturer	IBM
System Model	IBM Server -[8681]-
System Type	X86-based PC
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
BIOS Version	IBM BIOS Ver 6.0
Windows Directory	C:\WINNT
System Directory	C:\WINNT\System32
Boot Device	\Device\Harddisk0\Partition1
Locale	United States
User Name	RTNODE01\tpcc
Time Zone	Eastern Standard Time
Total Physical Memory	16,514,360 KB
Available Physical Memory	515,800 KB
Total Virtual Memory	34,983,416 KB
Available Virtual Memory	3,102,640 KB
Page File Space	18,469,056 KB
Page File	C:\pagefile.sys

[Hardware Resources]

[Following are sub-categories of this main category]

[Conflicts/Sharing]

Resource	Device
IRQ 51	Adaptec AIC-7896/AIC-7897 PCI Ultra2 SCSI Controller
IRQ 51	Adaptec AIC-7896/AIC-7897 PCI Ultra2 SCSI Controller

[DMA]

Channel	Device	Status
2	Standard floppy disk controller	OK
4	Direct memory access controller	OK

[Forced Hardware]

Device	PNP Device ID
No Forced Hardware	

[I/O]

Address Range	Device	Status
0x0000-0x0CF7	PCI bus	OK
0x0000-0x0CF7	Direct memory access controller	OK
0x0D00-0x5FFF	PCI bus	OK
0x2000-0x20FF	Other PCI Bridge Device	OK
0x2180-0x218F	Other PCI Bridge Device	OK
0x03B0-0x03BB	S3 Inc. Trio3D	OK
0x03C0-0x03DF	S3 Inc. Trio3D	OK
0x2200-0x22FF	IBM Netfinity ServeRAID 4H Controller	OK
0x2300-0x23FF	IBM Netfinity ServeRAID 4H Controller	OK
0x2400-0x24FF	IBM Netfinity ServeRAID 4H Controller	OK
0x2500-0x25FF	Adaptec AIC-7896/AIC-7897 PCI Ultra2 SCSI Controller	OK

0x2600-0x26FF	Adaptec AIC-7896/AIC-7897 PCI Ultra2 SCSI Controller	OK
0x0A79-0x0A79	ISAPNP Read Data Port	OK
0x0279-0x0279	ISAPNP Read Data Port	OK
0x02F4-0x02F7	ISAPNP Read Data Port	OK
0x002E-0x002F	Motherboard resources	OK
0x0438-0x0439	Motherboard resources	OK
0x0430-0x0437	Motherboard resources	OK
0x0060-0x0060	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard	OK
0x0064-0x0064	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard	OK
0x03F0-0x03F5	Standard floppy disk controller	OK
0x03F7-0x03F7	Standard floppy disk controller	OK
0x0378-0x037F	Printer Port (LPT1)	OK
0x03F8-0x03FF	Communications Port (COM1)	OK
0x02F8-0x02FF	Communications Port (COM2)	OK
0x0020-0x0021	Advanced programmable interrupt controller	OK
0x00A0-0x00A1	Advanced programmable interrupt controller	OK
0x0080-0x008F	Direct memory access controller	OK
0x00C0-0x00DF	Direct memory access controller	OK
0x0040-0x0043	System timer	OK
0x0070-0x0073	System CMOS/real time clock	OK
0x0061-0x0061	System speaker	OK
0x00F0-0x00FF	Numeric data processor	OK
0x04D0-0x04D1	Motherboard resources	OK
0x00E8-0x00EF	Motherboard resources	OK
0x5FF0-0x5FFF	Intel(r) 82371AB/EB PCI Bus Master IDE Controller	OK
0x01F0-0x01F7	Primary IDE Channel	OK
0x03F6-0x03F6	Primary IDE Channel	OK
0x5FC0-0x5FDF	Intel 82371AB/EB PCI to USB Universal Host Controller	OK
0x6000-0xAFFF	PCI bus	OK
0x6000-0xAFFF	IBM Netfinity ServeRAID 4H Controller	OK
0x6100-0x61FF	IBM Netfinity ServeRAID 4H Controller	OK
0x6200-0x62FF	IBM Netfinity ServeRAID 4H Controller	OK
0x6300-0x63FF	IBM Netfinity ServeRAID 4H Controller	OK
0xB000-0xCFFF	PCI bus	OK
0xB000-0xCFFF	IBM 10/100 EtherJet PCI Management Adapter	OK
0xB040-0xB07F	IBM 10/100 EtherJet PCI Management Adapter #2	OK
0xD000-0xFFFF	PCI bus	OK
0xD000-0xFFFF	IBM Netfinity ServeRAID 4H Controller	OK

[IRQs]

IRQ Number	Device
9	Microsoft ACPI-Compliant System
5	Other PCI Bridge Device
25	IBM Netfinity ServeRAID 4H Controller
26	IBM Netfinity ServeRAID 4H Controller
27	IBM Netfinity ServeRAID 4H Controller
51	Adaptec AIC-7896/AIC-7897 PCI Ultra2 SCSI Controller
51	Adaptec AIC-7896/AIC-7897 PCI Ultra2 SCSI Controller
1	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard
12	PS/2 Compatible Mouse
6	Standard floppy disk controller
4	Communications Port (COM1)
3	Communications Port (COM2)
8	System CMOS/real time clock
13	Numeric data processor
14	Primary IDE Channel
49	Intel 82371AB/EB PCI to USB Universal Host Controller
16	IBM Netfinity ServeRAID 4H Controller

17	IBM Netfinity ServeRAID 4H Controller
18	IBM Netfinity ServeRAID 4H Controller
19	IBM Netfinity ServeRAID 4H Controller
23	IBM 10/100 EtherJet PCI Management Adapter
24	IBM 10/100 EtherJet PCI Management Adapter #2
21	IBM Netfinity ServeRAID 4H Controller
22	cLAN Host Adapter

[Memory]

Range	Device	Status
0xA0000-0xBFFFF	PCI bus	OK
0xA0000-0xBFFFF	S3 Inc. Trio3D	OK
0xC8000-0xDFFFF	PCI bus	OK
0xF0000000-0xFC1FFFFF	PCI bus	OK
0xFEC00000-0xFFFFFFFF	PCI bus	OK
0xFEC00000-0xFFFFFFFF	Advanced programmable interrupt controller	OK
0xFC1FFC00-0xFC1FFCFF	Other PCI Bridge Device	OK
0xFC1E0000-0xFC1EFFFF	Other PCI Bridge Device	OK
0xF8000000-0xFBFFFFFF	S3 Inc. Trio3D	OK
0xFC000000-0xFC0FFFFF	IBM Netfinity ServeRAID 4H Controller	OK
0xF7F00000-0xF7FFFFFF	IBM Netfinity ServeRAID 4H Controller	OK
0xF7E00000-0xF7EFFFFF	IBM Netfinity ServeRAID 4H Controller	OK
0xFC1FE000-0xFC1FEFFF	Adaptec AIC-7896/AIC-7897 PCI Ultra2 SCSI Controller	OK
0xFC1FD000-0xFC1FDFFF	Adaptec AIC-7896/AIC-7897 PCI Ultra2 SCSI Controller	OK
0xFC200000-0xFCBFFFFFF	PCI bus	OK
0xFCB00000-0xFCBFFFFFF	IBM Netfinity ServeRAID 4H Controller	OK
0xFCA00000-0xFCAFFFFFF	IBM Netfinity ServeRAID 4H Controller	OK
0xFC900000-0xFC9FFFFF	IBM Netfinity ServeRAID 4H Controller	OK
0xFC800000-0xFC8FFFFFF	IBM Netfinity ServeRAID 4H Controller	OK
0xFCC00000-0xFCFFFFFF	PCI bus	OK
0xFCFFF000-0xFCFFFFFF	IBM 10/100 EtherJet PCI Management Adapter	OK
0xFCE00000-0xFCEFFFFFF	IBM 10/100 EtherJet PCI Management Adapter	OK
0xFCFFE000-0xFCFFEFFF	IBM 10/100 EtherJet PCI Management Adapter #2	OK
0xFCD00000-0xFCDFFFFFF	IBM 10/100 EtherJet PCI Management Adapter #2	OK
0xFD000000-0xFEBFFFFFF	PCI bus	OK
0xFD000000-0xFEBFFFFFF	cLAN Host Adapter	OK
0xFEB00000-0xFEBFFFFFF	IBM Netfinity ServeRAID 4H Controller	OK
0xFEAE0000-0xFEFFFFFF	cLAN Host Adapter	OK
0xFE800000-0xFE9FFFFF	cLAN Host Adapter	OK
0xFEAD0000-0xFEADFFFF	cLAN Host Adapter	OK

[Components]

[Following are sub-categories of this main category]

[Multimedia]

[Following are sub-categories of this main category]

[Audio Codecs]

Codec	Manufacturer	Description	Status	File
Version	Size	Creation Date		
c:\winnt\system32\lhacm.acm	Microsoft Corporation			
OK	C:\WINNT\System32\LHACM.ACM	4.4.3385		
33.27 KB (34,064 bytes)	12/19/2000 11:11:08 AM			
c:\winnt\system32\msg723.acm	Microsoft Corporation			
OK	C:\WINNT\System32\MSG723.ACM	4.4.3385		
106.77 KB (109,328 bytes)	12/19/2000 11:11:07 AM			
c:\winnt\system32\msg711.acm	Microsoft Corporation			
OK	C:\WINNT\System32\MSG711.ACM	5.00.2134.1		
10.27 KB (10,512 bytes)	7/27/2000 8:00:00 PM			
c:\winnt\system32\msadp32.acm	Microsoft Corporation			
OK	C:\WINNT\System32\MSADP32.ACM	5.00.2134.1		
14.77 KB (15,120 bytes)	7/27/2000 8:00:00 PM			
c:\winnt\system32\imaadp32.acm	Microsoft Corporation			
OK	C:\WINNT\System32\IMAADP32.ACM	5.00.2134.1		
16.27 KB (16,656 bytes)	7/27/2000			
8:00:00 PM				
c:\winnt\system32\iac25_32.ax	Intel Corporation	Indeo® audio		
software	OK	C:\WINNT\System32\IAC25_32.AX		
2.05.53	195.00 KB (199,680 bytes)	7/27/2000 8:00:00 PM		
c:\winnt\system32\msgsm32.acm	Microsoft Corporation			
OK	C:\WINNT\System32\MSGSM32.ACM	5.00.2134.1		
22.27 KB (22,800 bytes)	7/27/2000 8:00:00 PM			
c:\winnt\system32\tssoft32.acm	DSP GROUP, INC.	OK		
C:\WINNT\System32\TSSOFT32.ACM	1.01	9.27 KB (9,488 bytes)		
7/27/2000 8:00:00 PM				

[Video Codecs]

Codec	Manufacturer	Description	Status	File
Version	Size	Creation Date		
c:\winnt\system32\ir50_32.dll	Intel Corporation	Indeo® video		
5.10	OK	C:\WINNT\System32\IR50_32.DLL		
R.5.10.15.2.55	737.50 KB (755,200 bytes)	7/27/2000		
8:00:00 PM				
c:\winnt\system32\msh261.drv	Microsoft Corporation			
OK	C:\WINNT\System32\MSH261.DRV	4.4.3385		
163.77 KB (167,696 bytes)	12/19/2000 11:11:08 AM			
c:\winnt\system32\msh263.drv	Microsoft Corporation			
OK	C:\WINNT\System32\MSH263.DRV	4.4.3385		
252.27 KB (258,320 bytes)	12/19/2000 11:10:39 AM			
c:\winnt\system32\msrle32.dll	Microsoft Corporation			
OK	C:\WINNT\System32\MSRLE32.DLL	5.00.2134.1		
10.77 KB (11,024 bytes)	7/27/2000 8:00:00 PM			
c:\winnt\system32\iccvd.dll	Radius Inc.	OK		
C:\WINNT\System32\ICCVID.DLL	1.10.0.6	108.00 KB (110,592 bytes)		
7/27/2000 8:00:00 PM				
c:\winnt\system32\ir32_32.dll	Intel(R) Corporation	OK		
C:\WINNT\System32\IR32_32.DLL	Not Available			
194.50 KB (199,168 bytes)	7/27/2000 8:00:00 PM			
c:\winnt\system32\msvidc32.dll	Microsoft Corporation			
OK	C:\WINNT\System32\MSVIDC32.DLL	5.00.2134.1		
27.27 KB (27,920 bytes)	7/27/2000 8:00:00 PM			

[CD-ROM]

Item	Value
Drive	D:
Description	CD-ROM Drive
Media Loaded	False
Media Type	CD-ROM
Name	LG CD-ROM CRD-8400B
Manufacturer	(Standard CD-ROM drives)
Status	OK
Transfer Rate	Not Available
SCSI Target ID	0

PNP Device ID
 IDE\CDROMLG_CD-ROM_CRD-8400B_____1.
 10____\4249204D39393730393120202020202020202020

[Sound Device]

Item	Value
No sound devices	

[Display]

Item	Value
Name	S3 Inc. Trio3D
PNP Device ID	PCI\VEN_5333&DEV_8904&SUBSYS_00DB1014&REV_01\3&267A616A&0&20
Adapter Type	S3 Trio3D, S3 compatible
Adapter Description	S3 Inc. Trio3D
Adapter RAM	4.00 MB (4,194,304 bytes)
Installed Drivers	s3mt3d.sys
Driver Version	5.01.526.0007
INF File	s3trio3d.inf (S3Inc section)
Color Planes	1
Color Table Entries	65536
Resolution	800 x 600 x 60 hertz
Bits/Pixel	16

[Infrared]

Item	Value
No infrared devices	

[Input]

[Following are sub-categories of this main category]

[Keyboard]

Item	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&43B47AD&0
NumberOfFunctionKeys	12

[Pointing Device]

Item	Value
Hardware Type	PS/2 Compatible Mouse
Number of Buttons	3
Status	OK
PNP Device ID	ACPI\PNP0F13\4&43B47AD&0
Power Management Supported	False
Double Click Threshold	6
Handedness	Right Handed Operation

[Modem]

Item	Value
No modems	

[Network]

[Following are sub-categories of this main category]

[Adapter]

Item Value
Name [00000000] IBM 10/100 EtherJet PCI Management Adapter
Adapter Type Ethernet 802.3
Product Name IBM 10/100 EtherJet PCI Management Adapter
Installed True
PNP Device ID
PCI\VEN_8086&DEV_1229&SUBSYS_305C1014&REV_08\3&23C
0707C&0&40
Last Reset 3/5/2001 12:32:42 PM
Index 0
Service Name E100B
IP Address 9.67.188.2
IP Subnet 255.255.255.192
Default IP Gateway 9.67.188.1
DHCP Enabled False
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address 00:04:AC:93:0B:DC
Service Name E100B
IRQ Number 23
I/O Port 0xB000-0xCFFF
Driver c:\winnt\system32\drivers\e100bnt5.sys (119056,
5.40.11.0000)

Name [00000001] IBM 10/100 EtherJet PCI Management Adapter
Adapter Type Ethernet 802.3
Product Name IBM 10/100 EtherJet PCI Management Adapter
Installed True
PNP Device ID
PCI\VEN_8086&DEV_1229&SUBSYS_305C1014&REV_08\3&23C
0707C&0&48
Last Reset 3/5/2001 12:32:42 PM
Index 1
Service Name E100B
IP Address 192.6.1.1
IP Subnet 255.255.255.0
Default IP Gateway Not Available
DHCP Enabled False
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address 00:04:AC:93:45:5B
Service Name E100B
IRQ Number 24
I/O Port 0xB040-0xB07F
Driver c:\winnt\system32\drivers\e100bnt5.sys (119056,
5.40.11.0000)

Name [00000002] RAS Async Adapter
Adapter Type Not Available
Product Name RAS Async Adapter
Installed True
PNP Device ID Not Available
Last Reset 3/5/2001 12:32:42 PM
Index 2
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 [00000003] WAN Miniport (L2TP)
Adapter Type Not Available
Product Name WAN Miniport (L2TP)
Installed True
PNP Device ID ROOT\MS_L2TPMINIPORT\0000
Last Reset 3/5/2001 12:32:42 PM
Index 3
Service Name Rasl2tp
IP Address Not Available
IP Subnet Not Available
Default IP Gateway Not Available
DHCP Enabled False
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address Not Available
Service Name Rasl2tp
Driver c:\winnt\system32\drivers\rasl2tp.sys (50320, 5.00.2179.1)
Name [00000004] WAN Miniport (PPTP)
Adapter Type Wide Area Network (WAN)
Product Name WAN Miniport (PPTP)
Installed True
PNP Device ID ROOT\MS_PPTPMINIPORT\0000
Last Reset 3/5/2001 12:32:42 PM
Index 4
Service Name PptpMiniport
IP Address Not Available
IP Subnet Not Available
Default IP Gateway Not Available
DHCP Enabled False
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address 50:50:54:50:30:30
Service Name PptpMiniport
Driver c:\winnt\system32\drivers\raspptp.sys (47376, 5.00.2160.1)
Name [00000005] Direct Parallel
Adapter Type Not Available
Product Name Direct Parallel
Installed True
PNP Device ID ROOT\MS_PTMINIPORT\0000
Last Reset 3/5/2001 12:32:42 PM
Index 5
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 [00000006] WAN Miniport (IP)
Adapter Type Not Available
Product Name WAN Miniport (IP)
Installed True
PNP Device ID ROOT\MS_NDISWANIP\0000
Last Reset 3/5/2001 12:32:42 PM
Index 6
Service Name NdisWan

IP Address Not Available
 IP Subnet Not Available
 Default IP Gateway Not Available
 DHCP Enabled False
 DHCP Server Not Available
 DHCP Lease Expires Not Available
 DHCP Lease Obtained Not Available
 MAC Address Not Available
 Service Name NdisWan
 Driver c:\winnt\system32\drivers\ndiswan.sys (89808, 5.00.2184.1)

Name [00000007] cLAN Host Adapter
 Adapter Type Ethernet 802.3
 Product Name cLAN Host Adapter
 Installed True
 PNP Device ID
 PCI\VEN_135B&DEV_0001&SUBSYS_00000000&REV_00\3&146 CA173&0&38
 Last Reset 3/5/2001 12:32:42 PM
 Index 7
 Service Name GNINDIS
 IP Address 192.168.100.1
 IP Subnet 255.255.255.0
 Default IP Gateway Not Available
 DHCP Enabled False
 DHCP Server Not Available
 DHCP Lease Expires Not Available
 DHCP Lease Obtained Not Available
 MAC Address 00:90:FA:00:0D:28
 Service Name GNINDIS
 IRQ Number 22
 Driver c:\winnt\system32\drivers\gnindis.sys (22598, 4.1.1)

[Protocol]

Item Value
 Name MSAFD Tcpi [TCP/IP]
 ConnectionlessService False
 GuaranteesDelivery True
 GuaranteesSequencing True
 MaximumAddressSize 16 bytes
 MaximumMessageSize 0 bytes
 MessageOriented False
 MinimumAddressSize 16 bytes
 PseudoStreamOriented False
 SupportsBroadcasting False
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData True
 SupportsGracefulClosing True
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD Tcpi [UDP/IP]
 ConnectionlessService True
 GuaranteesDelivery False
 GuaranteesSequencing False
 MaximumAddressSize 16 bytes
 MaximumMessageSize 65467 bytes
 MessageOriented True
 MinimumAddressSize 16 bytes
 PseudoStreamOriented False
 SupportsBroadcasting True
 SupportsConnectData False
 SupportsDisconnectData False

SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting True

Name RSVP UDP Service Provider
 ConnectionlessService True
 GuaranteesDelivery False
 GuaranteesSequencing False
 MaximumAddressSize 16 bytes
 MaximumMessageSize 65467 bytes
 MessageOriented True
 MinimumAddressSize 16 bytes
 PseudoStreamOriented False
 SupportsBroadcasting True
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption True
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting True

Name RSVP TCP Service Provider
 ConnectionlessService False
 GuaranteesDelivery True
 GuaranteesSequencing True
 MaximumAddressSize 16 bytes
 MaximumMessageSize 0 bytes
 MessageOriented False
 MinimumAddressSize 16 bytes
 PseudoStreamOriented False
 SupportsBroadcasting False
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption True
 SupportsExpeditedData True
 SupportsGracefulClosing True
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS
 [\Device\NetBT_Tcpi_{5DF00DF2-07BB-418E-A0CA-F65C6DE9B 4AE}] SECPACKET 4
 ConnectionlessService False
 GuaranteesDelivery True
 GuaranteesSequencing True
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting False
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS
 [\Device\NetBT_Tcpi_{5DF00DF2-07BB-418E-A0CA-F65C6DE9B 4AE}] DATAGRAM 4
 ConnectionlessService True

GuaranteesDelivery False
 GuaranteesSequencing False
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting True
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT_Tcpip_{6A7B67FA-E902-4271-B61F-8E0D38AB83A8}] SEQPACKET 1
 ConnectionlessService False
 GuaranteesDelivery True
 GuaranteesSequencing True
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting False
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT_Tcpip_{6A7B67FA-E902-4271-B61F-8E0D38AB83A8}] DATAGRAM 1
 ConnectionlessService True
 GuaranteesDelivery False
 GuaranteesSequencing False
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting True
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT_Tcpip_{547923BB-E2B6-4573-9E9D-86C5342DEFCC}] SEQPACKET 0
 ConnectionlessService False
 GuaranteesDelivery True
 GuaranteesSequencing True
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True

MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting False
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT_Tcpip_{547923BB-E2B6-4573-9E9D-86C5342DEFCC}] DATAGRAM 0
 ConnectionlessService True
 GuaranteesDelivery False
 GuaranteesSequencing False
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting True
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT_Tcpip_{830180B7-E282-442B-AF14-C3327727986C}] SEQPACKET 2
 ConnectionlessService False
 GuaranteesDelivery True
 GuaranteesSequencing True
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting False
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT_Tcpip_{830180B7-E282-442B-AF14-C3327727986C}] DATAGRAM 2
 ConnectionlessService True
 GuaranteesDelivery False
 GuaranteesSequencing False
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting True
 SupportsConnectData False
 SupportsDisconnectData False

SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS

[\\Device\\NetBT_Tcpip_{FF9BB37E-81B3-40FA-BFB5-FE68770FDC8E}] SEQPACKE T 3

ConnectionlessService False
 GuaranteesDelivery True
 GuaranteesSequencing True
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting False
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS

[\\Device\\NetBT_Tcpip_{FF9BB37E-81B3-40FA-BFB5-FE68770FDC8E}] DATAGRAM 3

ConnectionlessService True
 GuaranteesDelivery False
 GuaranteesSequencing False
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting True
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

[WinSock]

Item	Value
File	c:\\winnt\\system32\\winsock.dll
Version	3.10
Size	2.80 KB (2,864 bytes)
File	c:\\winnt\\system32\\wsock32.dll
Version	5.00.2195.1207
Size	21.27 KB (21,776 bytes)

[Ports]

[Following are sub-categories of this main category]

[Serial]

Item	Value
------	-------

Name	COM1
Status	OK
PNP Device ID	ACPI\\PNP0501\\1
Maximum Input Buffer Size	0
Maximum Output Buffer Size	False
Settable Baud Rate	True
Settable Data Bits	True
Settable Flow Control	True
Settable Parity	True
Settable Parity Check	True
Settable Stop Bits	True
Settable RLSD	True
Supports RLSD	True
Supports 16 Bit Mode	False
Supports Special Characters	False
Baud Rate	9600
Bits/Byte	8
Stop Bits	1
Parity	None
Busy	0
Abort Read/Write on Error	0
Binary Mode Enabled-1	
Continue XMit on XOff	0
CTS Outflow Control	0
Discard NULL Bytes	0
DSR Outflow Control	0
DSR Sensitivity	0
DTR Flow Control Type	Enable
EOF Character	0
Error Replace Character	0
Error Replacement Enabled	0
Event Character	0
Parity Check Enabled	0
RTS Flow Control Type	Enable
XOff Character	19
XOffXMit Threshold	512
XOn Character	17
XOnXMit Threshold	2048
XOnXOff InFlow Control	0
XOnXOff OutFlow Control	0
IRQ Number	4
I/O Port	0x03F8-0x03FF
Driver	c:\\winnt\\system32\\drivers\\serial.sys (62448, 5.00.2134.1)

Name	COM2
Status	OK
PNP Device ID	ACPI\\PNP0501\\2
Maximum Input Buffer Size	0
Maximum Output Buffer Size	False
Settable Baud Rate	True
Settable Data Bits	True
Settable Flow Control	True
Settable Parity	True
Settable Parity Check	True
Settable Stop Bits	True
Settable RLSD	True
Supports RLSD	True
Supports 16 Bit Mode	False
Supports Special Characters	False
Baud Rate	9600
Bits/Byte	8
Stop Bits	1
Parity	None
Busy	0
Abort Read/Write on Error	0
Binary Mode Enabled-1	
Continue XMit on XOff	0
CTS Outflow Control	0

Discard NULL Bytes 0
 DSR Outflow Control 0
 DSR Sensitivity 0
 DTR Flow Control Type Enable
 EOF Character 0
 Error Replace Character 0
 Error Replacement Enabled 0
 Event Character 0
 Parity Check Enabled 0
 RTS Flow Control Type Enable
 XOff Character 19
 XOffXMit Threshold 512
 XOn Character 17
 XOnXMit Threshold 2048
 XOnXOff InFlow Control 0
 XOnXOff OutFlow Control 0
 IRQ Number 3
 I/O Port 0x02F8-0x02FF
 Driver c:\winnt\system32\drivers\serial.sys (62448, 5.00.2134.1)

[Parallel]

Item Value
 Name LPT1
 PNP Device ID ACPI\PNP0400\1

[Storage]

[Following are sub-categories of this main category]

[Drives]

Item Value
 Drive A:
 Description 3 1/2 Inch Floppy Drive

Drive C:
 Description Local Fixed Disk
 Compressed False
 File System NTFS
 Size 8.47 GB (9,097,424,896 bytes)
 Free Space 5.58 GB (5,990,879,232 bytes)
 Volume Name
 Volume Serial Number 9CD19247
 Partition Disk #0, Partition #0
 Partition Size 8.47 GB (9,097,428,992 bytes)
 Starting Offset 16384 bytes
 Drive Description Disk drive
 Drive Manufacturer (Standard disk drives)
 Drive Model IBM ServeRAID Failover Virtual Disk Device
 Drive BytesPerSector 512
 Drive MediaLoaded True
 Drive MediaType Fixed hard disk media
 Drive Partitions 1
 Drive SCSI Bus 0
 Drive SCSI Logical Unit 0
 Drive SCSI Port 3
 Drive SCSI Target Id 0
 Drive SectorsPerTrack 32
 Drive Size 9099542528 bytes
 Drive TotalCylinders 4339
 Drive TotalSectors 17772544
 Drive TotalTracks 555392
 Drive TracksPerCylinder 128

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 False
 File System NTFS
 Size 90.05 GB (96,694,558,720 bytes)
 Free Space 56.16 GB (60,303,040,512 bytes)
 Volume Name backup1
 Volume Serial Number E4F6BD1C
 Partition Disk #2, Partition #0
 Partition Size 130.08 GB (139,669,919,744 bytes)
 Starting Offset 2097152 bytes
 Drive Description Disk drive
 Drive Manufacturer (Standard disk drives)
 Drive Model IBM ServeRAID Failover Virtual Disk Device
 Drive BytesPerSector 512
 Drive MediaLoaded True
 Drive MediaType Fixed hard disk media
 Drive Partitions 3
 Drive SCSI Bus 0
 Drive SCSI Logical Unit 0
 Drive SCSI Port 4
 Drive SCSI Target Id 0
 Drive SectorsPerTrack 12
 Drive Size 136492941312 bytes
 Drive TotalCylinders 213612
 Drive TotalSectors 266587776
 Drive TotalTracks 22215648
 Drive TracksPerCylinder 104

Drive G:
 Description Local Fixed Disk
 Compressed False
 File System NTFS
 Size 90.05 GB (96,694,558,720 bytes)
 Free Space 56.16 GB (60,303,040,512 bytes)
 Volume Name backup2
 Volume Serial Number CC0DBDFC
 Partition Disk #4, Partition #0
 Partition Size 130.08 GB (139,669,919,744 bytes)
 Starting Offset 2097152 bytes
 Drive Description Disk drive
 Drive Manufacturer (Standard disk drives)
 Drive Model IBM ServeRAID Failover Virtual Disk Device
 Drive BytesPerSector 512
 Drive MediaLoaded True
 Drive MediaType Fixed hard disk media
 Drive Partitions 3
 Drive SCSI Bus 0
 Drive SCSI Logical Unit 0
 Drive SCSI Port 5
 Drive SCSI Target Id 0
 Drive SectorsPerTrack 12
 Drive Size 136492941312 bytes
 Drive TotalCylinders 213612
 Drive TotalSectors 266587776
 Drive TotalTracks 22215648
 Drive TracksPerCylinder 104

Drive H:
 Description Local Fixed Disk

Compressed False
 File System NTFS
 Size 90.05 GB (96,694,558,720 bytes)
 Free Space 56.16 GB (60,303,106,048 bytes)
 Volume Name backup3
 Volume Serial Number D01B3CAB
 Partition Disk #6, Partition #0
 Partition Size 130.08 GB (139,669,919,744 bytes)
 Starting Offset 2097152 bytes
 Drive Description Disk drive
 Drive Manufacturer (Standard disk drives)
 Drive Model IBM ServeRAID Failover Virtual Disk Device
 Drive BytesPerSector 512
 Drive MediaLoaded True
 Drive MediaType Fixed hard disk media
 Drive Partitions 3
 Drive SCSI Bus 0
 Drive SCSI LogicalUnit 0
 Drive SCSI Port 6
 Drive SCSI TargetId 0
 Drive SectorsPerTrack 12
 Drive Size 136492941312 bytes
 Drive TotalCylinders 213612
 Drive TotalSectors 266587776
 Drive TotalTracks 22215648
 Drive TracksPerCylinder 104

Drive I:
 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 J:
 Description Local Fixed Disk
 Compressed False
 File System NTFS
 Size 90.05 GB (96,694,558,720 bytes)
 Free Space 89.92 GB (96,550,621,184 bytes)
 Volume Name MDFfiles
 Volume Serial Number D43717C8
 Partition Disk #10, Partition #0
 Partition Size 130.08 GB (139,669,919,744 bytes)
 Starting Offset 2097152 bytes
 Drive Description Disk drive
 Drive Manufacturer (Standard disk drives)
 Drive Model IBM ServeRAID Failover Virtual Disk Device
 Drive BytesPerSector 512
 Drive MediaLoaded True
 Drive MediaType Fixed hard disk media
 Drive Partitions 3
 Drive SCSI Bus 0
 Drive SCSI LogicalUnit 0
 Drive SCSI Port 8
 Drive SCSI TargetId 0
 Drive SectorsPerTrack 63
 Drive Size 136490296320 bytes
 Drive TotalCylinders 16594
 Drive TotalSectors 266582610
 Drive TotalTracks 4231470
 Drive TracksPerCylinder 255

Drive K:
 Description Local Fixed Disk
 Compressed False

File System NTFS
 Size 90.05 GB (96,694,558,720 bytes)
 Free Space 56.16 GB (60,303,106,048 bytes)
 Volume Name backup4
 Volume Serial Number 1027E264
 Partition Disk #8, Partition #0
 Partition Size 130.08 GB (139,669,919,744 bytes)
 Starting Offset 2097152 bytes
 Drive Description Disk drive
 Drive Manufacturer (Standard disk drives)
 Drive Model IBM ServeRAID Failover Virtual Disk Device
 Drive BytesPerSector 512
 Drive MediaLoaded True
 Drive MediaType Fixed hard disk media
 Drive Partitions 3
 Drive SCSI Bus 0
 Drive SCSI LogicalUnit 0
 Drive SCSI Port 7
 Drive SCSI TargetId 0
 Drive SectorsPerTrack 63
 Drive Size 136490296320 bytes
 Drive TotalCylinders 16594
 Drive TotalSectors 266582610
 Drive TotalTracks 4231470
 Drive TracksPerCylinder 255

Drive L:
 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 M:
 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 N:
 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 O:
 Description Local Fixed Disk
 Compressed Not Available
 File System Not Available
 Size Not Available
 Free Space Not Available
 Volume Name Not Available
 Volume Serial Number Not Available

Drive P:
 Description Local Fixed Disk
 Compressed Not Available
 File System Not Available
 Size Not Available
 Free Space Not Available

Volume Name Not Available
Volume Serial Number Not Available

Drive Q:
Description Local Fixed Disk
Compressed Not Available
File System Not Available
Size Not Available
Free Space Not Available
Volume Name Not Available
Volume Serial Number Not Available

Drive R:
Description Local Fixed Disk
Compressed Not Available
File System Not Available
Size Not Available
Free Space Not Available
Volume Name Not Available
Volume Serial Number Not Available

Drive S:
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 T:
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 U:
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 V:
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

[SCSI]

Item Value
Name IBM Netfinity ServeRAID 4H Controller
Caption IBM Netfinity ServeRAID 4H Controller
Driver ipsraidn
Status OK
PNP Device ID
PCI\VEN_1014&DEV_002E&SUBSYS_022E1014&REV_10\3&267
A616A&0&50

Device ID
PCI\VEN_1014&DEV_002E&SUBSYS_022E1014&REV_10\3&267
A616A&0&50
Device Map Not Available
Index Not Available
Max Number Controlled Not Available
IRQ Number 25
I/O Port 0x2200-0x22FF
Driver c:\winnt\system32\drivers\ipsraidn.sys (102807, 4.70.08)

Name IBM Netfinity ServeRAID 4H Controller
Caption IBM Netfinity ServeRAID 4H Controller
Driver ipsraidn
Status OK
PNP Device ID
PCI\VEN_1014&DEV_002E&SUBSYS_022E1014&REV_10\3&267
A616A&0&58
Device ID
PCI\VEN_1014&DEV_002E&SUBSYS_022E1014&REV_10\3&267
A616A&0&58

Device Map Not Available
Index Not Available
Max Number Controlled Not Available
IRQ Number 26
I/O Port 0x2300-0x23FF
Driver c:\winnt\system32\drivers\ipsraidn.sys (102807, 4.70.08)

Name IBM Netfinity ServeRAID 4H Controller
Caption IBM Netfinity ServeRAID 4H Controller
Driver ipsraidn
Status OK
PNP Device ID
PCI\VEN_1014&DEV_002E&SUBSYS_022E1014&REV_10\3&267
A616A&0&60
Device ID
PCI\VEN_1014&DEV_002E&SUBSYS_022E1014&REV_10\3&267
A616A&0&60

Device Map Not Available
Index Not Available
Max Number Controlled Not Available
IRQ Number 27
I/O Port 0x2400-0x24FF
Driver c:\winnt\system32\drivers\ipsraidn.sys (102807, 4.70.08)

Name Adaptec AIC-7896/AIC-7897 PCI Ultra2 SCSI Controller
Caption Adaptec AIC-7896/AIC-7897 PCI Ultra2 SCSI Controller
Driver aic78u2
Status OK
PNP Device ID
PCI\VEN_9005&DEV_005F&SUBSYS_080F9005&REV_00\3&267
A616A&0&70

Device ID
PCI\VEN_9005&DEV_005F&SUBSYS_080F9005&REV_00\3&267
A616A&0&70
Device Map Not Available
Index Not Available
Max Number Controlled Not Available
IRQ Number 51
I/O Port 0x2500-0x25FF
Driver c:\winnt\system32\drivers\aic78u2.sys (65168, v3.00a)

Name Adaptec AIC-7896/AIC-7897 PCI Ultra2 SCSI Controller
Caption Adaptec AIC-7896/AIC-7897 PCI Ultra2 SCSI Controller
Driver aic78u2
Status OK
PNP Device ID
PCI\VEN_9005&DEV_005F&SUBSYS_080F9005&REV_00\3&267
A616A&0&71

Device ID
PCI\VEN_9005&DEV_005F&SUBSYS_080F9005&REV_00\3&267
A616A&0&71
Device Map Not Available
Index Not Available
Max Number Controlled Not Available
IRQ Number 51
I/O Port 0x2600-0x26FF
Driver c:\winnt\system32\drivers\aic78u2.sys (65168, v3.00a)

Name IBM Netfinity ServeRAID 4H Controller
Caption IBM Netfinity ServeRAID 4H Controller
Driver ipsraidn
Status OK
PNP Device ID
PCI\VEN_1014&DEV_002E&SUBSYS_022E1014&REV_10\3&172
E68DD&0&08
Device ID
PCI\VEN_1014&DEV_002E&SUBSYS_022E1014&REV_10\3&172
E68DD&0&08
Device Map Not Available
Index Not Available
Max Number Controlled Not Available
IRQ Number 16
I/O Port 0x6000-0xAFFF
Driver c:\winnt\system32\drivers\ipsraidn.sys (102807, 4.70.08)

Name IBM Netfinity ServeRAID 4H Controller
Caption IBM Netfinity ServeRAID 4H Controller
Driver ipsraidn
Status OK
PNP Device ID
PCI\VEN_1014&DEV_002E&SUBSYS_022E1014&REV_10\3&172
E68DD&0&10
Device ID
PCI\VEN_1014&DEV_002E&SUBSYS_022E1014&REV_10\3&172
E68DD&0&10
Device Map Not Available
Index Not Available
Max Number Controlled Not Available
IRQ Number 17
I/O Port 0x6100-0x61FF
Driver c:\winnt\system32\drivers\ipsraidn.sys (102807, 4.70.08)

Name IBM Netfinity ServeRAID 4H Controller
Caption IBM Netfinity ServeRAID 4H Controller
Driver ipsraidn
Status OK
PNP Device ID
PCI\VEN_1014&DEV_002E&SUBSYS_022E1014&REV_10\3&172
E68DD&0&18
Device ID
PCI\VEN_1014&DEV_002E&SUBSYS_022E1014&REV_10\3&172
E68DD&0&18
Device Map Not Available
Index Not Available
Max Number Controlled Not Available
IRQ Number 18
I/O Port 0x6200-0x62FF
Driver c:\winnt\system32\drivers\ipsraidn.sys (102807, 4.70.08)

Name IBM Netfinity ServeRAID 4H Controller
Caption IBM Netfinity ServeRAID 4H Controller
Driver ipsraidn
Status OK
PNP Device ID
PCI\VEN_1014&DEV_002E&SUBSYS_022E1014&REV_10\3&172
E68DD&0&20

Device ID
PCI\VEN_1014&DEV_002E&SUBSYS_022E1014&REV_10\3&172
E68DD&0&20
Device Map Not Available
Index Not Available
Max Number Controlled Not Available
IRQ Number 19
I/O Port 0x6300-0x63FF
Driver c:\winnt\system32\drivers\ipsraidn.sys (102807, 4.70.08)

Name IBM Netfinity ServeRAID 4H Controller
Caption IBM Netfinity ServeRAID 4H Controller
Driver ipsraidn
Status OK
PNP Device ID
PCI\VEN_1014&DEV_002E&SUBSYS_022E1014&REV_10\3&146
CA173&0&30
Device ID
PCI\VEN_1014&DEV_002E&SUBSYS_022E1014&REV_10\3&146
CA173&0&30
Device Map Not Available
Index Not Available
Max Number Controlled Not Available
IRQ Number 21
I/O Port 0xD000-0xFFFF
Driver c:\winnt\system32\drivers\ipsraidn.sys (102807, 4.70.08)

Name IBM Netfinity ServeRAID Failover Virtual Adapter
Caption IBM Netfinity ServeRAID Failover Virtual Adapter
Driver twintail
Status OK
PNP Device ID ROOT\TWINTAIL\0000
Device ID ROOT\TWINTAIL\0000
Device Map Not Available
Index Not Available
Max Number Controlled Not Available
Driver c:\winnt\system32\drivers\twintail.sys (26032, 4.70.06)

[Printing]

Name Port Name Server Name
No printing information

[Problem Devices]

Device	PNP Device ID	Error Code
Other PCI Bridge Device	PCI\VEN_1014&DEV_00DC&SUBSYS_00DC1014&REV_02\3&26 7A616A&0&10	28
Not Available	ACPI\IBM37D0\4&43B47AD&0	28

[USB]

Device	PNP Device ID
Intel 82371AB/EB PCI to USB Universal Host Controller	PCI\VEN_8086&DEV_7112&SUBSYS_00000000&REV_01\3&267 A616A&0&7A
USB Root Hub	USB\ROOT_HUB\4&B5B4E1B&0

March 6, 2001 9:13:53 AM EST

Configuration summary

Server name.....rtnode01
ServeRAID Manager Version.....4.00.34

Number of controllers.....8
Operating system.....Windows NT
Device driver version.....4.70.08

Information for controller 1

Controller type.....ServeRAID-4H
BIOS version.....4.70.06
Firmware version.....joe.05
Physical slot.....10
Battery-backup cache.....Installed
Read-ahead cache mode.....Adaptive
Stripe-unit size.....64K
Rebuild rate.....High
Hot-swap rebuild.....Enabled
Data scrubbing.....Enabled
Auto-synchronization.....Enabled
Clustering.....Disabled
Unattended mode.....Disabled
Number of arrays.....2
Number of logical drives.....2
Number of hot-spare drives.....0
Number of ready drives.....0

Array A

Array identifier.....A
Array size in MB.....17356
Free space in MB.....0
Number of logical drives.....1
Number of physical drives.....2

Logical drives in array A

Logical drive.....1
Array letter.....A
State.....Okay
RAID level.....1
Data space in MB.....8678
Parity space in MB.....8678
Date created.....12/15/00
Write-cache mode.....Write through
Merge group number.....207
Merge group state.....Non-shared

Physical drives in array A

Type.....Hard disk drive
Channel.....1
SCSI ID.....0
Size in MB.....8678
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....DMVS09D
Serial number.....F80328F4
Firmware level.....0180
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....0
Size in MB.....8678
State.....Online
Array letter.....A
Vendor.....IBM-PSG

Product or model number.....DMVS09D
Serial number.....F802FA17
Firmware level.....0180
PFA error.....No
Write-cache mode.....Write through

Spanned array 1

Array identifier.....1
Array size in MB.....138856
Free space in MB.....0
Number of logical drives.....1
Number of physical drives.....8

Arrays in spanned array 1

Array identifier.....B
Array size in MB.....34714
Number of physical drives.....2

Array identifier.....C
Array size in MB.....34714
Number of physical drives.....2

Array identifier.....D
Array size in MB.....34714
Number of physical drives.....2

Array identifier.....E
Array size in MB.....34714
Number of physical drives.....2

Physical drives in array B

Type.....Hard disk drive
Channel.....1
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR723266
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR713507
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Physical drives in array C

Type.....Hard disk drive
Channel.....1
SCSI ID.....2
Size in MB.....17357

State.....Online
Array letter.....C
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR730891
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....C
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR706759
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Physical drives in array D

Type.....Hard disk drive
Channel.....1
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....D
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR730712
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....D
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR730752
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Physical drives in array E

Type.....Hard disk drive
Channel.....1
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....E
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR673074
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive

Channel.....2
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....E
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR696290
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Logical drives in spanned array 1

Logical drive.....2
Spanned array number1
State.....Okay
RAID level.....10
Data space in MB.....69428
Parity space in MB.....69428
Date created.....2/6/01
Write-cache mode.....Write through
Merge group number.....207
Merge group state.....Non-shared

SCSI channel 1

Type.....Hard disk drive
Channel.....1
SCSI ID.....0
Size in MB.....8678
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....DMVS09D
Serial number.....F80328F4
Firmware level.....0180
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR723266
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....C
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR730891
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....D
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR730712
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....E
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR673074
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Enclosure
Channel.....1
SCSI ID.....15
Enclosure status.....Okay
Fan 1 status.....Okay
Fan 2 status.....Okay
Power supply 1 status.....Okay
Power supply 2 status.....Okay
Temperature status.....Normal
Vendor.....IBM
Product or model number.....EXP200
Serial number.....94614613
Firmware level.....1020
Enclosure ID.....0
FRU type.....MIDPLANE
FRU vendor.....IBM
FRU date of manufacture.....12/1999
FRU part number.....37L0061
FRU serial number.....23A1539
FRU type.....ESM_CARD
FRU vendor.....IBM
FRU date of manufacture.....11/1999
FRU part number.....37L0063
FRU serial number.....1T94614613

SCSI channel 2

Type.....Hard disk drive
Channel.....2
SCSI ID.....0
Size in MB.....8678
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....DMVS09D
Serial number.....F802FA17
Firmware level.....0180
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive

Channel.....2
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR713507
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....C
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR706759
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....D
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR730752
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....E
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR696290
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Enclosure
Channel.....2
SCSI ID.....15
Enclosure status.....Okay
Fan 1 status.....Okay
Fan 2 status.....Okay
Power supply 1 status.....Okay
Power supply 2 status.....Okay
Temperature status.....Normal
Vendor.....IBM
Product or model number.....EXP200
Serial number.....94614613
Firmware level.....1020
Enclosure ID.....0
FRU type.....MIDPLANE
FRU vendor.....IBM
FRU date of manufacture.....12/1999

FRU part number.....37L0061
FRU serial number.....23A1539
FRU type.....ESM_CARD
FRU vendor.....IBM
FRU date of manufacture.....11/1999
FRU part number.....37L0063
FRU serial number.....1T94614613

SCSI channel 3

SCSI channel 4

End of the configuration information for controller 1

Information for controller 2

Controller type.....ServeRAID-4H
BIOS version.....4.70.06
Firmware version.....joe.05
Physical slot.....11
Battery-backup cache.....Installed
Read-ahead cache mode.....Adaptive
Stripe-unit size.....64K
Rebuild rate.....High
Hot-swap rebuild.....Enabled
Data scrubbing.....Enabled
Auto-synchronization.....Enabled
Clustering.....Disabled
Unattended mode.....Disabled
Number of arrays.....2
Number of logical drives.....2
Number of hot-spare drives.....0
Number of ready drives.....0

Array A

Array identifier.....A
Array size in MB.....260355
Free space in MB.....15
Number of logical drives.....1
Number of physical drives.....15

Logical drives in array A

Logical drive.....1
Array letter.....A
State.....Okay
RAID level.....1E
Data space in MB.....130170
Parity space in MB.....130170
Date created.....12/15/00
Write-cache mode.....Write back
Merge group number.....207
Merge group state.....Non-shared

Physical drives in array A

Type.....Hard disk drive
Channel.....1
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203

Serial number.....LR778524
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR730150
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR726573
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR706689
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR721549
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR521045
Firmware level.....B227
PFA error.....No

Write-cache mode.....Write through
Type.....Hard disk drive
Channel.....2
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR701790
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR691402
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR678806
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR578394
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR696506
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive

Channel.....3
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR600148
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR703988
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR684116
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR704041
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Array B

Array identifier.....B
Array size in MB.....260355
Free space in MB.....15
Number of logical drives.....1
Number of physical drives.....15

Logical drives in array B

Logical drive.....2
Array letter.....B
State.....Okay
RAID level.....1E
Data space in MB.....130170
Parity space in MB.....130170
Date created.....12/15/00

Write-cache mode.....Write back
Merge group number.....207
Merge group state.....Non-shared

Physical drives in array B

Type.....Hard disk drive
Channel.....1
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR706137
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR726570
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR704053
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR725119
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....12
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR594070

Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR678317
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR580451
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR702657
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR594592
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....12
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR593150
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR704531
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR695280
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR699182
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR696050
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....12
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR595598
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

SCSI channel 1

Type.....Hard disk drive
Channel.....1
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR778524
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR730150
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR726573
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR706689
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR721549
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive

Channel.....1
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR706137
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR726570
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR704053
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR725119
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....12
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR594070
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Enclosure
Channel.....1
SCSI ID.....15
Enclosure status.....Okay

Fan 1 status.....Okay
Fan 2 status.....Okay
Power supply 1 status.....Okay
Power supply 2 status.....Okay
Temperature status.....Normal
Vendor.....IBM
Product or model number.....EXP200
Serial number.....94212241
Firmware level.....1020
Enclosure ID.....0
FRU type.....MIDPLANE
FRU vendor.....IBM
FRU date of manufacture.....12/1999
FRU part number.....37L0061
FRU serial number.....23A1456
FRU type.....ESM_CARD
FRU vendor.....IBM
FRU date of manufacture.....11/1999
FRU part number.....37L0063
FRU serial number.....1T94212241

SCSI channel 2

Type.....Hard disk drive
Channel.....2
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR521045
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR701790
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR691402
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A

Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR678806
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR578394
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR678317
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR580451
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR702657
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR594592

Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....12
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR593150
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Enclosure
Channel.....2
SCSI ID.....15
Enclosure status.....Okay
Fan 1 status.....Okay
Fan 2 status.....Okay
Power supply 1 status.....Okay
Power supply 2 status.....Okay
Temperature status.....Normal
Vendor.....IBM
Product or model number.....EXP200
Serial number.....94920493
Firmware level.....1020
Enclosure ID.....0
FRU type.....MIDPLANE
FRU vendor.....IBM
FRU date of manufacture.....12/1999
FRU part number.....37L0061
FRU serial number.....23A1727
FRU type.....ESM_CARD
FRU vendor.....IBM
FRU date of manufacture.....12/1999
FRU part number.....37L0063
FRU serial number.....1T94920493

SCSI channel 3

Type.....Hard disk drive
Channel.....3
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR696506
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR600148
Firmware level.....B227
PFA error.....No

Write-cache mode.....Write through
Type.....Hard disk drive
Channel.....3
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR703988
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR684116
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR704041
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR704531
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR695280
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive

Channel.....3
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR699182
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR696050
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....12
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR595598
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Enclosure
Channel.....3
SCSI ID.....15
Enclosure status.....Okay
Fan 1 status.....Okay
Fan 2 status.....Okay
Power supply 1 status.....Okay
Power supply 2 status.....Okay
Temperature status.....Normal
Vendor.....IBM
Product or model number.....EXP200
Serial number.....94212431
Firmware level.....1020
Enclosure ID.....0
FRU type.....MIDPLANE
FRU vendor.....IBM
FRU date of manufacture.....12/1999
FRU part number.....37L0061
FRU serial number.....23A1677
FRU type.....ESM_CARD
FRU vendor.....IBM
FRU date of manufacture.....11/1999
FRU part number.....37L0063
FRU serial number.....1T94212431

SCSI channel 4

End of the configuration information for controller 2

Information for controller 3

Controller type.....ServeRAID-4H
BIOS version.....4.70.06
Firmware version.....joe.05
Physical slot.....12
Battery-backup cache.....Installed
Read-ahead cache mode.....Adaptive
Stripe-unit size.....64K
Rebuild rate.....High
Hot-swap rebuild.....Enabled
Data scrubbing.....Enabled
Auto-synchronization.....Enabled
Clustering.....Disabled
Unattended mode.....Disabled
Number of arrays.....2
Number of logical drives.....2
Number of hot-spare drives.....0
Number of ready drives.....0

Array A

Array identifier.....A
Array size in MB.....260355
Free space in MB.....15
Number of logical drives.....1
Number of physical drives.....15

Logical drives in array A

Logical drive.....1
Array letter.....A
State.....Okay
RAID level.....1E
Data space in MB.....130170
Parity space in MB.....130170
Date created.....12/15/00
Write-cache mode.....Write back
Merge group number.....207
Merge group state.....Non-shared

Physical drives in array A

Type.....Hard disk drive
Channel.....1
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR675445
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR593068
Firmware level.....B227

PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR713614
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR696104
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR593296
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR769309
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR678323
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR696041
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR593147
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR701200
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR599285
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR598945
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....2

Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR593176
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR723640
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR701825
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Array B

Array identifier.....B
Array size in MB.....260355
Free space in MB.....15
Number of logical drives.....1
Number of physical drives.....15

Logical drives in array B

Logical drive.....2
Array letter.....B
State.....Okay
RAID level.....1E
Data space in MB.....130170
Parity space in MB.....130170
Date created.....12/15/00
Write-cache mode.....Write back
Merge group number.....207
Merge group state.....Non-shared

Physical drives in array B

Type.....Hard disk drive
Channel.....1
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203

Serial number.....LR678322
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR589165
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR684663
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR591188
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....12
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR699117
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR720965
Firmware level.....B227
PFA error.....No

Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR702864
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR634241
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR701684
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....12
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR580323
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR695979
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive

Channel.....3
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR592822
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR678320
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR704848
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....12
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR579799
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

SCSI channel 1

Type.....Hard disk drive
Channel.....1
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR675445
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR593068
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR713614
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR696104
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR593296
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR678322
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....9

Size in MB.....17357
 State.....Online
 Array letter.....B
 Vendor.....IBM-PSG
 Product or model number.....ST318203
 Serial number.....LR589165
 Firmware level.....B227
 PFA error.....No
 Write-cache mode.....Write through

 Type.....Hard disk drive
 Channel.....1
 SCSI ID.....10
 Size in MB.....17357
 State.....Online
 Array letter.....B
 Vendor.....IBM-PSG
 Product or model number.....ST318203
 Serial number.....LR684663
 Firmware level.....B227
 PFA error.....No
 Write-cache mode.....Write through

 Type.....Hard disk drive
 Channel.....1
 SCSI ID.....11
 Size in MB.....17357
 State.....Online
 Array letter.....B
 Vendor.....IBM-PSG
 Product or model number.....ST318203
 Serial number.....LR591188
 Firmware level.....B227
 PFA error.....No
 Write-cache mode.....Write through

 Type.....Hard disk drive
 Channel.....1
 SCSI ID.....12
 Size in MB.....17357
 State.....Online
 Array letter.....B
 Vendor.....IBM-PSG
 Product or model number.....ST318203
 Serial number.....LR699117
 Firmware level.....B227
 PFA error.....No
 Write-cache mode.....Write through

 Type.....Enclosure
 Channel.....1
 SCSI ID.....15
 Enclosure status.....Okay
 Fan 1 status.....Okay
 Fan 2 status.....Okay
 Power supply 1 status.....Okay
 Power supply 2 status.....Okay
 Temperature status.....Normal
 Vendor.....IBM
 Product or model number.....EXP200
 Serial number.....94614620
 Firmware level.....1020
 Enclosure ID.....0
 FRU type.....MIDPLANE
 FRU vendor.....IBM
 FRU date of manufacture.....12/1999
 FRU part number.....37L0061
 FRU serial number.....23A1681

FRU type.....ESM_CARD
 FRU vendor.....IBM
 FRU date of manufacture.....11/1999
 FRU part number.....37L0063
 FRU serial number.....1T94614620

 SCSI channel 2

 Type.....Hard disk drive
 Channel.....2
 SCSI ID.....0
 Size in MB.....17357
 State.....Online
 Array letter.....A
 Vendor.....IBM-PSG
 Product or model number.....ST318203
 Serial number.....LR769309
 Firmware level.....B227
 PFA error.....No
 Write-cache mode.....Write through

 Type.....Hard disk drive
 Channel.....2
 SCSI ID.....1
 Size in MB.....17357
 State.....Online
 Array letter.....A
 Vendor.....IBM-PSG
 Product or model number.....ST318203
 Serial number.....LR678323
 Firmware level.....B227
 PFA error.....No
 Write-cache mode.....Write through

 Type.....Hard disk drive
 Channel.....2
 SCSI ID.....2
 Size in MB.....17357
 State.....Online
 Array letter.....A
 Vendor.....IBM-PSG
 Product or model number.....ST318203
 Serial number.....LR696041
 Firmware level.....B227
 PFA error.....No
 Write-cache mode.....Write through

 Type.....Hard disk drive
 Channel.....2
 SCSI ID.....3
 Size in MB.....17357
 State.....Online
 Array letter.....A
 Vendor.....IBM-PSG
 Product or model number.....ST318203
 Serial number.....LR593147
 Firmware level.....B227
 PFA error.....No
 Write-cache mode.....Write through

 Type.....Hard disk drive
 Channel.....2
 SCSI ID.....4
 Size in MB.....17357
 State.....Online
 Array letter.....A
 Vendor.....IBM-PSG
 Product or model number.....ST318203

Serial number.....LR701200
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR720965
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR702864
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR634241
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR701684
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....12
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR580323
Firmware level.....B227
PFA error.....No

Write-cache mode.....Write through

Type.....Enclosure
Channel.....2
SCSI ID.....15
Enclosure status.....Okay
Fan 1 status.....Okay
Fan 2 status.....Okay
Power supply 1 status.....Okay
Power supply 2 status.....Okay
Temperature status.....Normal
Vendor.....IBM
Product or model number.....EXP200
Serial number.....94614501
Firmware level.....1020
Enclosure ID.....0
FRU type.....MIDPLANE
FRU vendor.....IBM
FRU date of manufacture.....12/1999
FRU part number.....37L0061
FRU serial number.....23A1432
FRU type.....ESM_CARD
FRU vendor.....IBM
FRU date of manufacture.....11/1999
FRU part number.....37L0063
FRU serial number.....1T94614501

SCSI channel 3

Type.....Hard disk drive
Channel.....3
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR599285
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR598945
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR593176
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR723640
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR701825
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR695979
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR592822
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR678320
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....11

Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR704848
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....12
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR579799
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Enclosure
Channel.....3
SCSI ID.....15
Enclosure status.....Okay
Fan 1 status.....Okay
Fan 2 status.....Okay
Power supply 1 status.....Okay
Power supply 2 status.....Okay
Temperature status.....Normal
Vendor.....IBM
Product or model number.....EXP200
Serial number.....94212081
Firmware level.....1020
Enclosure ID.....0
FRU type.....MIDPLANE
FRU vendor.....IBM
FRU date of manufacture.....12/1999
FRU part number.....37L0061
FRU serial number.....23A1459
FRU type.....ESM_CARD
FRU vendor.....IBM
FRU date of manufacture.....11/1999
FRU part number.....37L0063
FRU serial number.....1T94212081

SCSI channel 4

End of the configuration information for controller 3

Information for controller 4

Controller type.....ServeRAID-4H
BIOS version.....4.70.06
Firmware version.....joe.05
Physical slot.....1
Battery-backup cache.....Installed
Read-ahead cache mode.....Adaptive
Stripe-unit size.....64K
Rebuild rate.....High
Hot-swap rebuild.....Enabled
Data scrubbing.....Enabled
Auto-synchronization.....Enabled
Clustering.....Disabled

Unattended mode.....Disabled
Number of arrays.....2
Number of logical drives.....2
Number of hot-spare drives.....0
Number of ready drives.....0

Array A

Array identifier.....A
Array size in MB.....260355
Free space in MB.....15
Number of logical drives.....1
Number of physical drives.....15

Logical drives in array A

Logical drive.....1
Array letter.....A
State.....Okay
RAID level.....1E
Data space in MB.....130170
Parity space in MB.....130170
Date created.....12/15/00
Write-cache mode.....Write back
Merge group number.....207
Merge group state.....Non-shared

Physical drives in array A

Type.....Hard disk drive
Channel.....1
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR678981
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR684664
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR762032
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR701600
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR695906
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR678475
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR594735
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR583115
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2

SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR690440
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR766827
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR768736
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR766292
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR592503
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....3
Size in MB.....17357
State.....Online

Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR580478
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR769210
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Array B

Array identifier.....B
Array size in MB.....260355
Free space in MB.....15
Number of logical drives.....1
Number of physical drives.....15

Logical drives in array B

Logical drive.....2
Array letter.....B
State.....Okay
RAID level.....1E
Data space in MB.....130170
Parity space in MB.....130170
Date created.....12/15/00
Write-cache mode.....Write back
Merge group number.....207
Merge group state.....Non-shared

Physical drives in array B

Type.....Hard disk drive
Channel.....1
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR594825
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR594725
Firmware level.....B227

PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR695657
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR569421
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....12
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR565533
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR680266
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR700040
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR702450
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR678467
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....12
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR577997
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR759724
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR600239
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....10

Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR769370
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR569903
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....12
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR759631
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

SCSI channel 1

Type.....Hard disk drive
Channel.....1
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR678981
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR684664
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1

SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR762032
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR701600
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR695906
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR594825
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR594725
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....10
Size in MB.....17357
State.....Online

Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR695657
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR569421
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....12
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR565533
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Enclosure
Channel.....1
SCSI ID.....15
Enclosure status.....Okay
Fan 1 status.....Okay
Fan 2 status.....Okay
Power supply 1 status.....Okay
Power supply 2 status.....Okay
Temperature status.....Normal
Vendor.....IBM
Product or model number.....EXP200
Serial number.....94718893
Firmware level.....1020
Enclosure ID.....0
FRU type.....MIDPLANE
FRU vendor.....IBM
FRU date of manufacture.....12/1999
FRU part number.....37L0061
FRU serial number.....23A1732
FRU type.....ESM_CARD
FRU vendor.....IBM
FRU date of manufacture.....12/1999
FRU part number.....37L0063
FRU serial number.....1T94718893

SCSI channel 2

Type.....Hard disk drive
Channel.....2
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG

Product or model number.....ST318203
Serial number.....LR678475
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR594735
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR583115
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR690440
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR766827
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR680266
Firmware level.....B227

PFA error.....No
 Write-cache mode.....Write through

 Type.....Hard disk drive
 Channel.....2
 SCSI ID.....9
 Size in MB.....17357
 State.....Online
 Array letter.....B
 Vendor.....IBM-PSG
 Product or model number.....ST318203
 Serial number.....LR700040
 Firmware level.....B227
 PFA error.....No
 Write-cache mode.....Write through

 Type.....Hard disk drive
 Channel.....2
 SCSI ID.....10
 Size in MB.....17357
 State.....Online
 Array letter.....B
 Vendor.....IBM-PSG
 Product or model number.....ST318203
 Serial number.....LR702450
 Firmware level.....B227
 PFA error.....No
 Write-cache mode.....Write through

 Type.....Hard disk drive
 Channel.....2
 SCSI ID.....11
 Size in MB.....17357
 State.....Online
 Array letter.....B
 Vendor.....IBM-PSG
 Product or model number.....ST318203
 Serial number.....LR678467
 Firmware level.....B227
 PFA error.....No
 Write-cache mode.....Write through

 Type.....Hard disk drive
 Channel.....2
 SCSI ID.....12
 Size in MB.....17357
 State.....Online
 Array letter.....B
 Vendor.....IBM-PSG
 Product or model number.....ST318203
 Serial number.....LR577997
 Firmware level.....B227
 PFA error.....No
 Write-cache mode.....Write through

 Type.....Enclosure
 Channel.....2
 SCSI ID.....15
 Enclosure status.....Okay
 Fan 1 status.....Okay
 Fan 2 status.....Okay
 Power supply 1 status.....Okay
 Power supply 2 status.....Okay
 Temperature status.....Normal
 Vendor.....IBM
 Product or model number.....EXP200
 Serial number.....94920476
 Firmware level.....1020

Enclosure ID.....0
 FRU type.....MIDPLANE
 FRU vendor.....IBM
 FRU date of manufacture.....12/1999
 FRU part number.....37L0061
 FRU serial number.....23A1742
 FRU type.....ESM_CARD
 FRU vendor.....IBM
 FRU date of manufacture.....12/1999
 FRU part number.....37L0063
 FRU serial number.....1T94920476

 SCSI channel 3

 Type.....Hard disk drive
 Channel.....3
 SCSI ID.....0
 Size in MB.....17357
 State.....Online
 Array letter.....A
 Vendor.....IBM-PSG
 Product or model number.....ST318203
 Serial number.....LR768736
 Firmware level.....B227
 PFA error.....No
 Write-cache mode.....Write through

 Type.....Hard disk drive
 Channel.....3
 SCSI ID.....1
 Size in MB.....17357
 State.....Online
 Array letter.....A
 Vendor.....IBM-PSG
 Product or model number.....ST318203
 Serial number.....LR766292
 Firmware level.....B227
 PFA error.....No
 Write-cache mode.....Write through

 Type.....Hard disk drive
 Channel.....3
 SCSI ID.....2
 Size in MB.....17357
 State.....Online
 Array letter.....A
 Vendor.....IBM-PSG
 Product or model number.....ST318203
 Serial number.....LR592503
 Firmware level.....B227
 PFA error.....No
 Write-cache mode.....Write through

 Type.....Hard disk drive
 Channel.....3
 SCSI ID.....3
 Size in MB.....17357
 State.....Online
 Array letter.....A
 Vendor.....IBM-PSG
 Product or model number.....ST318203
 Serial number.....LR580478
 Firmware level.....B227
 PFA error.....No
 Write-cache mode.....Write through

 Type.....Hard disk drive
 Channel.....3

SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR769210
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR759724
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR600239
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR769370
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR569903
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....12
Size in MB.....17357
State.....Online

Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR759631
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Enclosure
Channel.....3
SCSI ID.....15
Enclosure status.....Okay
Fan 1 status.....Okay
Fan 2 status.....Okay
Power supply 1 status.....Okay
Power supply 2 status.....Okay
Temperature status.....Normal
Vendor.....IBM
Product or model number.....EXP200
Serial number.....94614611
Firmware level.....1020
Enclosure ID.....0
FRU type.....MIDPLANE
FRU vendor.....IBM
FRU date of manufacture.....12/1999
FRU part number.....37L0061
FRU serial number.....23A1680
FRU type.....ESM_CARD
FRU vendor.....IBM
FRU date of manufacture.....11/1999
FRU part number.....37L0063
FRU serial number.....1T94614611

SCSI channel 4

End of the configuration information for controller 4

Information for controller 5

Controller type.....ServeRAID-4H
BIOS version.....4.70.06
Firmware version.....joe.05
Physical slot.....2
Battery-backup cache.....Installed
Read-ahead cache mode.....Adaptive
Stripe-unit size.....64K
Rebuild rate.....High
Hot-swap rebuild.....Enabled
Data scrubbing.....Enabled
Auto-synchronization.....Enabled
Clustering.....Disabled
Unattended mode.....Disabled
Number of arrays.....2
Number of logical drives.....2
Number of hot-spare drives.....0
Number of ready drives.....0

Array A

Array identifier.....A
Array size in MB.....260355
Free space in MB.....15
Number of logical drives.....1
Number of physical drives.....15

Logical drives in array A

```

-----
Logical drive.....1
Array letter.....A
State.....Okay
RAID level.....1E
Data space in MB.....130170
Parity space in MB.....130170
Date created.....12/15/00
Write-cache mode.....Write back
Merge group number.....207
Merge group state.....Non-shared

Physical drives in array A
-----
Type.....Hard disk drive
Channel.....1
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR752492
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR735401
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR766197
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR769105
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive

```

```

Channel.....1
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR593911
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR595624
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR594556
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR758784
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR695895
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....4
Size in MB.....17357

```

State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR593565
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR762480
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR768077
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR589133
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR594818
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG

Product or model number.....ST318203
Serial number.....LR583612
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Array B

Array identifier.....B
Array size in MB.....260355
Free space in MB.....15
Number of logical drives.....1
Number of physical drives.....15

Logical drives in array B

Logical drive.....2
Array letter.....B
State.....Okay
RAID level.....1E
Data space in MB.....130170
Parity space in MB.....130170
Date created.....12/15/00
Write-cache mode.....Write back
Merge group number.....207
Merge group state.....Non-shared

Physical drives in array B

Type.....Hard disk drive
Channel.....1
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR768674
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR582934
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR575390
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR593918
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....12
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR766008
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR696877
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR769609
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR587676
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2

SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR589070
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....12
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR577283
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR588710
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR594204
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR595595
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....11
Size in MB.....17357
State.....Online

Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR599412
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....12
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR767125
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

SCSI channel 1

Type.....Hard disk drive
Channel.....1
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR752492
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR735401
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR766197
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....3
Size in MB.....17357

State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR769105
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR593911
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR768674
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR582934
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR575390
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG

Product or model number.....ST318203
Serial number.....LR593918
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....12
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR766008
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Enclosure
Channel.....1
SCSI ID.....15
Enclosure status.....Okay
Fan 1 status.....Okay
Fan 2 status.....Okay
Power supply 1 status.....Okay
Power supply 2 status.....Okay
Temperature status.....Normal
Vendor.....IBM
Product or model number.....EXP200
Serial number.....94007895
Firmware level.....1020
Enclosure ID.....0
FRU type.....MIDPLANE
FRU vendor.....IBM
FRU date of manufacture.....12/1999
FRU part number.....37L0061
FRU serial number.....23A1489
FRU type.....ESM_CARD
FRU vendor.....IBM
FRU date of manufacture.....11/1999
FRU part number.....37L0063
FRU serial number.....1T94007895

SCSI channel 2

Type.....Hard disk drive
Channel.....2
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR595624
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR594556

Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR758784
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR695895
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR593565
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR696877
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR769609
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR587676
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR589070
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....12
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR577283
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Enclosure
Channel.....2
SCSI ID.....15
Enclosure status.....Okay
Fan 1 status.....Okay
Fan 2 status.....Okay
Power supply 1 status.....Okay
Power supply 2 status.....Okay
Temperature status.....Normal
Vendor.....IBM
Product or model number.....EXP200
Serial number.....94614622
Firmware level.....1020
Enclosure ID.....0
FRU type.....MIDPLANE
FRU vendor.....IBM
FRU date of manufacture.....12/1999
FRU part number.....37L0061
FRU serial number.....23A1685
FRU type.....ESM_CARD
FRU vendor.....IBM
FRU date of manufacture.....11/1999
FRU part number.....37L0063
FRU serial number.....1T94614622

SCSI channel 3

Type.....Hard disk drive

Channel.....3
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR762480
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR768077
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR589133
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR594818
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR583612
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....8
Size in MB.....17357

State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR588710
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR594204
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR595595
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR599412
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....12
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR767125
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Enclosure
Channel.....3
SCSI ID.....15
Enclosure status.....Okay
Fan 1 status.....Okay
Fan 2 status.....Okay
Power supply 1 status.....Okay

Power supply 2 status.....Okay
Temperature status.....Normal
Vendor.....IBM
Product or model number.....EXP200
Serial number.....94212079
Firmware level.....1020
Enclosure ID.....0
FRU type.....MIDPLANE
FRU vendor.....IBM
FRU date of manufacture.....12/1999
FRU part number.....37L0061
FRU serial number.....23A1449
FRU type.....ESM_CARD
FRU vendor.....IBM
FRU date of manufacture.....11/1999
FRU part number.....37L0063
FRU serial number.....1T94212079

SCSI channel 4

End of the configuration information for controller 5

Information for controller 6

Controller type.....ServeRAID-4H
BIOS version.....4.70.06
Firmware version.....joe.05
Physical slot.....3
Battery-backup cache.....Installed
Read-ahead cache mode.....Adaptive
Stripe-unit size.....64K
Rebuild rate.....High
Hot-swap rebuild.....Enabled
Data scrubbing.....Disabled
Auto-synchronization.....Enabled
Clustering.....Disabled
Unattended mode.....Disabled
Number of arrays.....2
Number of logical drives.....2
Number of hot-spare drives.....0
Number of ready drives.....0

Array A

Array identifier.....A
Array size in MB.....260355
Free space in MB.....15
Number of logical drives.....1
Number of physical drives.....15

Logical drives in array A

Logical drive.....1
Array letter.....A
State.....Okay
RAID level.....1E
Data space in MB.....130170
Parity space in MB.....130170
Date created.....12/15/00
Write-cache mode.....Write back
Merge group number.....207
Merge group state.....Non-shared

Physical drives in array A

Type.....Hard disk drive
Channel.....1
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR600262
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR594703
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR506279
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR599069
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR762137
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....0

Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR744249
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR589274
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR768592
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR769499
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR699015
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A

Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR759217
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR594175
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR571155
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR599685
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR580028
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Array B

Array identifier.....B
Array size in MB.....260355
Free space in MB.....15
Number of logical drives.....1
Number of physical drives.....15

Logical drives in array B

Logical drive.....2
Array letter.....B
State.....Okay
RAID level.....1E
Data space in MB.....130170
Parity space in MB.....130170
Date created.....12/15/00
Write-cache mode.....Write back
Merge group number.....207
Merge group state.....Non-shared

Physical drives in array B

Type.....Hard disk drive
Channel.....1
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR767323
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR766899
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR768940
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR600159
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive

Channel.....1
SCSI ID.....12
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR744866
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR594862
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR587620
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR594627
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR768975
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....12
Size in MB.....17357

State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR584280
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR600143
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR725164
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR595709
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR768623
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....12
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG

Product or model number.....ST318203
Serial number.....LR769438
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

SCSI channel 1

Type.....Hard disk drive
Channel.....1
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR600262
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR594703
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR506279
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR599069
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A

Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR762137
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR767323
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR766899
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR768940
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR600159
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....12
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR744866

Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Enclosure
Channel.....1
SCSI ID.....15
Enclosure status.....Okay
Fan 1 status.....Okay
Fan 2 status.....Okay
Power supply 1 status.....Okay
Power supply 2 status.....Okay
Temperature status.....Normal
Vendor.....IBM
Product or model number.....EXP200
Serial number.....94211249
Firmware level.....1020
Enclosure ID.....0
FRU type.....MIDPLANE
FRU vendor.....IBM
FRU date of manufacture.....12/1999
FRU part number.....37L0061
FRU serial number.....23A1450
FRU type.....ESM_CARD
FRU vendor.....IBM
FRU date of manufacture.....11/1999
FRU part number.....37L0063
FRU serial number.....1T94211249

SCSI channel 2

Type.....Hard disk drive
Channel.....2
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR744249
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR589274
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR768592
Firmware level.....B227
PFA error.....No

Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR769499
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR699015
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR594862
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR587620
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR594627
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive

Channel.....2
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR768975
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....12
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR584280
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Enclosure
Channel.....2
SCSI ID.....15
Enclosure status.....Okay
Fan 1 status.....Okay
Fan 2 status.....Okay
Power supply 1 status.....Okay
Power supply 2 status.....Okay
Temperature status.....Normal
Vendor.....IBM
Product or model number.....EXP200
Serial number.....94212350
Firmware level.....1020
Enclosure ID.....0
FRU type.....MIDPLANE
FRU vendor.....IBM
FRU date of manufacture.....12/1999
FRU part number.....37L0061
FRU serial number.....23A1544
FRU type.....ESM_CARD
FRU vendor.....IBM
FRU date of manufacture.....11/1999
FRU part number.....37L0063
FRU serial number.....1T94212350

SCSI channel 3

Type.....Hard disk drive
Channel.....3
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR759217
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....1

Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR594175
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR571155
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR599685
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR580028
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR600143
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B

Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR725164
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR595709
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR768623
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....12
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR769438
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Enclosure
Channel.....3
SCSI ID.....15
Enclosure status.....Okay
Fan 1 status.....Okay
Fan 2 status.....Okay
Power supply 1 status.....Okay
Power supply 2 status.....Okay
Temperature status.....Normal
Vendor.....IBM
Product or model number.....EXP200
Serial number.....94614442
Firmware level.....1020
Enclosure ID.....0
FRU type.....MIDPLANE
FRU vendor.....IBM
FRU date of manufacture.....12/1999
FRU part number.....37L0061
FRU serial number.....23A1584
FRU type.....ESM_CARD
FRU vendor.....IBM
FRU date of manufacture.....11/1999

FRU part number.....37L0063
FRU serial number.....1T94614442

SCSI channel 4

End of the configuration information for controller 6

Information for controller 7

Controller type.....ServeRAID-4H
BIOS version.....4.70.06
Firmware version.....joe.05
Physical slot.....4
Battery-backup cache.....Installed
Read-ahead cache mode.....Adaptive
Stripe-unit size.....64K
Rebuild rate.....High
Hot-swap rebuild.....Enabled
Data scrubbing.....Enabled
Auto-synchronization.....Enabled
Clustering.....Disabled
Unattended mode.....Disabled
Number of arrays.....2
Number of logical drives.....2
Number of hot-spare drives.....0
Number of ready drives.....0

Array A

Array identifier.....A
Array size in MB.....260355
Free space in MB.....15
Number of logical drives.....1
Number of physical drives.....15

Logical drives in array A

Logical drive.....1
Array letter.....A
State.....Okay
RAID level.....1E
Data space in MB.....130170
Parity space in MB.....130170
Date created.....12/15/00
Write-cache mode.....Write back
Merge group number.....207
Merge group state.....Non-shared

Physical drives in array A

Type.....Hard disk drive
Channel.....1
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR580600
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1

SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR594822
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR583952
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR594908
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR594648
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR766426
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....1
Size in MB.....17357
State.....Online

Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR766901
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR758476
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR759800
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR712141
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR740227
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203

Serial number.....LR737959
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR758352
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR757814
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR580760
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Array B

Array identifier.....B
Array size in MB.....260355
Free space in MB.....15
Number of logical drives.....1
Number of physical drives.....15

Logical drives in array B

Logical drive.....2
Array letter.....B
State.....Okay
RAID level.....1E
Data space in MB.....130170
Parity space in MB.....130170
Date created.....12/15/00
Write-cache mode.....Write back
Merge group number.....207
Merge group state.....Non-shared

Physical drives in array B

Type.....Hard disk drive
Channel.....1
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR595123
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR751181
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR589222
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR589121
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....12
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR595013
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....8

Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR733219
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR595279
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR768887
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR697502
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....12
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR768743
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B

Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR766557
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR592895
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR758670
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR580392
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....12
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR762087
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

SCSI channel 1

Type.....Hard disk drive
Channel.....1
SCSI ID.....0
Size in MB.....17357
State.....Online

Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR580600
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR594822
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR583952
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR594908
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR594648
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203

Serial number.....LR595123
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR751181
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR589222
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR589121
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....12
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR595013
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Enclosure
Channel.....1
SCSI ID.....15
Enclosure status.....Okay
Fan 1 status.....Okay
Fan 2 status.....Okay
Power supply 1 status.....Okay
Power supply 2 status.....Okay
Temperature status.....Normal
Vendor.....IBM
Product or model number.....EXP200

Serial number.....94212461
Firmware level.....1020
Enclosure ID.....0
FRU type.....MIDPLANE
FRU vendor.....IBM
FRU date of manufacture.....12/1999
FRU part number.....37L0061
FRU serial number.....23A1460
FRU type.....ESM_CARD
FRU vendor.....IBM
FRU date of manufacture.....11/1999
FRU part number.....37L0063
FRU serial number.....1T94212461

SCSI channel 2

Type.....Hard disk drive
Channel.....2
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR766426
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR766901
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR758476
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR759800
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR712141
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR733219
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR595279
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR768887
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR697502
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....12

Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR768743
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Enclosure
Channel.....2
SCSI ID.....15
Enclosure status.....Okay
Fan 1 status.....Okay
Fan 2 status.....Okay
Power supply 1 status.....Okay
Power supply 2 status.....Okay
Temperature status.....Normal
Vendor.....IBM
Product or model number.....EXP200
Serial number.....94614667
Firmware level.....1020
Enclosure ID.....0
FRU type.....MIDPLANE
FRU vendor.....IBM
FRU date of manufacture.....12/1999
FRU part number.....37L0061
FRU serial number.....23A1629
FRU type.....ESM_CARD
FRU vendor.....IBM
FRU date of manufacture.....11/1999
FRU part number.....37L0063
FRU serial number.....1T94614667

SCSI channel 3

Type.....Hard disk drive
Channel.....3
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR740227
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR737959
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....2
Size in MB.....17357
State.....Online

Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR758352
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR757814
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR580760
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR766557
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR592895
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203

Serial number.....LR758670
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR580392
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....12
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR762087
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Enclosure
Channel.....3
SCSI ID.....15
Enclosure status.....Okay
Fan 1 status.....Okay
Fan 2 status.....Okay
Power supply 1 status.....Okay
Power supply 2 status.....Okay
Temperature status.....Normal
Vendor.....IBM
Product or model number.....EXP200
Serial number.....94614602
Firmware level.....1020
Enclosure ID.....0
FRU type.....MIDPLANE
FRU vendor.....IBM
FRU date of manufacture.....12/1999
FRU part number.....37L0061
FRU serial number.....23A1656
FRU type.....ESM_CARD
FRU vendor.....IBM
FRU date of manufacture.....11/1999
FRU part number.....37L0063
FRU serial number.....1T94614602

SCSI channel 4

End of the configuration information for controller 7

Information for controller 8

Controller type.....ServeRAID-4H
BIOS version.....4.70.06
Firmware version.....joe.05
Physical slot.....6

Battery-backup cache.....Installed
Read-ahead cache mode.....Adaptive
Stripe-unit size.....64K
Rebuild rate.....High
Hot-swap rebuild.....Enabled
Data scrubbing.....Enabled
Auto-synchronization.....Enabled
Clustering.....Disabled
Unattended mode.....Disabled
Number of arrays.....2
Number of logical drives.....2
Number of hot-spare drives.....0
Number of ready drives.....0

Array A

Array identifier.....A
Array size in MB.....260355
Free space in MB.....15
Number of logical drives.....1
Number of physical drives.....15

Logical drives in array A

Logical drive.....1
Array letter.....A
State.....Okay
RAID level.....1E
Data space in MB.....130170
Parity space in MB.....130170
Date created.....12/15/00
Write-cache mode.....Write back
Merge group number.....207
Merge group state.....Non-shared

Physical drives in array A

Type.....Hard disk drive
Channel.....1
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR587168
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR579779
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....2
Size in MB.....17357

State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR595622
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR570897
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR746936
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR571167
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR580340
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG

Product or model number.....ST318203
Serial number.....LR599315
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR580491
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR594712
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR595608
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR716832
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR580765
Firmware level.....B227

PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR578474
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR595125
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Array B

Array identifier.....B
Array size in MB.....260355
Free space in MB.....15
Number of logical drives.....1
Number of physical drives.....15

Logical drives in array B

Logical drive.....2
Array letter.....B
State.....Okay
RAID level.....1E
Data space in MB.....130170
Parity space in MB.....130170
Date created.....12/15/00
Write-cache mode.....Write back
Merge group number.....207
Merge group state.....Non-shared

Physical drives in array B

Type.....Hard disk drive
Channel.....1
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR593356
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1

SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR766289
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR600773
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR762502
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....12
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR580538
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR569956
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....9
Size in MB.....17357
State.....Online

Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR750481
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR593187
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR555947
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....12
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR732987
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR768418
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203

Serial number.....LR580540
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR594931
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR553744
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....12
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR580342
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

SCSI channel 1

Type.....Hard disk drive
Channel.....1
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR587168
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG

Product or model number.....ST318203
Serial number.....LR579779
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR595622
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR570897
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR746936
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR593356
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR766289
Firmware level.....B227

PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR600773
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR762502
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....12
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR580538
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Enclosure
Channel.....1
SCSI ID.....15
Enclosure status.....Okay
Fan 1 status.....Okay
Fan 2 status.....Okay
Power supply 1 status.....Okay
Power supply 2 status.....Okay
Temperature status.....Normal
Vendor.....IBM
Product or model number.....EXP200
Serial number.....94008076
Firmware level.....1020
Enclosure ID.....0
FRU type.....MIDPLANE
FRU vendor.....IBM
FRU date of manufacture.....12/1999
FRU part number.....37L0061
FRU serial number.....23A1605
FRU type.....ESM_CARD
FRU vendor.....IBM
FRU date of manufacture.....10/1999
FRU part number.....37L0063
FRU serial number.....1T94008076

SCSI channel 2

Type.....Hard disk drive
Channel.....2
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR571167
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR580340
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR599315
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR580491
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR594712
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2

SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR569956
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR750481
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR593187
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR555947
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....2
SCSI ID.....12
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR732987
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Enclosure
Channel.....2
SCSI ID.....15
Enclosure status.....Okay
Fan 1 status.....Okay

Fan 2 status.....Okay
Power supply 1 status.....Okay
Power supply 2 status.....Okay
Temperature status.....Normal
Vendor.....IBM
Product or model number.....EXP200
Serial number.....94614608
Firmware level.....1020
Enclosure ID.....0
FRU type.....MIDPLANE
FRU vendor.....IBM
FRU date of manufacture.....12/1999
FRU part number.....37L0061
FRU serial number.....23A1652
FRU type.....ESM_CARD
FRU vendor.....IBM
FRU date of manufacture.....11/1999
FRU part number.....37L0063
FRU serial number.....1T94614608

SCSI channel 3

Type.....Hard disk drive
Channel.....3
SCSI ID.....0
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR595608
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....1
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR716832
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....2
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR580765
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....3
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG

Product or model number.....ST318203
Serial number.....LR578474
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....4
Size in MB.....17357
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR595125
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....8
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR768418
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....9
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR580540
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....10
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR594931
Firmware level.....B227
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....3
SCSI ID.....11
Size in MB.....17357
State.....Online
Array letter.....B
Vendor.....IBM-PSG
Product or model number.....ST318203
Serial number.....LR553744
Firmware level.....B227

PFA error.....No
 Write-cache mode.....Write through

Type.....Hard disk drive
 Channel.....3
 SCSI ID.....12
 Size in MB.....17357
 State.....Online
 Array letter.....B
 Vendor.....IBM-PSG
 Product or model number.....ST318203
 Serial number.....LR580342
 Firmware level.....B227
 PFA error.....No
 Write-cache mode.....Write through

Type.....Enclosure
 Channel.....3
 SCSI ID.....15
 Enclosure status.....Okay
 Fan 1 status.....Okay
 Fan 2 status.....Okay
 Power supply 1 status.....Okay
 Power supply 2 status.....Okay
 Temperature status.....Normal
 Vendor.....IBM
 Product or model number.....EXP200
 Serial number.....94212317
 Firmware level.....1020
 Enclosure ID.....0
 FRU type.....MIDPLANE
 FRU vendor.....IBM
 FRU date of manufacture.....12/1999
 FRU part number.....37L0061
 FRU serial number.....23A1500
 FRU type.....ESM_CARD
 FRU vendor.....IBM
 FRU date of manufacture.....11/1999
 FRU part number.....37L0063
 FRU serial number.....1T94212317

SCSI channel 4

End of the configuration information for controller 8

DTC Server Configuration

Software Configuration

```
[boot loader]
timeout=30
default=multi(0)disk(0)rdisk(0)partition(1)\WINNT
[operating systems]
multi(0)disk(0)rdisk(0)partition(1)\WINNT="Microsoft Windows 2000 Advanced Server" /fastdetect
```

The DTC log size was set to 400MB for each DTC server. A local DTC coordinator was specified for each DTC server.

No applications were installed on the DTC servers

Hardware Configuration

System Information report written at: 03/06/2001 02:47:19 PM
 [System Information]

[Following are sub-categories of this main category]

[System Summary]

Item	Value
OS Name	Microsoft Windows 2000 Advanced Server
Version	5.0.2195 Build 2195
OS Manufacturer	Microsoft Corporation
System Name	DTCSESV1
System Manufacturer	IBM
System Model	IBM Server -[8681]-
System Type	X86-based PC
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
BIOS Version	IBM BIOS Ver 6.0
Windows Directory	C:\WINNT
System Directory	C:\WINNT\System32
Boot Device	\Device\Harddisk0\Partition1
Locale	United States
User Name	DTCSESV1\tpcc
Time Zone	Eastern Standard Time
Total Physical Memory	523,816 KB
Available Physical Memory	365,840 KB
Total Virtual Memory	3,110,916 KB
Available Virtual Memory	2,815,312 KB
Page File Space	2,587,100 KB
Page File	C:\pagefile.sys

[Hardware Resources]

[Following are sub-categories of this main category]

[Conflicts/Sharing]

Resource	Device
IRQ 51	Adaptec AIC-7896/AIC-7897 PCI Ultra2 SCSI Controller
IRQ 51	Adaptec AIC-7896/AIC-7897 PCI Ultra2 SCSI Controller

[DMA]

Channel	Device	Status
2	Standard floppy disk controller	OK
4	Direct memory access controller	OK

[Forced Hardware]

Device	PNP Device ID
No Forced Hardware	

[I/O]

Address Range	Device	Status
0x0000-0x0CF7	PCI bus	OK
0x0000-0x0CF7	Direct memory access controller	OK
0x0D00-0x5FFF	PCI bus	OK

0x2000-0x20FF	Other PCI Bridge Device	OK
0x2180-0x218F	Other PCI Bridge Device	OK
0x03B0-0x03BB	S3 Inc. Trio3D	OK
0x03C0-0x03DF	S3 Inc. Trio3D	OK
0x2200-0x22FF	IBM ServeRAID 4H Controller	OK
0x2300-0x23FF	Adaptec AIC-7896/AIC-7897 PCI Ultra2 SCSI Controller	OK
0x2400-0x24FF	Adaptec AIC-7896/AIC-7897 PCI Ultra2 SCSI Controller	OK
0x0A79-0x0A79	ISAPNP Read Data Port	OK
0x0279-0x0279	ISAPNP Read Data Port	OK
0x02F4-0x02F7	ISAPNP Read Data Port	OK
0x002E-0x002F	Motherboard resources	OK
0x0438-0x0439	Motherboard resources	OK
0x0430-0x0437	Motherboard resources	OK
0x0060-0x0060	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard	OK
0x0064-0x0064	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard	OK
0x03F0-0x03F5	Standard floppy disk controller	OK
0x03F7-0x03F7	Standard floppy disk controller	OK
0x0378-0x037F	Printer Port (LPT1)	OK
0x03F8-0x03FF	Communications Port (COM1)	OK
0x02F8-0x02FF	Communications Port (COM2)	OK
0x0020-0x0021	Advanced programmable interrupt controller	OK
0x00A0-0x00A1	Advanced programmable interrupt controller	OK
0x0080-0x008F	Direct memory access controller	OK
0x00C0-0x00DF	Direct memory access controller	OK
0x0040-0x0043	System timer	OK
0x0070-0x0073	System CMOS/real time clock	OK
0x0061-0x0061	System speaker	OK
0x00F0-0x00FF	Numeric data processor	OK
0x04D0-0x04D1	Motherboard resources	OK
0x00E8-0x00EF	Motherboard resources	OK
0x5FF0-0x5FFF	Intel(r) 82371AB/EB PCI Bus Master IDE Controller	OK
0x01F0-0x01F7	Primary IDE Channel	OK
0x03F6-0x03F6	Primary IDE Channel	OK
0x5FC0-0x5FDF	Intel 82371AB/EB PCI to USB Universal Host Controller	OK
0x6000-0xAFFF	PCI bus	OK
0xB000-0xCFFF	PCI bus	OK
0xD000-0xFFFF	PCI bus	OK

[IRQs]

IRQ Number	Device
9	Microsoft ACPI-Compliant System
5	Other PCI Bridge Device
25	IBM ServeRAID 4H Controller
51	Adaptec AIC-7896/AIC-7897 PCI Ultra2 SCSI Controller
51	Adaptec AIC-7896/AIC-7897 PCI Ultra2 SCSI Controller
1	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard
12	PS/2 Compatible Mouse
6	Standard floppy disk controller
4	Communications Port (COM1)
3	Communications Port (COM2)
8	System CMOS/real time clock
13	Numeric data processor
14	Primary IDE Channel
49	Intel 82371AB/EB PCI to USB Universal Host Controller
16	cLAN Host Adapter

[Memory]

Range	Device	Status
-------	--------	--------

0xA0000-0xBFFFF	PCI bus	OK
0xA0000-0xBFFFF	S3 Inc. Trio3D	OK
0xC8000-0xDFFFF	PCI bus	OK
0xF0000000-0xFBFFFFFF	PCI bus	OK
0xFEC00000-0xFFFFFFFF	PCI bus	OK
0xFEC00000-0xFFFFFFFF	Advanced programmable interrupt controller	OK
0xFBFFFC00-0xFBFFFCFF	Other PCI Bridge Device	OK
0xFBFE0000-0xFBFEFFFF	Other PCI Bridge Device	OK
0xF4000000-0xF7FFFFFF	S3 Inc. Trio3D	OK
0xFBE00000-0xFBEFFFFFF	IBM ServeRAID 4H Controller	OK
0xFBFFE000-0xFBFEFFFF	Adaptec AIC-7896/AIC-7897 PCI Ultra2 SCSI Controller	OK
0xFBFFD000-0xFBFFDFFF	Adaptec AIC-7896/AIC-7897 PCI Ultra2 SCSI Controller	OK
0xFC000000-0xFD7FFFFF	PCI bus	OK
0xFC000000-0xFD7FFFFF	cLAN Host Adapter	OK
0xFD7E0000-0xFD7FFFFF	cLAN Host Adapter	OK
0xFD400000-0xFD5FFFFF	cLAN Host Adapter	OK
0xFD7D0000-0xFD7DFFFF	cLAN Host Adapter	OK
0xFD800000-0xFE1FFFFF	PCI bus	OK
0xFE200000-0xFEBFFFFFF	PCI bus	OK

[Components]

[Following are sub-categories of this main category]

[Multimedia]

[Following are sub-categories of this main category]

[Audio Codecs]

Codec Version	Manufacturer Size	Description Creation Date	Status	File
c:\winnt\system32\msg711.acm	Microsoft Corporation			
OK	C:\WINNT\System32\MSG711.ACM	5.00.2134.1		
10.27 KB (10,512 bytes)		12/7/1999 6:00:00 AM		
c:\winnt\system32\tsssoft32.acm	DSP GROUP, INC.		OK	
C:\WINNT\System32\TSSOFT32.ACM	1.01	9.27 KB (9,488 bytes)		
12/7/1999 6:00:00 AM				
c:\winnt\system32\msadp32.acm	Microsoft Corporation			
OK	C:\WINNT\System32\MSADP32.ACM	5.00.2134.1		
14.77 KB (15,120 bytes)		12/7/1999 6:00:00 AM		
c:\winnt\system32\msg723.acm	Microsoft Corporation			
OK	C:\WINNT\System32\MSG723.ACM	4.4.3385		
106.77 KB (109,328 bytes)		2/17/2001 12:49:02 PM		
c:\winnt\system32\iac25_32.ax	Intel Corporation		Indeo® audio software	OK
C:\WINNT\System32\IAC25_32.AX				
2.05.53	195.00 KB (199,680 bytes)			
12/7/1999 6:00:00 AM				
c:\winnt\system32\imaadp32.acm	Microsoft Corporation			
OK	C:\WINNT\System32\IMAADP32.ACM			
5.00.2134.1	16.27 KB (16,656 bytes)			
12/7/1999 6:00:00 AM				
c:\winnt\system32\lhacm.acm	Microsoft Corporation			
OK	C:\WINNT\System32\LHACM.ACM	4.4.3385		
33.27 KB (34,064 bytes)		2/17/2001 12:49:03 PM		
c:\winnt\system32\msgsm32.acm	Microsoft Corporation			
OK	C:\WINNT\System32\MSGSM32.ACM	5.00.2134.1		
22.27 KB (22,800 bytes)		12/7/1999 6:00:00 AM		

[Video Codecs]

Codec Version	Manufacturer Size	Description Creation Date	Status	File
c:\winnt\system32\ir50_32.dll	Intel Corporation		Indeo® video	OK
5.10	OK	C:\WINNT\System32\IR50_32.DLL		

R.5.10.15.2.55 737.50 KB (755,200 bytes) 12/7/1999 6:00:00 AM
 c:\winnt\system32\msh261.drv Microsoft Corporation
 OK C:\WINNT\System32\MSH261.DRV 4.4.3385
 163.77 KB (167,696 bytes) 2/17/2001 12:49:03 PM
 c:\winnt\system32\msh263.drv Microsoft Corporation
 OK C:\WINNT\System32\MSH263.DRV 4.4.3385
 252.27 KB (258,320 bytes) 2/17/2001 12:48:37 PM
 c:\winnt\system32\msvidc32.dll Microsoft Corporation
 OK C:\WINNT\System32\MSVIDC32.DLL 5.00.2134.1
 27.27 KB (27,920 bytes) 12/7/1999 6:00:00 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 6:00:00 AM
 c:\winnt\system32\ir32_32.dll Intel(R) Corporation OK
 C:\WINNT\System32\IR32_32.DLL Not Available
 194.50 KB (199,168 bytes) 12/7/1999 6:00:00 AM
 c:\winnt\system32\iccvld.dll Radius Inc. OK
 C:\WINNT\System32\ICCVLD.DLL 1.10.0.6 108.00 KB
 (110,592 bytes) 12/7/1999 6:00:00 AM

[CD-ROM]

Item	Value
Drive	D:
Description	CD-ROM Drive
Media Loaded	False
Media Type	CD-ROM
Name	LG CD-ROM CRD-8400B
Manufacturer	(Standard CD-ROM drives)
Status	OK
Transfer Rate	Not Available
SCSI Target ID	0
PNP Device ID	IDE\CDROMLG_CD-ROM_CRD-8400B_____1.08____\4249204D393932303830202020202020202020

[Sound Device]

Item	Value
No sound devices	

[Display]

Item	Value
Name	S3 Inc. Trio3D
PNP Device ID	PCI\VEN_5333&DEV_8904&SUBSYS_00DB1014&REV_01\3&267A616A&0&20
Adapter Type	S3 Trio3D, S3 compatible
Adapter Description	S3 Inc. Trio3D
Adapter RAM	4.00 MB (4,194,304 bytes)
Installed Drivers	s3mt3d.sys
Driver Version	5.01.526.0007
INF File	s3trio3d.inf (S3Inc section)
Color Planes	1
Color Table Entries	4294967296
Resolution	800 x 600 x 60 hertz
Bits/Pixel	32

[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&43B47AD&0
NumberOfFunctionKeys	12

[Pointing Device]

Item	Value
Hardware Type	PS/2 Compatible Mouse
Number of Buttons	3
Status	OK
PNP Device ID	ACPI\PNP0F13\4&43B47AD&0
Power Management Supported	False
Double Click Threshold	6
Handedness	Right Handed Operation

[Modem]

Item	Value
No modems	

[Network]

[Following are sub-categories of this main category]

[Adapter]

Item	Value
Name	[00000000] RAS Async Adapter
Adapter Type	Not Available
Product Name	RAS Async Adapter
Installed	True
PNP Device ID	Not Available
Last Reset	2/28/2001 12:38:24 PM
Index	0
Service Name	AsyncMac
IP Address	Not Available
IP Subnet	Not Available
Default IP Gateway	Not Available
DHCP Enabled	False
DHCP Server	Not Available
DHCP Lease Expires	Not Available
DHCP Lease Obtained	Not Available
MAC Address	Not Available
Service Name	Not Available
Name	[00000001] WAN Miniport (L2TP)
Adapter Type	Not Available
Product Name	WAN Miniport (L2TP)
Installed	True
PNP Device ID	ROOT\MS_L2TPMINIPORT\0000
Last Reset	2/28/2001 12:38:24 PM
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 Rsl2tp
 Driver c:\winnt\system32\drivers\rsl2tp.sys (50800, 5.00.2179.1)

Name [00000002] WAN Miniport (PPTP)
 Adapter Type Wide Area Network (WAN)
 Product Name WAN Miniport (PPTP)
 Installed True
 PNP Device ID ROOT\MS_PPTPMINIPOINT\0000
 Last Reset 2/28/2001 12:38:24 PM
 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 (47856, 5.00.2160.1)

Name [00000003] Direct Parallel
 Adapter Type Not Available
 Product Name Direct Parallel
 Installed True
 PNP Device ID ROOT\MS_PTIMINIPOINT\0000
 Last Reset 2/28/2001 12:38:24 PM
 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 2/28/2001 12:38:24 PM
 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 (90768, 5.00.2184.1)

Name [00000005] cLAN Host Adapter
 Adapter Type Ethernet 802.3
 Product Name cLAN Host Adapter

Installed True
 PNP Device ID PCI\VEN_135B&DEV_0001&SUBSYS_00000000&REV_00\3&172E68DD&0&08
 Last Reset 2/28/2001 12:38:24 PM
 Index 5
 Service Name GNINDIS
 IP Address 192.168.100.101
 IP Subnet 255.255.255.0
 Default IP Gateway Not Available
 DHCP Enabled False
 DHCP Server Not Available
 DHCP Lease Expires Not Available
 DHCP Lease Obtained Not Available
 MAC Address 00:90:FA:00:11:44
 Service Name GNINDIS
 IRQ Number 16
 Driver c:\winnt\system32\drivers\gnindis.sys (22598, 4.1.1)

[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_{21167DBF-AF45-4F45-AD48-D24E7D5A11C9}] SEQUENCEPACKET 0
 ConnectionlessService False
 GuaranteesDelivery True
 GuaranteesSequencing True
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting False
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT_Tcpip_{21167DBF-AF45-4F45-AD48-D24E7D5A11C9}] DATAGRAM 0
 ConnectionlessService True
 GuaranteesDelivery False
 GuaranteesSequencing False
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting True
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT_Tcpip_{40DEE24F-AB8B-48B9-8B8D-F0E347D43261}] SEQUENCEPACKET 1
 ConnectionlessService False
 GuaranteesDelivery True
 GuaranteesSequencing True
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting False
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT_Tcpip_{40DEE24F-AB8B-48B9-8B8D-F0E347D43261}] DATAGRAM 1
 ConnectionlessService True
 GuaranteesDelivery False
 GuaranteesSequencing False
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting True
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT_Tcpip_{791936A8-C857-4151-A02A-EB495EC6F2FE}] SEQUENCEPACKET 2
 ConnectionlessService False
 GuaranteesDelivery True
 GuaranteesSequencing True
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting False
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT_Tcpip_{791936A8-C857-4151-A02A-EB495EC6F2FE}] DATAGRAM 2

ConnectionlessService True
 GuaranteesDelivery False
 GuaranteesSequencing False
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting True
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

[WinSock]

Item Value
 File c:\winnt\system32\winsock.dll
 Version 3.10
 Size 2.80 KB (2,864 bytes)

 File c:\winnt\system32\wsock32.dll
 Version 5.00.2152.1
 Size 21.27 KB (21,776 bytes)

[Ports]

[Following are sub-categories of this main category]

[Serial]

Item Value
 Name COM1
 Status OK
 PNP Device ID ACPI\PNP0501\1
 Maximum Input Buffer Size 0
 Maximum Output Buffer Size False
 Settable Baud Rate True
 Settable Data Bits True
 Settable Flow Control True
 Settable Parity True
 Settable Parity Check True
 Settable Stop Bits True
 Settable RLSD True
 Supports RLSD True
 Supports 16 Bit Mode False
 Supports Special Characters False
 Baud Rate 9600
 Bits/Byte 8
 Stop Bits 1
 Parity None
 Busy 0
 Abort Read/Write on Error 0
 Binary Mode Enabled-1
 Continue XMit on XOff 0
 CTS Outflow Control 0
 Discard NULL Bytes 0
 DSR Outflow Control 0
 DSR Sensitivity 0
 DTR Flow Control Type Enable
 EOF Character 0
 Error Replace Character 0
 Error Replacement Enabled 0

Event Character 0
 Parity Check Enabled 0
 RTS Flow Control Type Enable
 XOff Character 19
 XOffXMit Threshold 512
 XOn Character 17
 XOnXMit Threshold 2048
 XOnXOff InFlow Control 0
 XOnXOff OutFlow Control 0
 IRQ Number 4
 I/O Port 0x03F8-0x03FF
 Driver c:\winnt\system32\drivers\serial.sys (62448, 5.00.2134.1)

Name COM2
 Status OK
 PNP Device ID ACPI\PNP0501\2
 Maximum Input Buffer Size 0
 Maximum Output Buffer Size False
 Settable Baud Rate True
 Settable Data Bits True
 Settable Flow Control True
 Settable Parity True
 Settable Parity Check True
 Settable Stop Bits True
 Settable RLSD True
 Supports RLSD True
 Supports 16 Bit Mode False
 Supports Special Characters False
 Baud Rate 9600
 Bits/Byte 8
 Stop Bits 1
 Parity None
 Busy 0
 Abort Read/Write on Error 0
 Binary Mode Enabled-1
 Continue XMit on XOff 0
 CTS Outflow Control 0
 Discard NULL Bytes 0
 DSR Outflow Control 0
 DSR Sensitivity 0
 DTR Flow Control Type Enable
 EOF Character 0
 Error Replace Character 0
 Error Replacement Enabled 0
 Event Character 0
 Parity Check Enabled 0
 RTS Flow Control Type Enable
 XOff Character 19
 XOffXMit Threshold 512
 XOn Character 17
 XOnXMit Threshold 2048
 XOnXOff InFlow Control 0
 XOnXOff OutFlow Control 0
 IRQ Number 3
 I/O Port 0x02F8-0x02FF
 Driver c:\winnt\system32\drivers\serial.sys (62448, 5.00.2134.1)

[Parallel]

Item Value
 Name LPT1
 PNP Device ID ACPI\PNP0400\1

[Storage]

[Following are sub-categories of this main category]

[Drives]

```

Item      Value
Drive A:
Description      3 1/2 Inch Floppy Drive

Drive C:
Description      Local Fixed Disk
Compressed       False
File System      NTFS
Size             8.46 GB (9,088,901,120 bytes)
Free Space       5.77 GB (6,195,757,056 bytes)
Volume Name
Volume Serial Number      38BCB48F
Partition Disk #0, Partition #0
Partition Size           8.46 GB (9,088,902,144 bytes)
Starting Offset          32256 bytes
Drive Description       Disk drive
Drive Manufacturer     (Standard disk drives)
Drive Model             IBM-PSG ST39103LC  !# SCSI Disk Device
Drive BytesPerSector   512
Drive MediaLoaded      True
Drive MediaType        Fixed hard disk media
Drive Partitions       1
Drive SCSIbus         0
Drive SCSILogicalUnit  0
Drive SCSIPort        2
Drive SCSTargetId     0
Drive SectorsPerTrack 63
Drive Size             9097159680 bytes
Drive TotalCylinders  1106
Drive TotalSectors    17767890
Drive TotalTracks     282030
Drive TracksPerCylinder 255

Drive E:
Description      Local Fixed Disk
Compressed       False
File System      NTFS
Size             21.18 GB (22,742,863,872 bytes)
Free Space       20.73 GB (22,255,153,152 bytes)
Volume Name      New Volume
Volume Serial Number      087CBD58
Partition Disk #1, Partition #0
Partition Size           21.18 GB (22,742,866,944 bytes)
Starting Offset          32256 bytes
Drive Description       Disk drive
Drive Manufacturer     (Standard disk drives)
Drive Model             IBM ServeRAID Failover Virtual Disk Device
Drive BytesPerSector   512
Drive MediaLoaded      True
Drive MediaType        Fixed hard disk media
Drive Partitions       1
Drive SCSIbus         0
Drive SCSILogicalUnit  0
Drive SCSIPort        3
Drive SCSTargetId     0
Drive SectorsPerTrack 63
Drive Size             22742899200 bytes
Drive TotalCylinders  2765
Drive TotalSectors    44419725
Drive TotalTracks     705075
Drive TracksPerCylinder 255

```

[SCSI]

```

Item      Value
Name      IBM ServeRAID 4H Controller
Caption   IBM ServeRAID 4H Controller
Driver    ipsraidn
Status    OK
PNP Device ID
PCI\VEN_1014&DEV_002E&SUBSYS_022E1014&REV_10\3&267
A616A&0&50
Device ID
PCI\VEN_1014&DEV_002E&SUBSYS_022E1014&REV_10\3&267
A616A&0&50
Device Map      Not Available
Index           Not Available
Max Number Controlled      Not Available
IRQ Number       25
I/O Port        0x2200-0x22FF
Driver          c:\winnt\system32\drivers\ipsraidn.sys (102807, 4.70.08)

Name      Adaptec AIC-7896/AIC-7897 PCI Ultra2 SCSI Controller
Caption   Adaptec AIC-7896/AIC-7897 PCI Ultra2 SCSI Controller
Driver    aic78u2
Status    OK
PNP Device ID
PCI\VEN_9005&DEV_005F&SUBSYS_080F9005&REV_00\3&267
A616A&0&70
Device ID
PCI\VEN_9005&DEV_005F&SUBSYS_080F9005&REV_00\3&267
A616A&0&70
Device Map      Not Available
Index           Not Available
Max Number Controlled      Not Available
IRQ Number       51
I/O Port        0x2300-0x23FF
Driver          c:\winnt\system32\drivers\aic78u2.sys (65168, v3.00a)

Name      Adaptec AIC-7896/AIC-7897 PCI Ultra2 SCSI Controller
Caption   Adaptec AIC-7896/AIC-7897 PCI Ultra2 SCSI Controller
Driver    aic78u2
Status    OK
PNP Device ID
PCI\VEN_9005&DEV_005F&SUBSYS_080F9005&REV_00\3&267
A616A&0&71
Device ID
PCI\VEN_9005&DEV_005F&SUBSYS_080F9005&REV_00\3&267
A616A&0&71
Device Map      Not Available
Index           Not Available
Max Number Controlled      Not Available
IRQ Number       51
I/O Port        0x2400-0x24FF
Driver          c:\winnt\system32\drivers\aic78u2.sys (65168, v3.00a)

Name      IBM ServeRAID Failover Virtual Adapter
Caption   IBM ServeRAID Failover Virtual Adapter
Driver    twintail
Status    OK
PNP Device ID      ROOT\TWINTAIL\0000
Device ID          ROOT\TWINTAIL\0000
Device Map      Not Available
Index           Not Available
Max Number Controlled      Not Available
Driver          c:\winnt\system32\drivers\twintail.sys (26032, 4.70.06)

```

[Printing]

```

Name      Port Name Server Name
No printing information

```

[Problem Devices]

Device	PNP Device ID	Error Code
Other PCI Bridge Device		
PCI\VEN_1014&DEV_00DC&SUBSYS_00DC1014&REV_02\3&267A616A&0&10	28	
Not Available	ACPI\IBM37D0\4&43B47AD&0	28

[USB]

Device	PNP Device ID
Intel 82371AB/EB PCI to USB Universal Host Controller	
PCI\VEN_8086&DEV_7112&SUBSYS_00000000&REV_01\3&267A616A&0&7A	
USB Root Hub	USB\ROOT_HUB\4&B5B4E1B&0

March 6, 2001 2:46:45 PM EST

Configuration summary

```

Server name.....dtserv1
ServeRAID Manager Version.....4.00.34
Number of controllers.....1
Operating system.....Windows NT
Device driver version.....4.70.08

```

Information for controller 1

```

Controller type.....ServeRAID-4H
BIOS version.....4.70.06
Firmware version.....joe.05
Physical slot.....10
Battery-backup cache.....Installed
Read-ahead cache mode.....Adaptive
Stripe-unit size.....64K
Rebuild rate.....High
Hot-swap rebuild.....Enabled
Data scrubbing.....Enabled
Auto-synchronization.....Enabled
Clustering.....Disabled
Unattended mode.....Disabled
Number of arrays.....1
Number of logical drives.....1
Number of hot-spare drives.....0
Number of ready drives.....0

```

Array A

```

Array identifier.....A
Array size in MB.....43390
Free space in MB.....0
Number of logical drives.....1
Number of physical drives.....5

```

Logical drives in array A

```

Logical drive.....1
Array letter.....A
State.....Okay
RAID level.....1E
Data space in MB.....21695
Parity space in MB.....21695
Date created.....2/17/01
Write-cache mode.....Write back

```

```

Merge group number.....207
Merge group state.....Non-shared

```

Physical drives in array A

```

Type.....Hard disk drive
Channel.....1
SCSI ID.....0
Size in MB.....8678
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST39103L
Serial number.....LS075504
Firmware level.....B222
PFA error.....No
Write-cache mode.....Write through

```

```

Type.....Hard disk drive
Channel.....1
SCSI ID.....1
Size in MB.....8678
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST39103L
Serial number.....LS055308
Firmware level.....B222
PFA error.....No
Write-cache mode.....Write through

```

```

Type.....Hard disk drive
Channel.....1
SCSI ID.....2
Size in MB.....8678
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST39103L
Serial number.....LS056771
Firmware level.....B222
PFA error.....No
Write-cache mode.....Write through

```

```

Type.....Hard disk drive
Channel.....1
SCSI ID.....3
Size in MB.....8678
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST39103L
Serial number.....LS080038
Firmware level.....B222
PFA error.....No
Write-cache mode.....Write through

```

```

Type.....Hard disk drive
Channel.....1
SCSI ID.....4
Size in MB.....8678
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST39103L
Serial number.....LS066905
Firmware level.....B222

```

PFA error.....No
Write-cache mode.....Write through

SCSI channel 1

Type.....Hard disk drive
Channel.....1
SCSI ID.....0
Size in MB.....8678
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST39103L
Serial number.....LS075504
Firmware level.....B222
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....1
Size in MB.....8678
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST39103L
Serial number.....LS055308
Firmware level.....B222
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....2
Size in MB.....8678
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST39103L
Serial number.....LS056771
Firmware level.....B222
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....3
Size in MB.....8678
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST39103L
Serial number.....LS080038
Firmware level.....B222
PFA error.....No
Write-cache mode.....Write through

Type.....Hard disk drive
Channel.....1
SCSI ID.....4
Size in MB.....8678
State.....Online
Array letter.....A
Vendor.....IBM-PSG
Product or model number.....ST39103L
Serial number.....LS066905

Firmware level.....B222
PFA error.....No
Write-cache mode.....Write through

Type.....Enclosure
Channel.....1
SCSI ID.....15
Enclosure status.....Okay
Fan 1 status.....Okay
Fan 2 status.....Okay
Power supply 1 status.....Okay
Power supply 2 status.....Okay
Temperature status.....Normal
Vendor.....IBM
Product or model number.....EXP200
Serial number.....91961134
Firmware level.....1020
Enclosure ID.....0
FRU type.....MIDPLANE
FRU vendor.....IBM
FRU date of manufacture.....12/1999
FRU part number.....37L0061
FRU serial number.....23A1929
FRU type.....ESM_CARD
FRU vendor.....IBM
FRU date of manufacture.....12/1999
FRU part number.....37L0063
FRU serial number.....1T91961134

SCSI channel 2

SCSI channel 3

SCSI channel 4

End of the configuration information for controller 1

Client Configuration

Transaction Monitor: COM+ Settings on Clients

SYTPTUNE.TXT

Transaction Monitor: COM+ Settings on Clients

TPCC COM+ Component settings

TPCC.AllRemoteTxns.1
Activation
Enable Object pooling
Minimum pool size 7
Maximum pool size 7
Creation timeout (ms) 300000
Enable object construction
Object construction
Constructor string 'dummy string (do not remove)'
Enable Just In Time Activation
Disable Component supports events and statistics
Disable Must be activated in caller's context

TPCC.AllTxns.1

[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,01,02,00,01,01,00,00,00,00,00,\

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,01,02,00,01,01,01,00,00,\

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,\
00,01,01,00,00,00,00,05,12,00,00,00

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Enum]
"0"="Root\LEGACY_W3SVC\0000"
"Count"=dword:00000001
"NextInstance"=dword:00000001

System Information Report for Software Configuration

This software configuration inventory was obtained from 1 of 48 clients.
All 48 clients are identically configured.

System Information report written at: 03/05/2001 05:41:58 PM
[System Summary]

Item	Value
OS Name	Microsoft Windows 2000 Server
Version	5.0.2195 Build 2195
OS Manufacturer	Microsoft Corporation
System Name	RTCLIENT02
System Manufacturer	IBM
System Model	Netfinity 5100 -[86582RY]-
System Type	X86-based PC
Processor	x86 Family 6 Model 8 Stepping 3 GenuineIntel ~731 Mhz
Processor	x86 Family 6 Model 8 Stepping 3 GenuineIntel ~731 Mhz
BIOS Version	IBM BIOS Ver 3.0
Windows Directory	C:\WINNT
System Directory	C:\WINNT\System32
Boot Device	\Device\Harddisk0\Partition1
Locale	United States
User Name	RTCLIENT02\tpcc
Time Zone	Eastern Standard Time
Total Physical Memory	523,796 KB
Available Physical Memory	422,772 KB
Total Virtual Memory	1,801,796 KB
Available Virtual Memory	1,619,864 KB
Page File Space	1,278,000 KB
Page File	C:\pagefile.sys

The SQL Server Client Utilities were installed on each client from Microsoft SQL Server Build 8.00.194. See the file syswfe_sql.v1.RTF for a screen capture of the client SQL code version.

System Information Report for Hardware Configuration

This hardware configuration inventory was obtained from 1 of 48 clients.
All 48 clients are identically configured.

System Information report written at: 03/05/2001 05:32:34 PM
[System Information]

[Following are sub-categories of this main category]

[System Summary]

Item	Value
OS Name	Microsoft Windows 2000 Server
Version	5.0.2195 Build 2195
OS Manufacturer	Microsoft Corporation
System Name	RTCLIENT02
System Manufacturer	IBM
System Model	Netfinity 5100 -[86582RY]-
System Type	X86-based PC
Processor	x86 Family 6 Model 8 Stepping 3 GenuineIntel ~731 Mhz
Processor	x86 Family 6 Model 8 Stepping 3 GenuineIntel ~731 Mhz
BIOS Version	IBM BIOS Ver 3.0
Windows Directory	C:\WINNT
System Directory	C:\WINNT\System32
Boot Device	\Device\Harddisk0\Partition1
Locale	United States
User Name	RTCLIENT02\tpcc
Time Zone	Eastern Standard Time
Total Physical Memory	523,796 KB
Available Physical Memory	427,168 KB
Total Virtual Memory	1,801,796 KB
Available Virtual Memory	1,632,624 KB
Page File Space	1,278,000 KB
Page File	C:\pagefile.sys

[Hardware Resources]

[Following are sub-categories of this main category]

[Conflicts/Sharing]

Resource	Device
No conflicted/shared resources	

[DMA]

Channel	Device	Status
2	Standard floppy disk controller	OK
4	Direct memory access controller	OK

[Forced Hardware]

Device	PNP Device ID
No Forced Hardware	

[I/O]

Address Range	Device	Status
0x0000-0x0CF7	PCI bus	OK
0x0000-0x0CF7	Direct memory access controller	OK
0x0D00-0x4AEB	PCI bus	OK
0x82E8-0xFFFF	PCI bus	OK
0x03B0-0x03BB	S3 Inc. Savage4	OK
0x03C0-0x03DF	S3 Inc. Savage4	OK
0x2000-0x201F	IBM Netfinity Fault Tolerance PCI Adapter	OK
0x0A79-0x0A79	ISAPNP Read Data Port	OK
0x0279-0x0279	ISAPNP Read Data Port	OK
0x02F4-0x02F7	ISAPNP Read Data Port	OK
0x0060-0x0060	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard	OK
0x0064-0x0064	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard	OK
0x03F0-0x03F5	Standard floppy disk controller	OK
0x03F7-0x03F7	Standard floppy disk controller	OK
0x0378-0x037F	Printer Port (LPT1)	OK
0x03F8-0x03FF	Communications Port (COM1)	OK
0x02F8-0x02FF	Communications Port (COM2)	OK
0x0020-0x0021	Advanced programmable interrupt controller	OK
0x00A0-0x00A1	Advanced programmable interrupt controller	OK
0x04D0-0x04D1	Advanced programmable interrupt controller	OK
0x0080-0x008F	Direct memory access controller	OK
0x00C0-0x00DF	Direct memory access controller	OK
0x0040-0x0043	System timer	OK
0x0070-0x0073	System CMOS/real time clock	OK
0x0061-0x0061	System speaker	OK
0x00F0-0x00FF	Numeric data processor	OK
0x0F50-0x0F58	Motherboard resources	OK
0xFD00-0xFD3F	Motherboard resources	OK
0xFE80-0xFEBF	Motherboard resources	OK
0xFEC0-0xFEDF	Motherboard resources	OK
0xEE9B-0xEE9B	Motherboard resources	OK
0x00E8-0x00E9	Not Available	OK
0x0840-0x084F	Standard Dual Channel PCI IDE Controller	OK
0x01F0-0x01F7	Primary IDE Channel	OK
0x03F6-0x03F6	Primary IDE Channel	OK
0x0170-0x0177	Secondary IDE Channel	OK
0x0376-0x0376	Secondary IDE Channel	OK
0x4AEC-0x82E7	PCI bus	OK
0x4B00-0x4BFF	Adaptec AIC-7899 Ultra160/m PCI SCSI Card	OK
0x4C00-0x4CFF	Adaptec AIC-7899 Ultra160/m PCI SCSI Card	OK
0x7000-0x7FFF	DEC 21152 PCI to PCI bridge	OK
0x7100-0x711F	Intel(R) PRO/100+ Dual Port Server Adapter	OK
0x7120-0x713F	Intel(R) PRO/100+ Dual Port Server Adapter #2	OK
0x6000-0x6FFF	DEC 21152 PCI to PCI bridge	OK
0x6100-0x611F	Intel(R) PRO/100+ Dual Port Server Adapter #3	OK
0x6120-0x613F	Intel(R) PRO/100+ Dual Port Server Adapter #4	OK
0x5000-0x5FFF	DEC 21152 PCI to PCI bridge	OK
0x5100-0x511F	Intel(R) PRO/100+ Dual Port Server Adapter #5	OK
0x5120-0x513F	Intel(R) PRO/100+ Dual Port Server Adapter #6	OK

[IRQs]

IRQ Number	Device
30	Microsoft ACPI-Compliant System

31	S3 Inc. Savage4
27	IBM Netfinity Fault Tolerance PCI Adapter
1	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard
12	PS/2 Compatible Mouse
6	Standard floppy disk controller
4	Communications Port (COM1)
3	Communications Port (COM2)
8	System CMOS/real time clock
13	Numeric data processor
5	Not Available
14	Primary IDE Channel
11	Standard OpenHCD USB Host Controller
28	Adaptec AIC-7899 Ultra160/m PCI SCSI Card
29	Adaptec AIC-7899 Ultra160/m PCI SCSI Card
20	Intel(R) PRO/100+ Dual Port Server Adapter
21	Intel(R) PRO/100+ Dual Port Server Adapter #2
22	Intel(R) PRO/100+ Dual Port Server Adapter #3
23	Intel(R) PRO/100+ Dual Port Server Adapter #4
24	Intel(R) PRO/100+ Dual Port Server Adapter #5
25	Intel(R) PRO/100+ Dual Port Server Adapter #6

[Memory]

Range	Device	Status
0xC8000-0xCFFFF	PCI bus	OK
0xC8000-0xCFFFF	System board	OK
0xF0000000-0xFEBFFFFFFF	PCI bus	OK
0xF0000000-0xFEBFFFFFFF	S3 Inc. Savage4	OK
0xFED00000-0xFEDFFFFFFF	PCI bus	OK
0xFE000000-0xFFDFFFFFFF	PCI bus	OK
0xFEB80000-0xFEBFFFFFFF	S3 Inc. Savage4	OK
0xA0000-0xBFFFF	S3 Inc. Savage4	OK
0xFEB7FC00-0xFEB7FC1F	IBM Netfinity Fault Tolerance PCI Adapter	OK
0xFEC00000-0xFECFFFFFFF	Advanced programmable interrupt controller	OK
0xFEE00000-0xFEEFFFFFFF	Advanced programmable interrupt controller	OK
0xFF700000-0xFF700FFF	Standard OpenHCD USB Host Controller	OK
0xD0000-0xDFFFF	PCI bus	OK
0x20000000-0xEFFFFFFF	PCI bus	OK
0xEFFFF000-0xEFFFFFFF	Adaptec AIC-7899 Ultra160/m PCI SCSI Card	OK
0xEFFFE000-0xEFFFEFFF	Adaptec AIC-7899 Ultra160/m PCI SCSI Card	OK
0xEA000000-0xEBAFFFFFFF	DEC 21152 PCI to PCI bridge	OK
0xE8800000-0xE8FFFFFFF	DEC 21152 PCI to PCI bridge	OK
0xEBA00000-0xEBA00FFF	Intel(R) PRO/100+ Dual Port Server Adapter	OK
0xEB800000-0xEB8FFFFFFF	Intel(R) PRO/100+ Dual Port Server Adapter	OK
0xEBA01000-0xEBA01FFF	Intel(R) PRO/100+ Dual Port Server Adapter #2	OK
0xEB900000-0xEB9FFFFFFF	Intel(R) PRO/100+ Dual Port Server Adapter #2	OK
0xEC000000-0xEDAFFFFFFF	DEC 21152 PCI to PCI bridge	OK
0xE9000000-0xE97FFFFFFF	DEC 21152 PCI to PCI bridge	OK
0xEDA00000-0xEDA00FFF	Intel(R) PRO/100+ Dual Port Server Adapter #3	OK
0xED800000-0xED8FFFFFFF	Intel(R) PRO/100+ Dual Port Server Adapter #3	OK
0xEDA01000-0xEDA01FFF	Intel(R) PRO/100+ Dual Port Server Adapter #4	OK
0xED900000-0xED9FFFFFFF	Intel(R) PRO/100+ Dual Port Server Adapter #4	OK
0xEE000000-0xEFAFFFFFFF	DEC 21152 PCI to PCI bridge	OK
0xE9800000-0xE9FFFFFFF	DEC 21152 PCI to PCI bridge	OK

```

0xEFA00000-0xEFA00FFF Intel(R) PRO/100+ Dual Port Server
Adapter #5 OK
0xEF800000-0xEF8FFFFF Intel(R) PRO/100+ Dual Port Server
Adapter #5 OK
0xEFA01000-0xEFA01FFF Intel(R) PRO/100+ Dual Port Server
Adapter #6 OK
0xEF900000-0xEF9FFFFF Intel(R) PRO/100+ Dual Port Server
Adapter #6 OK
0x0000-0x9FFFFF System board OK
0x100000-0xFFFFFFFF System board OK
0x1000000-0x1FFFFFFF System board OK
0xE0000-0xFFFFF System board OK
0xFFFE0000-0xFFFFFFFF System board OK
0xCC000-0xCFFFF System board OK

```

[Components]

[Following are sub-categories of this main category]

[Multimedia]

[Following are sub-categories of this main category]

[Audio Codecs]

Codec Version	Manufacturer Size	Description Creation Date	Status	File
c:\winnt\system32\iac25_32.ax	OK	Intel Corporation	Indeo® audio software	2.05.53 195.00 KB (199,680 bytes) 12/7/1999 7:00:00 AM
c:\winnt\system32\msg723.acm	OK	Microsoft Corporation		106.77 KB (109,328 bytes) 8/24/2000 3:44:00 PM
c:\winnt\system32\lhacm.acm	OK	Microsoft Corporation		33.27 KB (34,064 bytes) 8/24/2000 3:44:01 PM
c:\winnt\system32\tsoft32.acm	OK	DSP GROUP, INC.		9.27 KB (9,488 bytes) 12/7/1999 7:00:00 AM
c:\winnt\system32\msgsm32.acm	OK	Microsoft Corporation		22.27 KB (22,800 bytes) 12/7/1999 7:00:00 AM
c:\winnt\system32\msg711.acm	OK	Microsoft Corporation		10.27 KB (10,512 bytes) 12/7/1999 7:00:00 AM
c:\winnt\system32\msadp32.acm	OK	Microsoft Corporation		14.77 KB (15,120 bytes) 12/7/1999 7:00:00 AM
c:\winnt\system32\imaadp32.acm	OK	Microsoft Corporation		16.27 KB (16,656 bytes) 12/7/1999 7:00:00 AM

[Video Codecs]

Codec Version	Manufacturer Size	Description Creation Date	Status	File
c:\winnt\system32\ir50_32.dll	OK	Intel Corporation	Indeo® video	5.10.15.2.55 737.50 KB (755,200 bytes) 12/7/1999 7:00:00 AM
c:\winnt\system32\msrle32.dll	OK	Microsoft Corporation		10.77 KB (11,024 bytes) 12/7/1999 7:00:00 AM
c:\winnt\system32\msh263.drv	OK	Microsoft Corporation		252.27 KB (258,320 bytes) 8/24/2000 3:43:29 PM

```

c:\winnt\system32\msh261.drv Microsoft Corporation
OK C:\WINNT\System32\MSH261.DRV 4.4.3385
163.77 KB (167,696 bytes) 8/24/2000 3:44:00 PM
c:\winnt\system32\ir32_32.dll Intel(R) Corporation OK
C:\WINNT\System32\IR32_32.DLL Not Available
194.50 KB (199,168 bytes) 12/7/1999 7:00:00 AM
c:\winnt\system32\msvidc32.dll Microsoft Corporation
OK C:\WINNT\System32\MSVIDC32.DLL 5.00.2134.1
27.27 KB (27,920 bytes) 12/7/1999 7:00:00 AM
c:\winnt\system32\iccvld.dll Radius Inc. OK
C:\WINNT\System32\ICCVLD.DLL 1.10.0.6 108.00 KB
(110,592 bytes) 12/7/1999 7:00:00 AM

```

[CD-ROM]

Item	Value
Drive	D:
Description	CD-ROM Drive
Media Loaded	False
Media Type	CD-ROM
Name	LITEON CD-ROM LTN403
Manufacturer	(Standard CD-ROM drives)
Status	OK
Transfer Rate	Not Available
SCSI Target ID	0
PNP Device ID	IDE\CDROMLITEON_CD-ROM_LTN403_____D
U26_____5&326853DD&0&0.0.0	

[Sound Device]

Item	Value
No sound devices	

[Display]

Item	Value
Name	S3 Inc. Savage4
PNP Device ID	PCI\VEN_5333&DEV_8A22&SUBSYS_01C51014&REV_04\3&267A616A&0&08
Adapter Type	S3 Savage4, S3 compatible
Adapter Description	S3 Inc. Savage4
Adapter RAM	8.00 MB (8,388,608 bytes)
Installed Drivers	s3sav4.dll
Driver Version	5.01.840.0001
INF File	s3sav4.inf (S3Inc section)
Color Planes	1
Color Table Entries	4294967296
Resolution	800 x 600 x 60 hertz
Bits/Pixel	32

[Infrared]

Item	Value
No infrared devices	

[Input]

[Following are sub-categories of this main category]

[Keyboard]

Item	Value
Description	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard

Name Enhanced (101- or 102-key)
 Layout 00000409
 PNP Device ID ACPI\PNP0303\4&F0B8F99&0
 NumberOfFunctionKeys 12

[Pointing Device]

Item Value
 Hardware Type PS/2 Compatible Mouse
 Number of Buttons 3
 Status OK
 PNP Device ID ACPI\PNP0F13\4&F0B8F99&0
 Power Management Supported False
 Double Click Threshold 6
 Handedness Right Handed Operation

[Modem]

Item Value
 Name 3Com Windows Modem PCI ADI
 Description 3Com Windows Modem PCI ADI
 Device ID Modem0
 Device Type Internal Modem
 Attached To COM3
 Answer Mode Not Available
 PNP Device ID Not Available
 ProviderName 3Com
 ModemInfPath mdm3cisa.inf
 ModemInfSection ADIPCI_Device
 BlindOff X4
 BlindOn X3
 CompressionOff &K0S15=128
 CompressionOn &K1
 ErrorControlForced &M5
 ErrorControlOff &M0S15=128
 ErrorControlOn &M4
 FlowControlHard &H1&R2&I0
 FlowControlOff &H0&R1&I0
 FlowControlSoft &H2&R1&I2
 DCB
 Default <
 InactivityTimeout 0
 ModulationBell B1
 ModulationCCITT B0
 Prefix AT
 Properties b
 Pulse P
 Reset ATZ<cr>
 ResponsesKeyName 3Com Windows Modem PCI ADI::3Com::3Com
 SpeakerModeDial M1
 SpeakerModeOff M0
 SpeakerModeOn M2
 SpeakerModeSetup M3
 SpeakerVolumeHigh L3
 SpeakerVolumeLow L0
 SpeakerVolumeMed L2
 StringFormat Not Available
 Terminator <cr>
 Tone T

Name U.S. Robotics 56K FAX EXT PnP
 Description U.S. Robotics 56K FAX EXT PnP
 Device ID Modem1
 Device Type External Modem
 Attached To COM1
 Answer Mode Not Available

PNP Device ID Not Available
 ProviderName 3Com
 ModemInfPath mdm3com.inf
 ModemInfSection ModemPCMex
 BlindOff X4
 BlindOn X3
 CompressionOff &K0
 CompressionOn &K1
 ErrorControlForced &M5
 ErrorControlOff &M0
 ErrorControlOn &M4
 FlowControlHard &H1&R2&I0
 FlowControlOff &H0&I0&R1
 FlowControlSoft &H2&I2&R1
 DCB "
 Default <
 InactivityTimeout 0
 ModulationBell B1
 ModulationCCITT B0
 Prefix AT
 Properties b
 Pulse P
 Reset AT&F1<cr>
 ResponsesKeyName U.S. Robotics 56K FAX EXT
 PnP::3Com::3Com
 SpeakerModeDial M1
 SpeakerModeOff M0
 SpeakerModeOn M2
 SpeakerModeSetup M3
 SpeakerVolumeHigh L3
 SpeakerVolumeLow L0
 SpeakerVolumeMed L2
 StringFormat Not Available
 Terminator <cr>
 Tone T

[Network]

[Following are sub-categories of this main category]

[Adapter]

Item Value
 Name [00000000] IBM Netfinity Fault Tolerance PCI Adapter
 Adapter Type Ethernet 802.3
 Product Name IBM Netfinity Fault Tolerance PCI Adapter
 Installed True
 PNP Device ID PCI\VEN_1022&DEV_2000&SUBSYS_20001014&REV_44\3&267A616A&0&10
 Last Reset 3/5/2001 12:23:09 PM
 Index 0
 Service Name PCNet5
 IP Address 192.6.1.18
 IP Subnet 255.255.255.0
 Default IP Gateway Not Available
 DHCP Enabled False
 DHCP Server Not Available
 DHCP Lease Expires Not Available
 DHCP Lease Obtained Not Available
 MAC Address 00:06:29:D5:77:83
 Service Name PCnet
 IRQ Number 27
 I/O Port 0x2000-0x201F
 Driver c:\winnt\system32\drivers\pcent5m.sys (33811, 4.23.00)
 Name [00000001] Intel(R) PRO/100+ Dual Port Server Adapter

Adapter Type Ethernet 802.3
 Product Name Intel(R) PRO/100+ Dual Port Server Adapter
 Installed True
 PNP Device ID
 PCI\VEN_8086&DEV_1229&SUBSYS_10F08086&REV_05\4&15F5026D&0&2028
 Last Reset 3/5/2001 12:23:09 PM
 Index 1
 Service Name E100B
 IP Address 192.1.7.1
 IP Subnet 255.255.255.0
 Default IP Gateway Not Available
 DHCP Enabled False
 DHCP Server Not Available
 DHCP Lease Expires Not Available
 DHCP Lease Obtained Not Available
 MAC Address 00:90:27:EE:96:3E
 Service Name E100B
 IRQ Number 20
 I/O Port 0x7100-0x711F
 Driver c:\winnt\system32\drivers\e100bnt5.sys (80144, 4.01.67.0000)

Name [00000002] Intel(R) PRO/100+ Dual Port Server Adapter
 Adapter Type Ethernet 802.3
 Product Name Intel(R) PRO/100+ Dual Port Server Adapter
 Installed True
 PNP Device ID
 PCI\VEN_8086&DEV_1229&SUBSYS_10F08086&REV_05\4&15F5026D&0&2828
 Last Reset 3/5/2001 12:23:09 PM
 Index 2
 Service Name E100B
 IP Address 192.1.8.1
 IP Subnet 255.255.255.0
 Default IP Gateway Not Available
 DHCP Enabled False
 DHCP Server Not Available
 DHCP Lease Expires Not Available
 DHCP Lease Obtained Not Available
 MAC Address 00:90:27:EE:96:3F
 Service Name E100B
 IRQ Number 21
 I/O Port 0x7120-0x713F
 Driver c:\winnt\system32\drivers\e100bnt5.sys (80144, 4.01.67.0000)

Name [00000003] Intel(R) PRO/100+ Dual Port Server Adapter
 Adapter Type Ethernet 802.3
 Product Name Intel(R) PRO/100+ Dual Port Server Adapter
 Installed True
 PNP Device ID
 PCI\VEN_8086&DEV_1229&SUBSYS_10F08086&REV_05\4&3286C9C3&0&2030
 Last Reset 3/5/2001 12:23:09 PM
 Index 3
 Service Name E100B
 IP Address 192.1.9.1
 IP Subnet 255.255.255.0
 Default IP Gateway Not Available
 DHCP Enabled False
 DHCP Server Not Available
 DHCP Lease Expires Not Available
 DHCP Lease Obtained Not Available
 MAC Address 00:90:27:EE:9B:B0
 Service Name E100B
 IRQ Number 22
 I/O Port 0x6100-0x611F

Driver c:\winnt\system32\drivers\e100bnt5.sys (80144, 4.01.67.0000)
 Name [00000004] Intel(R) PRO/100+ Dual Port Server Adapter
 Adapter Type Ethernet 802.3
 Product Name Intel(R) PRO/100+ Dual Port Server Adapter
 Installed True
 PNP Device ID
 PCI\VEN_8086&DEV_1229&SUBSYS_10F08086&REV_05\4&3286C9C3&0&2830
 Last Reset 3/5/2001 12:23:09 PM
 Index 4
 Service Name E100B
 IP Address 192.1.10.1
 IP Subnet 255.255.255.0
 Default IP Gateway Not Available
 DHCP Enabled False
 DHCP Server Not Available
 DHCP Lease Expires Not Available
 DHCP Lease Obtained Not Available
 MAC Address 00:90:27:EE:9B:B1
 Service Name E100B
 IRQ Number 23
 I/O Port 0x6120-0x613F
 Driver c:\winnt\system32\drivers\e100bnt5.sys (80144, 4.01.67.0000)

Name [00000005] Intel(R) PRO/100+ Dual Port Server Adapter
 Adapter Type Ethernet 802.3
 Product Name Intel(R) PRO/100+ Dual Port Server Adapter
 Installed True
 PNP Device ID
 PCI\VEN_8086&DEV_1229&SUBSYS_10F08086&REV_05\4&27ABF15E&0&2038
 Last Reset 3/5/2001 12:23:09 PM
 Index 5
 Service Name E100B
 IP Address 192.1.11.1
 IP Subnet 255.255.255.0
 Default IP Gateway Not Available
 DHCP Enabled False
 DHCP Server Not Available
 DHCP Lease Expires Not Available
 DHCP Lease Obtained Not Available
 MAC Address 00:90:27:FC:39:18
 Service Name E100B
 IRQ Number 24
 I/O Port 0x5100-0x511F
 Driver c:\winnt\system32\drivers\e100bnt5.sys (80144, 4.01.67.0000)

Name [00000006] Intel(R) PRO/100+ Dual Port Server Adapter
 Adapter Type Ethernet 802.3
 Product Name Intel(R) PRO/100+ Dual Port Server Adapter
 Installed True
 PNP Device ID
 PCI\VEN_8086&DEV_1229&SUBSYS_10F08086&REV_05\4&27ABF15E&0&2838
 Last Reset 3/5/2001 12:23:09 PM
 Index 6
 Service Name E100B
 IP Address 192.1.12.1
 IP Subnet 255.255.255.0
 Default IP Gateway Not Available
 DHCP Enabled False
 DHCP Server Not Available
 DHCP Lease Expires Not Available
 DHCP Lease Obtained Not Available

MAC Address 00:90:27:FC:39:19
Service Name E100B
IRQ Number 25
I/O Port 0x5120-0x513F
Driver c:\winnt\system32\drivers\e100bnt5.sys (80144, 4.01.67.0000)

Name [00000007] RAS Async Adapter
Adapter Type Not Available
Product Name RAS Async Adapter
Installed True
PNP Device ID Not Available
Last Reset 3/5/2001 12:23:09 PM
Index 7
Service Name AsyncMac
IP Address Not Available
IP Subnet Not Available
Default IP Gateway Not Available
DHCP Enabled False
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address Not Available
Service Name Not Available

Name [00000008] WAN Miniport (L2TP)
Adapter Type Not Available
Product Name WAN Miniport (L2TP)
Installed True
PNP Device ID ROOT\MS_L2TPMINIPOINT\0000
Last Reset 3/5/2001 12:23:09 PM
Index 8
Service Name Rasl2tp
IP Address Not Available
IP Subnet Not Available
Default IP Gateway Not Available
DHCP Enabled False
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address Not Available
Service Name Rasl2tp
Driver c:\winnt\system32\drivers\rasl2tp.sys (50800, 5.00.2179.1)

Name [00000009] WAN Miniport (PPTP)
Adapter Type Wide Area Network (WAN)
Product Name WAN Miniport (PPTP)
Installed True
PNP Device ID ROOT\MS_PPTPMINIPOINT\0000
Last Reset 3/5/2001 12:23:09 PM
Index 9
Service Name PptpMiniport
IP Address Not Available
IP Subnet Not Available
Default IP Gateway Not Available
DHCP Enabled False
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address 50:50:54:50:30:30
Service Name PptpMiniport
Driver c:\winnt\system32\drivers\raspptp.sys (47856, 5.00.2160.1)

Name [00000010] Direct Parallel
Adapter Type Not Available
Product Name Direct Parallel
Installed True
PNP Device ID ROOT\MS_PTIMINIPOINT\0000

Last Reset 3/5/2001 12:23:09 PM
Index 10
Service Name Raspti
IP Address Not Available
IP Subnet Not Available
Default IP Gateway Not Available
DHCP Enabled False
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address Not Available
Service Name Raspti
Driver c:\winnt\system32\drivers\raspti.sys (16880, 5.00.2146.1)

Name [00000011] WAN Miniport (IP)
Adapter Type Not Available
Product Name WAN Miniport (IP)
Installed True
PNP Device ID ROOT\MS_NDISWANIP\0000
Last Reset 3/5/2001 12:23:09 PM
Index 11
Service Name NdisWan
IP Address Not Available
IP Subnet Not Available
Default IP Gateway Not Available
DHCP Enabled False
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address Not Available
Service Name NdisWan
Driver c:\winnt\system32\drivers\ndiswan.sys (90768, 5.00.2184.1)

Name [00000012] Intel(R) PRO/100+ Dual Port Server Adapter
Adapter Type Not Available
Product Name Intel(R) PRO/100+ Dual Port Server Adapter
Installed True
PNP Device ID Not Available
Last Reset 3/5/2001 12:23:09 PM
Index 12
Service Name E100B
IP Address Not Available
IP Subnet Not Available
Default IP Gateway Not Available
DHCP Enabled True
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address Not Available
Service Name Not Available

Name [00000013] Intel(R) PRO/100+ Dual Port Server Adapter
Adapter Type Not Available
Product Name Intel(R) PRO/100+ Dual Port Server Adapter
Installed True
PNP Device ID Not Available
Last Reset 3/5/2001 12:23:09 PM
Index 13
Service Name E100B
IP Address 192.6.1.18
IP Subnet 255.255.255.0
Default IP Gateway Not Available
DHCP Enabled True
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address 00:06:29:D5:77:83

Service Name Not Available

[Protocol]

Item Value

Name MSAFD Tcpi [TCP/IP]

ConnectionlessService False

GuaranteesDelivery True

GuaranteesSequencing True

MaximumAddressSize 16 bytes

MaximumMessageSize 0 bytes

MessageOriented False

MinimumAddressSize 16 bytes

PseudoStreamOriented False

SupportsBroadcasting False

SupportsConnectData False

SupportsDisconnectData False

SupportsEncryption False

SupportsExpeditedData True

SupportsGracefulClosing True

SupportsGuaranteedBandwidth False

SupportsMulticasting False

Name MSAFD Tcpi [UDP/IP]

ConnectionlessService True

GuaranteesDelivery False

GuaranteesSequencing False

MaximumAddressSize 16 bytes

MaximumMessageSize 65467 bytes

MessageOriented True

MinimumAddressSize 16 bytes

PseudoStreamOriented False

SupportsBroadcasting True

SupportsConnectData False

SupportsDisconnectData False

SupportsEncryption False

SupportsExpeditedData False

SupportsGracefulClosing False

SupportsGuaranteedBandwidth False

SupportsMulticasting True

Name RSVP UDP Service Provider

ConnectionlessService True

GuaranteesDelivery False

GuaranteesSequencing False

MaximumAddressSize 16 bytes

MaximumMessageSize 65467 bytes

MessageOriented True

MinimumAddressSize 16 bytes

PseudoStreamOriented False

SupportsBroadcasting True

SupportsConnectData False

SupportsDisconnectData False

SupportsEncryption True

SupportsExpeditedData False

SupportsGracefulClosing False

SupportsGuaranteedBandwidth False

SupportsMulticasting True

Name RSVP TCP Service Provider

ConnectionlessService False

GuaranteesDelivery True

GuaranteesSequencing True

MaximumAddressSize 16 bytes

MaximumMessageSize 0 bytes

MessageOriented False

MinimumAddressSize 16 bytes

PseudoStreamOriented False

SupportsBroadcasting False

SupportsConnectData False

SupportsDisconnectData False

SupportsEncryption True

SupportsExpeditedData True

SupportsGracefulClosing True

SupportsGuaranteedBandwidth False

SupportsMulticasting False

Name MSAFD NetBIOS

[\\Device\\NetBT_Tcpi_{A0002A43-3B3B-4624-9900-1BB24D86C448}] SEQPACKET 10

ConnectionlessService False

GuaranteesDelivery True

GuaranteesSequencing True

MaximumAddressSize 20 bytes

MaximumMessageSize 64000 bytes

MessageOriented True

MinimumAddressSize 20 bytes

PseudoStreamOriented False

SupportsBroadcasting False

SupportsConnectData False

SupportsDisconnectData False

SupportsEncryption False

SupportsExpeditedData False

SupportsGracefulClosing False

SupportsGuaranteedBandwidth False

SupportsMulticasting False

Name MSAFD NetBIOS

[\\Device\\NetBT_Tcpi_{A0002A43-3B3B-4624-9900-1BB24D86C448}] DATAGRAM 10

ConnectionlessService True

GuaranteesDelivery False

GuaranteesSequencing False

MaximumAddressSize 20 bytes

MaximumMessageSize 64000 bytes

MessageOriented True

MinimumAddressSize 20 bytes

PseudoStreamOriented False

SupportsBroadcasting True

SupportsConnectData False

SupportsDisconnectData False

SupportsEncryption False

SupportsExpeditedData False

SupportsGracefulClosing False

SupportsGuaranteedBandwidth False

SupportsMulticasting False

Name MSAFD NetBIOS

[\\Device\\NetBT_Tcpi_{71E47429-1922-473C-BAD4-293446B8330D}] SEQPACKET 9

ConnectionlessService False

GuaranteesDelivery True

GuaranteesSequencing True

MaximumAddressSize 20 bytes

MaximumMessageSize 64000 bytes

MessageOriented True

MinimumAddressSize 20 bytes

PseudoStreamOriented False

SupportsBroadcasting False

SupportsConnectData False

SupportsDisconnectData False

SupportsEncryption False

SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT_Tcpip_{71E47429-1922-473C-BAD4-293446B8330D}] DATAGRAM 9

ConnectionlessService True
 GuaranteesDelivery False
 GuaranteesSequencing False
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting True
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT_Tcpip_{E27E7E51-7F48-467C-AE8A-CE4BD42A72CA}] SEQPACKE T 0

ConnectionlessService False
 GuaranteesDelivery True
 GuaranteesSequencing True
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting False
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT_Tcpip_{E27E7E51-7F48-467C-AE8A-CE4BD42A72CA}] DATAGRAM 0

ConnectionlessService True
 GuaranteesDelivery False
 GuaranteesSequencing False
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting True
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT_Tcpip_{382205FA-9666-48B6-8057-8BE0D9A10729}] SEQPACKE T 1

ConnectionlessService False
 GuaranteesDelivery True
 GuaranteesSequencing True
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting False
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT_Tcpip_{382205FA-9666-48B6-8057-8BE0D9A10729}] DATAGRAM 1

ConnectionlessService True
 GuaranteesDelivery False
 GuaranteesSequencing False
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting True
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT_Tcpip_{51B7711C-5716-4D5D-B91B-89A576C82962}] SEQPACKE T 2

ConnectionlessService False
 GuaranteesDelivery True
 GuaranteesSequencing True
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting False
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT_Tcpip_{51B7711C-5716-4D5D-B91B-89A576C82962}] DATAGRAM 2

ConnectionlessService True

GuaranteesDelivery False
 GuaranteesSequencing False
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting True
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS

[\\Device\NetBT_Tcpip_{43CB931E-C90D-475C-9CAA-FF9812C70086}] SEQUENCE 3
 ConnectionlessService False
 GuaranteesDelivery True
 GuaranteesSequencing True
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting False
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS

[\\Device\NetBT_Tcpip_{43CB931E-C90D-475C-9CAA-FF9812C70086}] DATAGRAM 3
 ConnectionlessService True
 GuaranteesDelivery False
 GuaranteesSequencing False
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting True
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS

[\\Device\NetBT_Tcpip_{DEE0939D-006C-4A4D-8562-1F194D19A758}] SEQUENCE 4
 ConnectionlessService False
 GuaranteesDelivery True
 GuaranteesSequencing True
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True

MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting False
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS

[\\Device\NetBT_Tcpip_{DEE0939D-006C-4A4D-8562-1F194D19A758}] DATAGRAM 4
 ConnectionlessService True
 GuaranteesDelivery False
 GuaranteesSequencing False
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting True
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS

[\\Device\NetBT_Tcpip_{B844D407-10F6-48EC-A318-2F0684C97622}] SEQUENCE 5
 ConnectionlessService False
 GuaranteesDelivery True
 GuaranteesSequencing True
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting False
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS

[\\Device\NetBT_Tcpip_{B844D407-10F6-48EC-A318-2F0684C97622}] DATAGRAM 5
 ConnectionlessService True
 GuaranteesDelivery False
 GuaranteesSequencing False
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting True
 SupportsConnectData False
 SupportsDisconnectData False

SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT_Tcpip_{3D4874F5-B3F5-46CC-971E-8417C74F7222}] SEQPACKET 6

ConnectionlessService False
 GuaranteesDelivery True
 GuaranteesSequencing True
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting False
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT_Tcpip_{3D4874F5-B3F5-46CC-971E-8417C74F7222}] DATAGRAM 6

ConnectionlessService True
 GuaranteesDelivery False
 GuaranteesSequencing False
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting True
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT_Tcpip_{97BE54C2-5CD7-41B5-92D2-5E2A10F376D7}] SEQPACKET 7

ConnectionlessService False
 GuaranteesDelivery True
 GuaranteesSequencing True
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting False
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT_Tcpip_{97BE54C2-5CD7-41B5-92D2-5E2A10F376D7}] DATAGRAM 7

ConnectionlessService True
 GuaranteesDelivery False
 GuaranteesSequencing False
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting True
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT_Tcpip_{B71D38CE-D026-4A75-A265-AF35F16F7BDC}] SEQPACKET 8

ConnectionlessService False
 GuaranteesDelivery True
 GuaranteesSequencing True
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting False
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS

[\Device\NetBT_Tcpip_{B71D38CE-D026-4A75-A265-AF35F16F7BDC}] DATAGRAM 8

ConnectionlessService True
 GuaranteesDelivery False
 GuaranteesSequencing False
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting True
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

[WinSock]

Item Value

File c:\winnt\system32\winsock.dll
 Version 3.10
 Size 2.80 KB (2,864 bytes)

File c:\winnt\system32\wsock32.dll
 Version 5.00.2152.1
 Size 21.27 KB (21,776 bytes)

[Ports]

[Following are sub-categories of this main category]

[Serial]

Item	Value
Name	COM1
Status	OK
PNP Device ID	ACPI\PNP0501\1
Maximum Input Buffer Size	0
Maximum Output Buffer Size	False
Settable Baud Rate	True
Settable Data Bits	True
Settable Flow Control	True
Settable Parity	True
Settable Parity Check	True
Settable Stop Bits	True
Settable RLSD	True
Supports RLSD	True
Supports 16 Bit Mode	False
Supports Special Characters	False
Baud Rate	9600
Bits/Byte	8
Stop Bits	1
Parity	None
Busy	0
Abort Read/Write on Error	0
Binary Mode Enabled-1	0
Continue XMit on XOff	0
CTS Outflow Control	0
Discard NULL Bytes	0
DSR Outflow Control	0
DSR Sensitivity	0
DTR Flow Control Type	Enable
EOF Character	0
Error Replace Character	0
Error Replacement Enabled	0
Event Character	0
Parity Check Enabled	0
RTS Flow Control Type	Enable
XOff Character	19
XOffXMit Threshold	512
XOn Character	17
XOnXMit Threshold	2048
XOnXOff InFlow Control	0
XOnXOff OutFlow Control	0
IRQ Number	4
I/O Port	0x03F8-0x03FF
Driver	c:\winnt\system32\drivers\serial.sys (62448, 5.00.2134.1)

Name	COM2
Status	OK
PNP Device ID	ACPI\PNP0501\2
Maximum Input Buffer Size	0
Maximum Output Buffer Size	False
Settable Baud Rate	True
Settable Data Bits	True
Settable Flow Control	True

Settable Parity	True
Settable Parity Check	True
Settable Stop Bits	True
Settable RLSD	True
Supports RLSD	True
Supports 16 Bit Mode	False
Supports Special Characters	False
Baud Rate	9600
Bits/Byte	8
Stop Bits	1
Parity	None
Busy	0
Abort Read/Write on Error	0
Binary Mode Enabled-1	0
Continue XMit on XOff	0
CTS Outflow Control	0
Discard NULL Bytes	0
DSR Outflow Control	0
DSR Sensitivity	0
DTR Flow Control Type	Enable
EOF Character	0
Error Replace Character	0
Error Replacement Enabled	0
Event Character	0
Parity Check Enabled	0
RTS Flow Control Type	Enable
XOff Character	19
XOffXMit Threshold	512
XOn Character	17
XOnXMit Threshold	2048
XOnXOff InFlow Control	0
XOnXOff OutFlow Control	0
IRQ Number	3
I/O Port	0x02F8-0x02FF
Driver	c:\winnt\system32\drivers\serial.sys (62448, 5.00.2134.1)

[Parallel]

Item	Value
Name	LPT1
PNP Device ID	ACPI\PNP0400\1

[Storage]

[Following are sub-categories of this main category]

[Drives]

Item	Value
Drive	A:
Description	3 1/2 Inch Floppy Drive
Drive	C:
Description	Local Fixed Disk
Compressed	False
File System	NTFS
Size	8.47 GB (9,097,125,888 bytes)
Free Space	6.82 GB (7,327,989,760 bytes)
Volume Name	
Volume Serial Number	08F069E2
Partition	Disk #0, Partition #0
Partition Size	8.47 GB (9,097,127,424 bytes)
Starting Offset	32256 bytes
Drive Description	Disk drive
Drive Manufacturer	(Standard disk drives)
Drive Model	IBM-PSG DMVS09D !# SCSI Disk Device

Drive BytesPerSector 512
 Drive MediaLoaded True
 Drive MediaType Fixed hard disk media
 Drive Partitions 1
 Drive SCSIbus 0
 Drive SCSILogicalUnit 0
 Drive SCSIPort 2
 Drive SCISITargetId 0
 Drive SectorsPerTrack 63
 Drive Size 9097159680 bytes
 Drive TotalCylinders 1106
 Drive TotalSectors 17767890
 Drive TotalTracks 282030
 Drive TracksPerCylinder 255

Standard OpenHCD USB Host Controller
 PCI\VEN_1166&DEV_0220&SUBSYS_02201166&REV_04\3&267
 A616A&0&7A
 USB Root Hub USB\ROOT_HUB\4&372644EA&0

[SCSI]

Item Value
 Name Adaptec AIC-7899 Ultra160/m PCI SCSI Card
 Caption Adaptec AIC-7899 Ultra160/m PCI SCSI Card
 Driver adpu160m
 Status OK
 PNP Device ID
 PCI\VEN_9005&DEV_00CF&SUBSYS_019D1014&REV_01\3&13
 C0B0C5&0&18
 Device ID
 PCI\VEN_9005&DEV_00CF&SUBSYS_019D1014&REV_01\3&13
 C0B0C5&0&18
 Device Map Not Available
 Index Not Available
 Max Number Controlled Not Available
 IRQ Number 28
 I/O Port 0x4B00-0x4BFF
 Driver c:\winnt\system32\drivers\adpu160m.sys (64432, v3.10a)

Name Adaptec AIC-7899 Ultra160/m PCI SCSI Card
 Caption Adaptec AIC-7899 Ultra160/m PCI SCSI Card
 Driver adpu160m
 Status OK
 PNP Device ID
 PCI\VEN_9005&DEV_00CF&SUBSYS_019D1014&REV_01\3&13
 C0B0C5&0&19
 Device ID
 PCI\VEN_9005&DEV_00CF&SUBSYS_019D1014&REV_01\3&13
 C0B0C5&0&19
 Device Map Not Available
 Index Not Available
 Max Number Controlled Not Available
 IRQ Number 29
 I/O Port 0x4C00-0x4CFF
 Driver c:\winnt\system32\drivers\adpu160m.sys (64432, v3.10a)

[Printing]

Name Port Name Server Name
 Fax MSFAX: Not Available

[Problem Devices]

Device PNP Device ID Error Code
 Not Available ACPI\IBM37C0\4&F0B8F99&0 28

[USB]

Device PNP Device ID

RTE Input Parameters

Profile: 28800wh
File Path: C:\Benchcraft\28800wh.pro
Version: 3

Number of Engines: 288

Name: RTRTE01A
Description: 01A
Directory: c:\rtelogs\log01A.log
Machine: RTRTE01
Parameter Set: PARAM2
Index: 0
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE01A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE01B
Description: 01B
Directory: c:\rtelogs\log01B.log
Machine: RTRTE01
Parameter Set: PARAM2
Index: 3000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE01B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE01C
Description: 01C
Directory: c:\rtelogs\log01C.log
Machine: RTRTE01
Parameter Set: PARAM2
Index: 6000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE01C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE01D
Description: 01D
Directory: c:\rtelogs\log01D.log
Machine: RTRTE01
Parameter Set: PARAM2
Index: 9000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE01D670328
Connect Rate: 140
Start Rate: 0

Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE01E
Description: 01E
Directory: c:\rtelogs\log01E.log
Machine: RTRTE01
Parameter Set: PARAM2
Index: 12000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE01E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE01F
Description: 01F
Directory: c:\rtelogs\log01F.log
Machine: RTRTE01
Parameter Set: PARAM2
Index: 15000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE01F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE02A
Description: 02A
Directory: c:\rtelogs\log02A.log
Machine: RTRTE02
Parameter Set: PARAM2
Index: 18000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE02A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE02B
Description: 02B
Directory: c:\rtelogs\log02B.log
Machine: RTRTE02
Parameter Set: PARAM2
Index: 21000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE02B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE02C
Description: 02C
Directory: c:\rtelogs\log02C.log
Machine: RTRTE02
Parameter Set: PARAM2
Index: 24000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE02C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE02D
Description: 02D
Directory: c:\rtelogs\log02D.log
Machine: RTRTE02
Parameter Set: PARAM2
Index: 27000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE02D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE02E
Description: 02E
Directory: c:\rtelogs\log02E.log
Machine: RTRTE02
Parameter Set: PARAM2
Index: 30000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE02E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE02F
Description: 02F
Directory: c:\rtelogs\log02F.log
Machine: RTRTE02
Parameter Set: PARAM2
Index: 33000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE02F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE03A
Description: 03A
Directory: c:\rtelogs\log03A.log

Machine: RTRTE03
Parameter Set: PARAM2
Index: 36000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE03A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE03B
Description: 03B
Directory: c:\rtelogs\log03B.log
Machine: RTRTE03
Parameter Set: PARAM2
Index: 39000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE03B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE03C
Description: 03C
Directory: c:\rtelogs\log03C.log
Machine: RTRTE03
Parameter Set: PARAM2
Index: 42000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE03C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE03D
Description: 03D
Directory: c:\rtelogs\log03D.log
Machine: RTRTE03
Parameter Set: PARAM2
Index: 45000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE03D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE03E
Description: 03E
Directory: c:\rtelogs\log03E.log
Machine: RTRTE03
Parameter Set: PARAM2
Index: 48000000
Seed: 46329

Configured Users: 1000
Pipe Name: RTRTE03E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE03F
Description: 03F
Directory: c:\rtelogs\log03F.log
Machine: RTRTE03
Parameter Set: PARAM2
Index: 51000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE03F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE04A
Description: 04A
Directory: c:\rtelogs\log04A.log
Machine: RTRTE04
Parameter Set: PARAM2
Index: 54000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE04A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE04B
Description: 04B
Directory: c:\rtelogs\log04B.log
Machine: RTRTE04
Parameter Set: PARAM2
Index: 57000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE04B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE04C
Description: 04C
Directory: c:\rtelogs\log04C.log
Machine: RTRTE04
Parameter Set: PARAM2
Index: 60000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE04C670328
Connect Rate: 140
Start Rate: 0

Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE04D
Description: 04D
Directory: c:\rtelogs\log04D.log
Machine: RTRTE04
Parameter Set: PARAM2
Index: 63000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE04D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE04E
Description: 04E
Directory: c:\rtelogs\log04E.log
Machine: RTRTE04
Parameter Set: PARAM2
Index: 66000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE04E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE04F
Description: 04F
Directory: c:\rtelogs\log04F.log
Machine: RTRTE04
Parameter Set: PARAM2
Index: 69000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE04F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE05A
Description: 05A
Directory: c:\rtelogs\log05A.log
Machine: RTRTE05
Parameter Set: PARAM2
Index: 72000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE05A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE05B
Description: 05B
Directory: c:\rtelogs\log05B.log
Machine: RTRTE05
Parameter Set: PARAM2
Index: 75000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE05B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE05C
Description: 05C
Directory: c:\rtelogs\log05C.log
Machine: RTRTE05
Parameter Set: PARAM2
Index: 78000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE05C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE05D
Description: 05D
Directory: c:\rtelogs\log05D.log
Machine: RTRTE05
Parameter Set: PARAM2
Index: 81000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE05D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE05E
Description: 05E
Directory: c:\rtelogs\log05E.log
Machine: RTRTE05
Parameter Set: PARAM2
Index: 84000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE05E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE05F
Description: 05F
Directory: c:\rtelogs\log05F.log

Machine: RTRTE05
Parameter Set: PARAM2
Index: 87000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE05F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE06A
Description: 06A
Directory: c:\rtelogs\log06A.log
Machine: RTRTE06
Parameter Set: PARAM2
Index: 90000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE06A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE06B
Description: 06B
Directory: c:\rtelogs\log06B.log
Machine: RTRTE06
Parameter Set: PARAM2
Index: 93000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE06B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE06C
Description: 06C
Directory: c:\rtelogs\log06C.log
Machine: RTRTE06
Parameter Set: PARAM2
Index: 96000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE06C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE06D
Description: 06D
Directory: c:\rtelogs\log06D.log
Machine: RTRTE06
Parameter Set: PARAM2
Index: 99000000
Seed: 46329

Configured Users: 1000
Pipe Name: RTRTE06D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE06E
Description: 06E
Directory: c:\rtelogs\log06E.log
Machine: RTRTE06
Parameter Set: PARAM2
Index: 102000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE06E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE06F
Description: 06F
Directory: c:\rtelogs\log06F.log
Machine: RTRTE06
Parameter Set: PARAM2
Index: 105000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE06F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE07A
Description: 07A
Directory: c:\rtelogs\log07A.log
Machine: RTRTE07
Parameter Set: PARAM2
Index: 108000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE07A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE07B
Description: 07B
Directory: c:\rtelogs\log07B.log
Machine: RTRTE07
Parameter Set: PARAM2
Index: 111000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE07B670328
Connect Rate: 140
Start Rate: 0

Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE07C
Description: 07C
Directory: c:\rtelogs\log07C.log
Machine: RTRTE07
Parameter Set: PARAM2
Index: 114000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE07C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE07D
Description: 07D
Directory: c:\rtelogs\log07D.log
Machine: RTRTE07
Parameter Set: PARAM2
Index: 117000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE07D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE07E
Description: 07E
Directory: c:\rtelogs\log07E.log
Machine: RTRTE07
Parameter Set: PARAM2
Index: 120000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE07E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE07F
Description: 07F
Directory: c:\rtelogs\log07F.log
Machine: RTRTE07
Parameter Set: PARAM2
Index: 123000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE07F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE08A
Description: 08A
Directory: c:\rtelogs\log08A.log
Machine: RTRTE08
Parameter Set: PARAM2
Index: 126000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE08A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE08B
Description: 08B
Directory: c:\rtelogs\log08B.log
Machine: RTRTE08
Parameter Set: PARAM2
Index: 129000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE08B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE08C
Description: 08C
Directory: c:\rtelogs\log08C.log
Machine: RTRTE08
Parameter Set: PARAM2
Index: 132000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE08C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE08D
Description: 08D
Directory: c:\rtelogs\log08D.log
Machine: RTRTE08
Parameter Set: PARAM2
Index: 135000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE08D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE08E
Description: 08E
Directory: c:\rtelogs\log08E.log

Machine: RTRTE08
Parameter Set: PARAM2
Index: 138000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE08E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE08F
Description: 08F
Directory: c:\rtelogs\log08F.log
Machine: RTRTE08
Parameter Set: PARAM2
Index: 141000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE08F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE09A
Description: 09A
Directory: c:\rtelogs\log09A.log
Machine: RTRTE09
Parameter Set: PARAM2
Index: 144000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE09A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE09B
Description: 09B
Directory: c:\rtelogs\log09B.log
Machine: RTRTE09
Parameter Set: PARAM2
Index: 147000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE09B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE09C
Description: 09C
Directory: c:\rtelogs\log09C.log
Machine: RTRTE09
Parameter Set: PARAM2
Index: 150000000
Seed: 46329

Configured Users: 1000
Pipe Name: RTRTE09C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE09D
Description: 09D
Directory: c:\rtelogs\log09D.log
Machine: RTRTE09
Parameter Set: PARAM2
Index: 153000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE09D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE09E
Description: 09E
Directory: c:\rtelogs\log09E.log
Machine: RTRTE09
Parameter Set: PARAM2
Index: 156000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE09E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE09F
Description: 09F
Directory: c:\rtelogs\log09F.log
Machine: RTRTE09
Parameter Set: PARAM2
Index: 159000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE09F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE10A
Description: 10A
Directory: c:\rtelogs\log10A.log
Machine: RTRTE10
Parameter Set: PARAM2
Index: 162000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE10A670328
Connect Rate: 140
Start Rate: 0

Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE10B
Description: 10B
Directory: c:\rtelogs\log10B.log
Machine: RTRTE10
Parameter Set: PARAM2
Index: 165000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE10B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE10C
Description: 10C
Directory: c:\rtelogs\log10C.log
Machine: RTRTE10
Parameter Set: PARAM2
Index: 168000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE10C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE10D
Description: 10D
Directory: c:\rtelogs\log10D.log
Machine: RTRTE10
Parameter Set: PARAM2
Index: 171000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE10D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE10E
Description: 10E
Directory: c:\rtelogs\log10E.log
Machine: RTRTE10
Parameter Set: PARAM2
Index: 174000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE10E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE10F
Description: 10F
Directory: c:\rtelogs\log10F.log
Machine: RTRTE10
Parameter Set: PARAM2
Index: 177000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE10F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE11A
Description: 11A
Directory: c:\rtelogs\log11A.log
Machine: RTRTE11
Parameter Set: PARAM2
Index: 180000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE11A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE11B
Description: 11B
Directory: c:\rtelogs\log11B.log
Machine: RTRTE11
Parameter Set: PARAM2
Index: 183000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE11B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE11C
Description: 11C
Directory: c:\rtelogs\log11C.log
Machine: RTRTE11
Parameter Set: PARAM2
Index: 186000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE11C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE11D
Description: 11D
Directory: c:\rtelogs\log11D.log

Machine: RTRTE11
Parameter Set: PARAM2
Index: 189000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE11D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE11E
Description: 11E
Directory: c:\rtelogs\log11E.log
Machine: RTRTE11
Parameter Set: PARAM2
Index: 192000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE11E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE11F
Description: 11F
Directory: c:\rtelogs\log11F.log
Machine: RTRTE11
Parameter Set: PARAM2
Index: 195000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE11F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE12A
Description: 12A
Directory: c:\rtelogs\log12A.log
Machine: RTRTE12
Parameter Set: PARAM2
Index: 198000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE12A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE12B
Description: 12B
Directory: c:\rtelogs\log12B.log
Machine: RTRTE12
Parameter Set: PARAM2
Index: 201000000
Seed: 46329

Configured Users: 1000
Pipe Name: RTRTE12B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE12C
Description: 12C
Directory: c:\rtelogs\log12C.log
Machine: RTRTE12
Parameter Set: PARAM2
Index: 204000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE12C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE12D
Description: 12D
Directory: c:\rtelogs\log12D.log
Machine: RTRTE12
Parameter Set: PARAM2
Index: 207000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE12D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE12E
Description: 12E
Directory: c:\rtelogs\log12E.log
Machine: RTRTE12
Parameter Set: PARAM2
Index: 210000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE12E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE12F
Description: 12F
Directory: c:\rtelogs\log12F.log
Machine: RTRTE12
Parameter Set: PARAM2
Index: 213000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE12F670328
Connect Rate: 140
Start Rate: 0

Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE13A
Description: 13A
Directory: c:\rtelogs\log13A.log
Machine: RTRTE13
Parameter Set: PARAM2
Index: 216000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE13A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE13B
Description: 13B
Directory: c:\rtelogs\log13B.log
Machine: RTRTE13
Parameter Set: PARAM2
Index: 219000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE13B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE13C
Description: 13C
Directory: c:\rtelogs\log13C.log
Machine: RTRTE13
Parameter Set: PARAM2
Index: 222000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE13C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE13D
Description: 13D
Directory: c:\rtelogs\log13D.log
Machine: RTRTE13
Parameter Set: PARAM2
Index: 225000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE13D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE13E
Description: 13E
Directory: c:\rtelogs\log13E.log
Machine: RTRTE13
Parameter Set: PARAM2
Index: 228000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE13E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE13F
Description: 13F
Directory: c:\rtelogs\log13F.log
Machine: RTRTE13
Parameter Set: PARAM2
Index: 231000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE13F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE14A
Description: 14A
Directory: c:\rtelogs\log14A.log
Machine: RTRTE14
Parameter Set: PARAM2
Index: 234000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE14A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE14B
Description: 14B
Directory: c:\rtelogs\log14B.log
Machine: RTRTE14
Parameter Set: PARAM2
Index: 237000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE14B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE14C
Description: 14C
Directory: c:\rtelogs\log14C.log

Machine: RTRTE14
Parameter Set: PARAM2
Index: 240000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE14C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE14D
Description: 14D
Directory: c:\rtelogs\log14D.log
Machine: RTRTE14
Parameter Set: PARAM2
Index: 243000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE14D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE14E
Description: 14E
Directory: c:\rtelogs\log14E.log
Machine: RTRTE14
Parameter Set: PARAM2
Index: 246000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE14E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE14F
Description: 14F
Directory: c:\rtelogs\log14F.log
Machine: RTRTE14
Parameter Set: PARAM2
Index: 249000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE14F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE15A
Description: 15A
Directory: c:\rtelogs\log15A.log
Machine: RTRTE15
Parameter Set: PARAM2
Index: 252000000
Seed: 46329

Configured Users: 1000
Pipe Name: RTRTE15A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE15B
Description: 15B
Directory: c:\rtelogs\log15B.log
Machine: RTRTE15
Parameter Set: PARAM2
Index: 255000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE15B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE15C
Description: 15C
Directory: c:\rtelogs\log15C.log
Machine: RTRTE15
Parameter Set: PARAM2
Index: 258000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE15C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE15D
Description: 15D
Directory: c:\rtelogs\log15D.log
Machine: RTRTE15
Parameter Set: PARAM2
Index: 261000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE15D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE15E
Description: 15E
Directory: c:\rtelogs\log15E.log
Machine: RTRTE15
Parameter Set: PARAM2
Index: 264000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE15E670328
Connect Rate: 140
Start Rate: 0

Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE15F
Description: 15F
Directory: c:\rtelogs\log15F.log
Machine: RTRTE15
Parameter Set: PARAM2
Index: 267000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE15F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE16A
Description: 16A
Directory: c:\rtelogs\log16A.log
Machine: RTRTE16
Parameter Set: PARAM2
Index: 270000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE16A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE16B
Description: 16B
Directory: c:\rtelogs\log16B.log
Machine: RTRTE16
Parameter Set: PARAM2
Index: 273000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE16B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE16C
Description: 16C
Directory: c:\rtelogs\log16C.log
Machine: RTRTE16
Parameter Set: PARAM2
Index: 276000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE16C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE16D
Description: 16D
Directory: c:\rtelogs\log16D.log
Machine: RTRTE16
Parameter Set: PARAM2
Index: 279000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE16D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE16E
Description: 16E
Directory: c:\rtelogs\log16E.log
Machine: RTRTE16
Parameter Set: PARAM2
Index: 282000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE16E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE16F
Description: 16F
Directory: c:\rtelogs\log16F.log
Machine: RTRTE16
Parameter Set: PARAM2
Index: 285000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE16F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE17A
Description: 17A
Directory: c:\rtelogs\log17A.log
Machine: RTRTE17
Parameter Set: PARAM2
Index: 288000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE17A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE17B
Description: 17B
Directory: c:\rtelogs\log17B.log

Machine: RTRTE17
Parameter Set: PARAM2
Index: 291000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE17B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE17C
Description: 17C
Directory: c:\rtelogs\log17C.log
Machine: RTRTE17
Parameter Set: PARAM2
Index: 294000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE17C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE17D
Description: 17D
Directory: c:\rtelogs\log17D.log
Machine: RTRTE17
Parameter Set: PARAM2
Index: 297000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE17D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE17E
Description: 17E
Directory: c:\rtelogs\log17E.log
Machine: RTRTE17
Parameter Set: PARAM2
Index: 300000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE17E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE17F
Description: 17F
Directory: c:\rtelogs\log17F.log
Machine: RTRTE17
Parameter Set: PARAM2
Index: 303000000
Seed: 46329

Configured Users: 1000
Pipe Name: RTRTE17F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE18A
Description: 18A
Directory: c:\rtelogs\log18A.log
Machine: RTRTE18
Parameter Set: PARAM2
Index: 306000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE18A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE18B
Description: 18B
Directory: c:\rtelogs\log18B.log
Machine: RTRTE18
Parameter Set: PARAM2
Index: 309000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE18B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE18C
Description: 18C
Directory: c:\rtelogs\log18C.log
Machine: RTRTE18
Parameter Set: PARAM2
Index: 312000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE18C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE18D
Description: 18D
Directory: c:\rtelogs\log18D.log
Machine: RTRTE18
Parameter Set: PARAM2
Index: 315000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE18D670328
Connect Rate: 140
Start Rate: 0

Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE18E
Description: 18E
Directory: c:\rtelogs\log18E.log
Machine: RTRTE18
Parameter Set: PARAM2
Index: 318000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE18E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE18F
Description: 18F
Directory: c:\rtelogs\log18F.log
Machine: RTRTE18
Parameter Set: PARAM2
Index: 321000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE18F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE19A
Description: 19A
Directory: c:\rtelogs\log19A.log
Machine: RTRTE19
Parameter Set: PARAM2
Index: 324000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE19A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE19B
Description: 19B
Directory: c:\rtelogs\log19B.log
Machine: RTRTE19
Parameter Set: PARAM2
Index: 327000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE19B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE19C
Description: 19C
Directory: c:\rtelogs\log19C.log
Machine: RTRTE19
Parameter Set: PARAM2
Index: 330000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE19C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE19D
Description: 19D
Directory: c:\rtelogs\log19D.log
Machine: RTRTE19
Parameter Set: PARAM2
Index: 333000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE19D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE19E
Description: 19E
Directory: c:\rtelogs\log19E.log
Machine: RTRTE19
Parameter Set: PARAM2
Index: 336000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE19E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE19F
Description: 19F
Directory: c:\rtelogs\log19F.log
Machine: RTRTE19
Parameter Set: PARAM2
Index: 339000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE19F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE20A
Description: 20A
Directory: c:\rtelogs\log20A.log

Machine: RTRTE20
Parameter Set: PARAM2
Index: 342000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE20A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE20B
Description: 20B
Directory: c:\rtelogs\log20B.log
Machine: RTRTE20
Parameter Set: PARAM2
Index: 345000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE20B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE20C
Description: 20C
Directory: c:\rtelogs\log20C.log
Machine: RTRTE20
Parameter Set: PARAM2
Index: 348000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE20C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE20D
Description: 20D
Directory: c:\rtelogs\log20D.log
Machine: RTRTE20
Parameter Set: PARAM2
Index: 351000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE20D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE20E
Description: 20E
Directory: c:\rtelogs\log20E.log
Machine: RTRTE20
Parameter Set: PARAM2
Index: 354000000
Seed: 46329

Configured Users: 1000
Pipe Name: RTRTE20E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE20F
Description: 20F
Directory: c:\rtelogs\log20F.log
Machine: RTRTE20
Parameter Set: PARAM2
Index: 357000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE20F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE21A
Description: 21A
Directory: c:\rtelogs\log21A.log
Machine: RTRTE21
Parameter Set: PARAM2
Index: 360000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE21A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE21B
Description: 21B
Directory: c:\rtelogs\log21B.log
Machine: RTRTE21
Parameter Set: PARAM2
Index: 363000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE21B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE21C
Description: 21C
Directory: c:\rtelogs\log21C.log
Machine: RTRTE21
Parameter Set: PARAM2
Index: 366000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE21C670328
Connect Rate: 140
Start Rate: 0

Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE21D
Description: 21D
Directory: c:\rtelogs\log21D.log
Machine: RTRTE21
Parameter Set: PARAM2
Index: 369000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE21D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE21E
Description: 21E
Directory: c:\rtelogs\log21E.log
Machine: RTRTE21
Parameter Set: PARAM2
Index: 372000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE21E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE21F
Description: 21F
Directory: c:\rtelogs\log21F.log
Machine: RTRTE21
Parameter Set: PARAM2
Index: 375000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE21F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE22A
Description: 22A
Directory: c:\rtelogs\log22A.log
Machine: RTRTE22
Parameter Set: PARAM2
Index: 378000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE22A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE22B
Description: 22B
Directory: c:\rtelogs\log22B.log
Machine: RTRTE22
Parameter Set: PARAM2
Index: 381000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE22B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE22C
Description: 22C
Directory: c:\rtelogs\log22C.log
Machine: RTRTE22
Parameter Set: PARAM2
Index: 384000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE22C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE22D
Description: 22D
Directory: c:\rtelogs\log22D.log
Machine: RTRTE22
Parameter Set: PARAM2
Index: 387000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE22D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE22E
Description: 22E
Directory: c:\rtelogs\log22E.log
Machine: RTRTE22
Parameter Set: PARAM2
Index: 390000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE22E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE22F
Description: 22F
Directory: c:\rtelogs\log22F.log

Machine: RTRTE22
Parameter Set: PARAM2
Index: 393000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE22F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE23A
Description: 23A
Directory: c:\rtelogs\log23A.log
Machine: RTRTE23
Parameter Set: PARAM2
Index: 396000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE23A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE23B
Description: 23B
Directory: c:\rtelogs\log23B.log
Machine: RTRTE23
Parameter Set: PARAM2
Index: 399000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE23B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE23C
Description: 23C
Directory: c:\rtelogs\log23C.log
Machine: RTRTE23
Parameter Set: PARAM2
Index: 402000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE23C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE23D
Description: 23D
Directory: c:\rtelogs\log23D.log
Machine: RTRTE23
Parameter Set: PARAM2
Index: 405000000
Seed: 46329

Configured Users: 1000
Pipe Name: RTRTE23D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE23E
Description: 23E
Directory: c:\rtelogs\log23E.log
Machine: RTRTE23
Parameter Set: PARAM2
Index: 408000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE23E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE23F
Description: 23F
Directory: c:\rtelogs\log23F.log
Machine: RTRTE23
Parameter Set: PARAM2
Index: 411000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE23F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE24A
Description: 24A
Directory: c:\rtelogs\log24A.log
Machine: RTRTE24
Parameter Set: PARAM2
Index: 414000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE24A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE24B
Description: 24B
Directory: c:\rtelogs\log24B.log
Machine: RTRTE24
Parameter Set: PARAM2
Index: 417000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE24B670328
Connect Rate: 140
Start Rate: 0

Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE24C
Description: 24C
Directory: c:\rtelogs\log24C.log
Machine: RTRTE24
Parameter Set: PARAM2
Index: 420000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE24C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE24D
Description: 24D
Directory: c:\rtelogs\log24D.log
Machine: RTRTE24
Parameter Set: PARAM2
Index: 423000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE24D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE24E
Description: 24E
Directory: c:\rtelogs\log24E.log
Machine: RTRTE24
Parameter Set: PARAM2
Index: 426000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE24E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE24F
Description: 24F
Directory: c:\rtelogs\log24F.log
Machine: RTRTE24
Parameter Set: PARAM2
Index: 429000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE24F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE25A
Description: 25A
Directory: c:\rtelogs\log25A.log
Machine: RTRTE25
Parameter Set: PARAM2
Index: 432000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE25A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE25B
Description: 25B
Directory: c:\rtelogs\log25B.log
Machine: RTRTE25
Parameter Set: PARAM2
Index: 435000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE25B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE25C
Description: 25C
Directory: c:\rtelogs\log25C.log
Machine: RTRTE25
Parameter Set: PARAM2
Index: 438000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE25C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE25D
Description: 25D
Directory: c:\rtelogs\log25D.log
Machine: RTRTE25
Parameter Set: PARAM2
Index: 441000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE25D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE25E
Description: 25E
Directory: c:\rtelogs\log25E.log

Machine: RTRTE25
Parameter Set: PARAM2
Index: 444000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE25E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE25F
Description: 25F
Directory: c:\rtelogs\log25F.log
Machine: RTRTE25
Parameter Set: PARAM2
Index: 447000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE25F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE26A
Description: 26A
Directory: c:\rtelogs\log26A.log
Machine: RTRTE26
Parameter Set: PARAM2
Index: 450000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE26A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE26B
Description: 26B
Directory: c:\rtelogs\log26B.log
Machine: RTRTE26
Parameter Set: PARAM2
Index: 453000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE26B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE26C
Description: 26C
Directory: c:\rtelogs\log26C.log
Machine: RTRTE26
Parameter Set: PARAM2
Index: 456000000
Seed: 46329

Configured Users: 1000
Pipe Name: RTRTE26C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE26D
Description: 26D
Directory: c:\rtelogs\log26D.log
Machine: RTRTE26
Parameter Set: PARAM2
Index: 459000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE26D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE26E
Description: 26E
Directory: c:\rtelogs\log26E.log
Machine: RTRTE26
Parameter Set: PARAM2
Index: 462000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE26E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE26F
Description: 26F
Directory: c:\rtelogs\log26F.log
Machine: RTRTE26
Parameter Set: PARAM2
Index: 465000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE26F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE27A
Description: 27A
Directory: c:\rtelogs\log27A.log
Machine: RTRTE27
Parameter Set: PARAM2
Index: 468000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE27A670328
Connect Rate: 140
Start Rate: 0

Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE27B
Description: 27B
Directory: c:\rtelogs\log27B.log
Machine: RTRTE27
Parameter Set: PARAM2
Index: 471000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE27B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE27C
Description: 27C
Directory: c:\rtelogs\log27C.log
Machine: RTRTE27
Parameter Set: PARAM2
Index: 474000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE27C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE27D
Description: 27D
Directory: c:\rtelogs\log27D.log
Machine: RTRTE27
Parameter Set: PARAM2
Index: 477000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE27D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE27E
Description: 27E
Directory: c:\rtelogs\log27E.log
Machine: RTRTE27
Parameter Set: PARAM2
Index: 480000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE27E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE27F
Description: 27F
Directory: c:\rtelogs\log27F.log
Machine: RTRTE27
Parameter Set: PARAM2
Index: 483000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE27F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE28A
Description: 28A
Directory: c:\rtelogs\log28A.log
Machine: RTRTE28
Parameter Set: PARAM2
Index: 486000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE28A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE28B
Description: 28B
Directory: c:\rtelogs\log28B.log
Machine: RTRTE28
Parameter Set: PARAM2
Index: 489000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE28B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE28C
Description: 28C
Directory: c:\rtelogs\log28C.log
Machine: RTRTE28
Parameter Set: PARAM2
Index: 492000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE28C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE28D
Description: 28D
Directory: c:\rtelogs\log28D.log

Machine: RTRTE28
Parameter Set: PARAM2
Index: 495000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE28D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE28E
Description: 28E
Directory: c:\rtelogs\log28E.log
Machine: RTRTE28
Parameter Set: PARAM2
Index: 498000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE28E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE28F
Description: 28F
Directory: c:\rtelogs\log28F.log
Machine: RTRTE28
Parameter Set: PARAM2
Index: 501000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE28F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE29A
Description: 29A
Directory: c:\rtelogs\log29A.log
Machine: RTRTE29
Parameter Set: PARAM2
Index: 504000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE29A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE29B
Description: 29B
Directory: c:\rtelogs\log29B.log
Machine: RTRTE29
Parameter Set: PARAM2
Index: 507000000
Seed: 46329

Configured Users: 1000
Pipe Name: RTRTE29B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE29C
Description: 29C
Directory: c:\rtelogs\log29C.log
Machine: RTRTE29
Parameter Set: PARAM2
Index: 510000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE29C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE29D
Description: 29D
Directory: c:\rtelogs\log29D.log
Machine: RTRTE29
Parameter Set: PARAM2
Index: 513000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE29D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE29E
Description: 29E
Directory: c:\rtelogs\log29E.log
Machine: RTRTE29
Parameter Set: PARAM2
Index: 516000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE29E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE29F
Description: 29F
Directory: c:\rtelogs\log29F.log
Machine: RTRTE29
Parameter Set: PARAM2
Index: 519000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE29F670328
Connect Rate: 140
Start Rate: 0

Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE30A
Description: 30A
Directory: c:\rtelogs\log30A.log
Machine: RTRTE30
Parameter Set: PARAM2
Index: 522000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE30A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE30B
Description: 30B
Directory: c:\rtelogs\log30B.log
Machine: RTRTE30
Parameter Set: PARAM2
Index: 525000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE30B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE30C
Description: 30C
Directory: c:\rtelogs\log30C.log
Machine: RTRTE30
Parameter Set: PARAM2
Index: 528000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE30C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE30D
Description: 30D
Directory: c:\rtelogs\log30D.log
Machine: RTRTE30
Parameter Set: PARAM2
Index: 531000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE30D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE30E
Description: 30E
Directory: c:\rtelogs\log30E.log
Machine: RTRTE30
Parameter Set: PARAM2
Index: 534000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE30E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE30F
Description: 30F
Directory: c:\rtelogs\log30F.log
Machine: RTRTE30
Parameter Set: PARAM2
Index: 537000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE30F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE31A
Description: 31A
Directory: c:\rtelogs\log31A.log
Machine: RTRTE31
Parameter Set: PARAM2
Index: 540000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE31A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE31B
Description: 31B
Directory: c:\rtelogs\log31B.log
Machine: RTRTE31
Parameter Set: PARAM2
Index: 543000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE31B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE31C
Description: 31C
Directory: c:\rtelogs\log31C.log

Machine: RTRTE31
Parameter Set: PARAM2
Index: 546000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE31C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE31D
Description: 31D
Directory: c:\rtelogs\log31D.log
Machine: RTRTE31
Parameter Set: PARAM2
Index: 549000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE31D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE31E
Description: 31E
Directory: c:\rtelogs\log31E.log
Machine: RTRTE31
Parameter Set: PARAM2
Index: 552000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE31E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE31F
Description: 31F
Directory: c:\rtelogs\log31F.log
Machine: RTRTE31
Parameter Set: PARAM2
Index: 555000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE31F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE32A
Description: 32A
Directory: c:\rtelogs\log32A.log
Machine: RTRTE32
Parameter Set: PARAM2
Index: 558000000
Seed: 46329

Configured Users: 1000
Pipe Name: RTRTE32A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE32B
Description: 32B
Directory: c:\rtelogs\log32B.log
Machine: RTRTE32
Parameter Set: PARAM2
Index: 561000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE32B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE32C
Description: 32C
Directory: c:\rtelogs\log32C.log
Machine: RTRTE32
Parameter Set: PARAM2
Index: 564000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE32C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE32D
Description: 32D
Directory: c:\rtelogs\log32D.log
Machine: RTRTE32
Parameter Set: PARAM2
Index: 567000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE32D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE32E
Description: 32E
Directory: c:\rtelogs\log32E.log
Machine: RTRTE32
Parameter Set: PARAM2
Index: 570000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE32E670328
Connect Rate: 140
Start Rate: 0

Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE32F
Description: 32F
Directory: c:\rtelogs\log32F.log
Machine: RTRTE32
Parameter Set: PARAM2
Index: 573000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE32F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE33A
Description: 33A
Directory: c:\rtelogs\log33A.log
Machine: RTRTE33
Parameter Set: PARAM2
Index: 576000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE33A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE33B
Description: 33B
Directory: c:\rtelogs\log33B.log
Machine: RTRTE33
Parameter Set: PARAM2
Index: 579000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE33B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE33C
Description: 33C
Directory: c:\rtelogs\log33C.log
Machine: RTRTE33
Parameter Set: PARAM2
Index: 582000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE33C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE33D
Description: 33D
Directory: c:\rtelogs\log33D.log
Machine: RTRTE33
Parameter Set: PARAM2
Index: 585000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE33D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE33E
Description: 33E
Directory: c:\rtelogs\log33E.log
Machine: RTRTE33
Parameter Set: PARAM2
Index: 588000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE33E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE33F
Description: 33F
Directory: c:\rtelogs\log33F.log
Machine: RTRTE33
Parameter Set: PARAM2
Index: 591000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE33F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE34A
Description: 34A
Directory: c:\rtelogs\log34A.log
Machine: RTRTE34
Parameter Set: PARAM2
Index: 594000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE34A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE34B
Description: 34B
Directory: c:\rtelogs\log34B.log

Machine: RTRTE34
Parameter Set: PARAM2
Index: 597000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE34B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE34C
Description: 34C
Directory: c:\rtelogs\log34C.log
Machine: RTRTE34
Parameter Set: PARAM2
Index: 600000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE34C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE34D
Description: 34D
Directory: c:\rtelogs\log34D.log
Machine: RTRTE34
Parameter Set: PARAM2
Index: 603000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE34D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE34E
Description: 34E
Directory: c:\rtelogs\log34E.log
Machine: RTRTE34
Parameter Set: PARAM2
Index: 606000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE34E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE34F
Description: 34F
Directory: c:\rtelogs\log34F.log
Machine: RTRTE34
Parameter Set: PARAM2
Index: 609000000
Seed: 46329

Configured Users: 1000
Pipe Name: RTRTE34F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE35A
Description: 35A
Directory: c:\rtelogs\log35A.log
Machine: RTRTE35
Parameter Set: PARAM2
Index: 612000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE35A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE35B
Description: 35B
Directory: c:\rtelogs\log35B.log
Machine: RTRTE35
Parameter Set: PARAM2
Index: 615000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE35B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE35C
Description: 35C
Directory: c:\rtelogs\log35C.log
Machine: RTRTE35
Parameter Set: PARAM2
Index: 618000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE35C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE35D
Description: 35D
Directory: c:\rtelogs\log35D.log
Machine: RTRTE35
Parameter Set: PARAM2
Index: 621000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE35D670328
Connect Rate: 140
Start Rate: 0

Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE35E
Description: 35E
Directory: c:\rtelogs\log35E.log
Machine: RTRTE35
Parameter Set: PARAM2
Index: 624000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE35E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE35F
Description: 35F
Directory: c:\rtelogs\log35F.log
Machine: RTRTE35
Parameter Set: PARAM2
Index: 627000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE35F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE36A
Description: 36A
Directory: c:\rtelogs\log36A.log
Machine: RTRTE36
Parameter Set: PARAM2
Index: 630000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE36A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE36B
Description: 36B
Directory: c:\rtelogs\log36B.log
Machine: RTRTE36
Parameter Set: PARAM2
Index: 633000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE36B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE36C
Description: 36C
Directory: c:\rtelogs\log36C.log
Machine: RTRTE36
Parameter Set: PARAM2
Index: 636000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE36C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE36D
Description: 36D
Directory: c:\rtelogs\log36D.log
Machine: RTRTE36
Parameter Set: PARAM2
Index: 639000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE36D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE36E
Description: 36E
Directory: c:\rtelogs\log36E.log
Machine: RTRTE36
Parameter Set: PARAM2
Index: 642000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE36E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE36F
Description: 36F
Directory: c:\rtelogs\log36F.log
Machine: RTRTE36
Parameter Set: PARAM2
Index: 645000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE36F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE37A
Description: 37A
Directory: c:\rtelogs\log37A.log

Machine: RTRTE37
Parameter Set: PARAM2
Index: 648000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE37A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE37B
Description: 37B
Directory: c:\rtelogs\log37B.log
Machine: RTRTE37
Parameter Set: PARAM2
Index: 651000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE37B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE37C
Description: 37C
Directory: c:\rtelogs\log37C.log
Machine: RTRTE37
Parameter Set: PARAM2
Index: 654000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE37C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE37D
Description: 37D
Directory: c:\rtelogs\log37D.log
Machine: RTRTE37
Parameter Set: PARAM2
Index: 657000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE37D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE37E
Description: 37E
Directory: c:\rtelogs\log37E.log
Machine: RTRTE37
Parameter Set: PARAM2
Index: 660000000
Seed: 46329

Configured Users: 1000
Pipe Name: RTRTE37E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE37F
Description: 37F
Directory: c:\rtelogs\log37F.log
Machine: RTRTE37
Parameter Set: PARAM2
Index: 663000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE37F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE38A
Description: 38A
Directory: c:\rtelogs\log38A.log
Machine: RTRTE38
Parameter Set: PARAM2
Index: 666000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE38A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE38B
Description: 38B
Directory: c:\rtelogs\log38B.log
Machine: RTRTE38
Parameter Set: PARAM2
Index: 669000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE38B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE38C
Description: 38C
Directory: c:\rtelogs\log38C.log
Machine: RTRTE38
Parameter Set: PARAM2
Index: 672000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE38C670328
Connect Rate: 140
Start Rate: 0

Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE38D
Description: 38D
Directory: c:\rtelogs\log38D.log
Machine: RTRTE38
Parameter Set: PARAM2
Index: 675000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE38D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE38E
Description: 38E
Directory: c:\rtelogs\log38E.log
Machine: RTRTE38
Parameter Set: PARAM2
Index: 678000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE38E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE38F
Description: 38F
Directory: c:\rtelogs\log38F.log
Machine: RTRTE38
Parameter Set: PARAM2
Index: 681000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE38F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE39A
Description: 39A
Directory: c:\rtelogs\log39A.log
Machine: RTRTE39
Parameter Set: PARAM2
Index: 684000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE39A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE39B
Description: 39B
Directory: c:\rtelogs\log39B.log
Machine: RTRTE39
Parameter Set: PARAM2
Index: 687000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE39B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE39C
Description: 39C
Directory: c:\rtelogs\log39C.log
Machine: RTRTE39
Parameter Set: PARAM2
Index: 690000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE39C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE39D
Description: 39D
Directory: c:\rtelogs\log39D.log
Machine: RTRTE39
Parameter Set: PARAM2
Index: 693000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE39D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE39E
Description: 39E
Directory: c:\rtelogs\log39E.log
Machine: RTRTE39
Parameter Set: PARAM2
Index: 696000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE39E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE39F
Description: 39F
Directory: c:\rtelogs\log39F.log

Machine: RTRTE39
Parameter Set: PARAM2
Index: 699000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE39F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE40A
Description: 40A
Directory: c:\rtelogs\log40A.log
Machine: RTRTE40
Parameter Set: PARAM2
Index: 702000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE40A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE40B
Description: 40B
Directory: c:\rtelogs\log40B.log
Machine: RTRTE40
Parameter Set: PARAM2
Index: 705000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE40B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE40C
Description: 40C
Directory: c:\rtelogs\log40C.log
Machine: RTRTE40
Parameter Set: PARAM2
Index: 708000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE40C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE40D
Description: 40D
Directory: c:\rtelogs\log40D.log
Machine: RTRTE40
Parameter Set: PARAM2
Index: 711000000
Seed: 46329

Configured Users: 1000
Pipe Name: RTRTE40D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE40E
Description: 40E
Directory: c:\rtelogs\log40E.log
Machine: RTRTE40
Parameter Set: PARAM2
Index: 714000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE40E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE40F
Description: 40F
Directory: c:\rtelogs\log40F.log
Machine: RTRTE40
Parameter Set: PARAM2
Index: 717000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE40F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE41A
Description: 41A
Directory: c:\rtelogs\log41A.log
Machine: RTRTE41
Parameter Set: PARAM2
Index: 720000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE41A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE41B
Description: 41B
Directory: c:\rtelogs\log41B.log
Machine: RTRTE41
Parameter Set: PARAM2
Index: 723000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE41B670328
Connect Rate: 140
Start Rate: 0

Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE41C
Description: 41C
Directory: c:\rtelogs\log41C.log
Machine: RTRTE41
Parameter Set: PARAM2
Index: 726000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE41C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE41D
Description: 41D
Directory: c:\rtelogs\log41D.log
Machine: RTRTE41
Parameter Set: PARAM2
Index: 729000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE41D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE41E
Description: 41E
Directory: c:\rtelogs\log41E.log
Machine: RTRTE41
Parameter Set: PARAM2
Index: 732000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE41E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE41F
Description: 41F
Directory: c:\rtelogs\log41F.log
Machine: RTRTE41
Parameter Set: PARAM2
Index: 735000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE41F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE42A
Description: 42A
Directory: c:\rtelogs\log42A.log
Machine: RTRTE42
Parameter Set: PARAM2
Index: 738000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE42A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE42B
Description: 42B
Directory: c:\rtelogs\log42B.log
Machine: RTRTE42
Parameter Set: PARAM2
Index: 741000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE42B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE42C
Description: 42C
Directory: c:\rtelogs\log42C.log
Machine: RTRTE42
Parameter Set: PARAM2
Index: 744000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE42C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE42D
Description: 42D
Directory: c:\rtelogs\log42D.log
Machine: RTRTE42
Parameter Set: PARAM2
Index: 747000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE42D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE42E
Description: 42E
Directory: c:\rtelogs\log42E.log

Machine: RTRTE42
Parameter Set: PARAM2
Index: 750000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE42E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE42F
Description: 42F
Directory: c:\rtelogs\log42F.log
Machine: RTRTE42
Parameter Set: PARAM2
Index: 753000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE42F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE43A
Description: 43A
Directory: c:\rtelogs\log43A.log
Machine: RTRTE43
Parameter Set: PARAM2
Index: 756000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE43A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE43B
Description: 43B
Directory: c:\rtelogs\log43B.log
Machine: RTRTE43
Parameter Set: PARAM2
Index: 759000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE43B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE43C
Description: 43C
Directory: c:\rtelogs\log43C.log
Machine: RTRTE43
Parameter Set: PARAM2
Index: 762000000
Seed: 46329

Configured Users: 1000
Pipe Name: RTRTE43C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE43D
Description: 43D
Directory: c:\rtelogs\log43D.log
Machine: RTRTE43
Parameter Set: PARAM2
Index: 765000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE43D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE43E
Description: 43E
Directory: c:\rtelogs\log43E.log
Machine: RTRTE43
Parameter Set: PARAM2
Index: 768000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE43E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE43F
Description: 43F
Directory: c:\rtelogs\log43F.log
Machine: RTRTE43
Parameter Set: PARAM2
Index: 771000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE43F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE44A
Description: 44A
Directory: c:\rtelogs\log44A.log
Machine: RTRTE44
Parameter Set: PARAM2
Index: 774000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE44A670328
Connect Rate: 140
Start Rate: 0

Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE44B
Description: 44B
Directory: c:\rtelogs\log44B.log
Machine: RTRTE44
Parameter Set: PARAM2
Index: 777000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE44B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE44C
Description: 44C
Directory: c:\rtelogs\log44C.log
Machine: RTRTE44
Parameter Set: PARAM2
Index: 780000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE44C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE44D
Description: 44D
Directory: c:\rtelogs\log44D.log
Machine: RTRTE44
Parameter Set: PARAM2
Index: 783000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE44D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE44E
Description: 44E
Directory: c:\rtelogs\log44E.log
Machine: RTRTE44
Parameter Set: PARAM2
Index: 786000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE44E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE44F
Description: 44F
Directory: c:\rtelogs\log44F.log
Machine: RTRTE44
Parameter Set: PARAM2
Index: 789000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE44F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE45A
Description: 45A
Directory: c:\rtelogs\log45A.log
Machine: RTRTE45
Parameter Set: PARAM2
Index: 792000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE45A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE45B
Description: 45B
Directory: c:\rtelogs\log45B.log
Machine: RTRTE45
Parameter Set: PARAM2
Index: 795000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE45B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE45C
Description: 45C
Directory: c:\rtelogs\log45C.log
Machine: RTRTE45
Parameter Set: PARAM2
Index: 798000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE45C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE45D
Description: 45D
Directory: c:\rtelogs\log45D.log

Machine: RTRTE45
Parameter Set: PARAM2
Index: 801000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE45D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE45E
Description: 45E
Directory: c:\rtelogs\log45E.log
Machine: RTRTE45
Parameter Set: PARAM2
Index: 804000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE45E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE45F
Description: 45F
Directory: c:\rtelogs\log45F.log
Machine: RTRTE45
Parameter Set: PARAM2
Index: 807000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE45F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE46A
Description: 46A
Directory: c:\rtelogs\log46A.log
Machine: RTRTE46
Parameter Set: PARAM2
Index: 810000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE46A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE46B
Description: 46B
Directory: c:\rtelogs\log46B.log
Machine: RTRTE46
Parameter Set: PARAM2
Index: 813000000
Seed: 46329

Configured Users: 1000
Pipe Name: RTRTE46B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE46C
Description: 46C
Directory: c:\rtelogs\log46C.log
Machine: RTRTE46
Parameter Set: PARAM2
Index: 816000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE46C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE46D
Description: 46D
Directory: c:\rtelogs\log46D.log
Machine: RTRTE46
Parameter Set: PARAM2
Index: 819000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE46D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE46E
Description: 46E
Directory: c:\rtelogs\log46E.log
Machine: RTRTE46
Parameter Set: PARAM2
Index: 822000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE46E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE46F
Description: 46F
Directory: c:\rtelogs\log46F.log
Machine: RTRTE46
Parameter Set: PARAM2
Index: 825000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE46F670328
Connect Rate: 140
Start Rate: 0

Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE47A
Description: 47A
Directory: c:\rtelogs\log47A.log
Machine: RTRTE47
Parameter Set: PARAM2
Index: 828000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE47A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE47B
Description: 47B
Directory: c:\rtelogs\log47B.log
Machine: RTRTE47
Parameter Set: PARAM2
Index: 831000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE47B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE47C
Description: 47C
Directory: c:\rtelogs\log47C.log
Machine: RTRTE47
Parameter Set: PARAM2
Index: 834000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE47C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE47D
Description: 47D
Directory: c:\rtelogs\log47D.log
Machine: RTRTE47
Parameter Set: PARAM2
Index: 837000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE47D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE47E
Description: 47E
Directory: c:\rtelogs\log47E.log
Machine: RTRTE47
Parameter Set: PARAM2
Index: 840000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE47E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE47F
Description: 47F
Directory: c:\rtelogs\log47F.log
Machine: RTRTE47
Parameter Set: PARAM2
Index: 843000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE47F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE48A
Description: 48A
Directory: c:\rtelogs\log48A.log
Machine: RTRTE48
Parameter Set: PARAM2
Index: 846000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE48A670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE48B
Description: 48B
Directory: c:\rtelogs\log48B.log
Machine: RTRTE48
Parameter Set: PARAM2
Index: 849000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE48B670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE48C
Description: 48C
Directory: c:\rtelogs\log48C.log

Machine: RTRTE48
Parameter Set: PARAM2
Index: 852000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE48C670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE48D
Description: 48D
Directory: c:\rtelogs\log48D.log
Machine: RTRTE48
Parameter Set: PARAM2
Index: 855000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE48D670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Name: RTRTE48E
Description: 48E
Directory: c:\rtelogs\log48E.log
Machine: RTRTE48
Parameter Set: PARAM2
Index: 858000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE48E670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0

Name: RTRTE48F
Description: 48F
Directory: c:\rtelogs\log48F.log
Machine: RTRTE48
Parameter Set: PARAM2
Index: 861000000
Seed: 46329
Configured Users: 1000
Pipe Name: RTRTE48F670328
Connect Rate: 140
Start Rate: 0
Max. Concurrency: 1000
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 1

Number of User groups: 288

Driver Engine: RTRTE01A
IIS Server: RTCLIENT01A
SQL Server: RTNODE01CL
Database: tpcc
User: sa

Protocol: HTML
w_id Range: 1 - 100
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE01B
IIS Server: RTCLIENT01B
SQL Server: RTNODE01CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 101 - 200
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE01C
IIS Server: RTCLIENT01C
SQL Server: RTNODE01CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 201 - 300
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE01D
IIS Server: RTCLIENT01D
SQL Server: RTNODE01CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 301 - 400
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE01E
IIS Server: RTCLIENT01E
SQL Server: RTNODE01CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 401 - 500
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE01F
IIS Server: RTCLIENT01F
SQL Server: RTNODE01CL

Database: tpcc
User: sa
Protocol: HTML
w_id Range: 501 - 600
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE02A
IIS Server: RTCLIENT02A
SQL Server: RTNODE01CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 601 - 700
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE02B
IIS Server: RTCLIENT02B
SQL Server: RTNODE01CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 701 - 800
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE02C
IIS Server: RTCLIENT02C
SQL Server: RTNODE01CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 801 - 900
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE02D
IIS Server: RTCLIENT02D
SQL Server: RTNODE01CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 901 - 1000
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE02E

IIS Server: RTCLIENT02E
SQL Server: RTNODE01CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 1001 - 1100
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE02F
IIS Server: RTCLIENT02F
SQL Server: RTNODE01CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 1101 - 1200
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE03A
IIS Server: RTCLIENT03A
SQL Server: RTNODE01CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 1201 - 1300
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE03B
IIS Server: RTCLIENT03B
SQL Server: RTNODE01CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 1301 - 1400
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE03C
IIS Server: RTCLIENT03C
SQL Server: RTNODE01CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 1401 - 1500
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE03D
IIS Server: RTCLIENT03D
SQL Server: RTNODE01CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 1501 - 1600
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE03E
IIS Server: RTCLIENT03E
SQL Server: RTNODE01CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 1601 - 1700
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE03F
IIS Server: RTCLIENT03F
SQL Server: RTNODE01CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 1701 - 1800
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE04A
IIS Server: RTCLIENT04A
SQL Server: RTNODE02CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 1801 - 1900
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE04B
IIS Server: RTCLIENT04B
SQL Server: RTNODE02CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 1901 - 2000
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000

District id: 1
Scale Down: No

Driver Engine: RTRTE04C
IIS Server: RTCLIENT04C
SQL Server: RTNODE02CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 2001 - 2100
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE04D
IIS Server: RTCLIENT04D
SQL Server: RTNODE02CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 2101 - 2200
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE04E
IIS Server: RTCLIENT04E
SQL Server: RTNODE02CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 2201 - 2300
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE04F
IIS Server: RTCLIENT04F
SQL Server: RTNODE02CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 2301 - 2400
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE05A
IIS Server: RTCLIENT05A
SQL Server: RTNODE02CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 2401 - 2500
w_id Min Warehouse: 1
w_id Max Warehouse: 28800

Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE05B
IIS Server: RTCLIENT05B
SQL Server: RTNODE02CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 2501 - 2600
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE05C
IIS Server: RTCLIENT05C
SQL Server: RTNODE02CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 2601 - 2700
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE05D
IIS Server: RTCLIENT05D
SQL Server: RTNODE02CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 2701 - 2800
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE05E
IIS Server: RTCLIENT05E
SQL Server: RTNODE02CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 2801 - 2900
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE05F
IIS Server: RTCLIENT05F
SQL Server: RTNODE02CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 2901 - 3000

w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE06A
IIS Server: RTCLIENT06A
SQL Server: RTNODE02CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 3001 - 3100
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE06B
IIS Server: RTCLIENT06B
SQL Server: RTNODE02CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 3101 - 3200
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE06C
IIS Server: RTCLIENT06C
SQL Server: RTNODE02CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 3201 - 3300
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE06D
IIS Server: RTCLIENT06D
SQL Server: RTNODE02CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 3301 - 3400
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE06E
IIS Server: RTCLIENT06E
SQL Server: RTNODE02CL
Database: tpcc
User: sa

Protocol: HTML
w_id Range: 3401 - 3500
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE06F
IIS Server: RTCLIENT06F
SQL Server: RTNODE02CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 3501 - 3600
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE07A
IIS Server: RTCLIENT07A
SQL Server: RTNODE03CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 3601 - 3700
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE07B
IIS Server: RTCLIENT07B
SQL Server: RTNODE03CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 3701 - 3800
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE07C
IIS Server: RTCLIENT07C
SQL Server: RTNODE03CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 3801 - 3900
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE07D
IIS Server: RTCLIENT07D
SQL Server: RTNODE03CL

Database: tpcc
User: sa
Protocol: HTML
w_id Range: 3901 - 4000
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE07E
IIS Server: RTCLIENT07E
SQL Server: RTNODE03CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 4001 - 4100
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE07F
IIS Server: RTCLIENT07F
SQL Server: RTNODE03CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 4101 - 4200
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE08A
IIS Server: RTCLIENT08A
SQL Server: RTNODE03CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 4201 - 4300
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE08B
IIS Server: RTCLIENT08B
SQL Server: RTNODE03CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 4301 - 4400
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE08C

IIS Server: RTCLIENT08C
SQL Server: RTNODE03CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 4401 - 4500
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE08D
IIS Server: RTCLIENT08D
SQL Server: RTNODE03CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 4501 - 4600
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE08E
IIS Server: RTCLIENT08E
SQL Server: RTNODE03CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 4601 - 4700
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE08F
IIS Server: RTCLIENT08F
SQL Server: RTNODE03CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 4701 - 4800
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE09A
IIS Server: RTCLIENT09A
SQL Server: RTNODE03CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 4801 - 4900
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE09B
IIS Server: RTCLIENT09B
SQL Server: RTNODE03CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 4901 - 5000
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE09C
IIS Server: RTCLIENT09C
SQL Server: RTNODE03CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 5001 - 5100
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE09D
IIS Server: RTCLIENT09D
SQL Server: RTNODE03CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 5101 - 5200
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE09E
IIS Server: RTCLIENT09E
SQL Server: RTNODE03CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 5201 - 5300
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE09F
IIS Server: RTCLIENT09F
SQL Server: RTNODE03CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 5301 - 5400
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000

District id: 1
Scale Down: No

Driver Engine: RTRTE10A
IIS Server: RTCLIENT10A
SQL Server: RTNODE04CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 5401 - 5500
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE10B
IIS Server: RTCLIENT10B
SQL Server: RTNODE04CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 5501 - 5600
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE10C
IIS Server: RTCLIENT10C
SQL Server: RTNODE04CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 5601 - 5700
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE10D
IIS Server: RTCLIENT10D
SQL Server: RTNODE04CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 5701 - 5800
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE10E
IIS Server: RTCLIENT10E
SQL Server: RTNODE04CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 5801 - 5900
w_id Min Warehouse: 1
w_id Max Warehouse: 28800

Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE10F
IIS Server: RTCLIENT10F
SQL Server: RTNODE04CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 5901 - 6000
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE11A
IIS Server: RTCLIENT11A
SQL Server: RTNODE04CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 6001 - 6100
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE11B
IIS Server: RTCLIENT11B
SQL Server: RTNODE04CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 6101 - 6200
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE11C
IIS Server: RTCLIENT11C
SQL Server: RTNODE04CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 6201 - 6300
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE11D
IIS Server: RTCLIENT11D
SQL Server: RTNODE04CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 6301 - 6400

w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE11E
IIS Server: RTCLIENT11E
SQL Server: RTNODE04CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 6401 - 6500
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE11F
IIS Server: RTCLIENT11F
SQL Server: RTNODE04CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 6501 - 6600
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE12A
IIS Server: RTCLIENT12A
SQL Server: RTNODE04CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 6601 - 6700
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE12B
IIS Server: RTCLIENT12B
SQL Server: RTNODE04CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 6701 - 6800
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE12C
IIS Server: RTCLIENT12C
SQL Server: RTNODE04CL
Database: tpcc
User: sa

Protocol: HTML
w_id Range: 6801 - 6900
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE12D
IIS Server: RTCLIENT12D
SQL Server: RTNODE04CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 6901 - 7000
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE12E
IIS Server: RTCLIENT12E
SQL Server: RTNODE04CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 7001 - 7100
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE12F
IIS Server: RTCLIENT12F
SQL Server: RTNODE04CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 7101 - 7200
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE13A
IIS Server: RTCLIENT13A
SQL Server: RTNODE05CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 7201 - 7300
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE13B
IIS Server: RTCLIENT13B
SQL Server: RTNODE05CL

Database: tpcc
User: sa
Protocol: HTML
w_id Range: 7301 - 7400
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE13C
IIS Server: RTCLIENT13C
SQL Server: RTNODE05CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 7401 - 7500
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE13D
IIS Server: RTCLIENT13D
SQL Server: RTNODE05CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 7501 - 7600
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE13E
IIS Server: RTCLIENT13E
SQL Server: RTNODE05CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 7601 - 7700
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE13F
IIS Server: RTCLIENT13F
SQL Server: RTNODE05CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 7701 - 7800
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE14A

IIS Server: RTCLIENT14A
SQL Server: RTNODE05CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 7801 - 7900
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE14B
IIS Server: RTCLIENT14B
SQL Server: RTNODE05CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 7901 - 8000
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE14C
IIS Server: RTCLIENT14C
SQL Server: RTNODE05CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 8001 - 8100
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE14D
IIS Server: RTCLIENT14D
SQL Server: RTNODE05CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 8101 - 8200
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE14E
IIS Server: RTCLIENT14E
SQL Server: RTNODE05CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 8201 - 8300
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE14F
IIS Server: RTCLIENT14F
SQL Server: RTNODE05CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 8301 - 8400
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE15A
IIS Server: RTCLIENT15A
SQL Server: RTNODE05CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 8401 - 8500
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE15B
IIS Server: RTCLIENT15B
SQL Server: RTNODE05CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 8501 - 8600
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE15C
IIS Server: RTCLIENT15C
SQL Server: RTNODE05CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 8601 - 8700
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE15D
IIS Server: RTCLIENT15D
SQL Server: RTNODE05CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 8701 - 8800
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000

District id: 1
Scale Down: No

Driver Engine: RTRTE15E
IIS Server: RTCLIENT15E
SQL Server: RTNODE05CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 8801 - 8900
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE15F
IIS Server: RTCLIENT15F
SQL Server: RTNODE05CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 8901 - 9000
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE16A
IIS Server: RTCLIENT16A
SQL Server: RTNODE06CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 9001 - 9100
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE16B
IIS Server: RTCLIENT16B
SQL Server: RTNODE06CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 9101 - 9200
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE16C
IIS Server: RTCLIENT16C
SQL Server: RTNODE06CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 9201 - 9300
w_id Min Warehouse: 1
w_id Max Warehouse: 28800

Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE16D
IIS Server: RTCLIENT16D
SQL Server: RTNODE06CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 9301 - 9400
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE16E
IIS Server: RTCLIENT16E
SQL Server: RTNODE06CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 9401 - 9500
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE16F
IIS Server: RTCLIENT16F
SQL Server: RTNODE06CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 9501 - 9600
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE17A
IIS Server: RTCLIENT17A
SQL Server: RTNODE06CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 9601 - 9700
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE17B
IIS Server: RTCLIENT17B
SQL Server: RTNODE06CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 9701 - 9800

w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE17C
IIS Server: RTCLIENT17C
SQL Server: RTNODE06CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 9801 - 9900
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE17D
IIS Server: RTCLIENT17D
SQL Server: RTNODE06CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 9901 - 10000
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE17E
IIS Server: RTCLIENT17E
SQL Server: RTNODE06CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 10001 - 10100
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE17F
IIS Server: RTCLIENT17F
SQL Server: RTNODE06CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 10101 - 10200
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE18A
IIS Server: RTCLIENT18A
SQL Server: RTNODE06CL
Database: tpcc
User: sa

Protocol: HTML
w_id Range: 10201 - 10300
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE18B
IIS Server: RTCLIENT18B
SQL Server: RTNODE06CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 10301 - 10400
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE18C
IIS Server: RTCLIENT18C
SQL Server: RTNODE06CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 10401 - 10500
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE18D
IIS Server: RTCLIENT18D
SQL Server: RTNODE06CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 10501 - 10600
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE18E
IIS Server: RTCLIENT18E
SQL Server: RTNODE06CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 10601 - 10700
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE18F
IIS Server: RTCLIENT18F
SQL Server: RTNODE06CL

Database: tpcc
User: sa
Protocol: HTML
w_id Range: 10701 - 10800
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE19A
IIS Server: RTCLIENT19A
SQL Server: RTNODE07CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 10801 - 10900
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE19B
IIS Server: RTCLIENT19B
SQL Server: RTNODE07CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 10901 - 11000
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE19C
IIS Server: RTCLIENT19C
SQL Server: RTNODE07CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 11001 - 11100
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE19D
IIS Server: RTCLIENT19D
SQL Server: RTNODE07CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 11101 - 11200
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE19E

IIS Server: RTCLIENT19E
SQL Server: RTNODE07CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 11201 - 11300
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE19F
IIS Server: RTCLIENT19F
SQL Server: RTNODE07CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 11301 - 11400
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE20A
IIS Server: RTCLIENT20A
SQL Server: RTNODE07CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 11401 - 11500
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE20B
IIS Server: RTCLIENT20B
SQL Server: RTNODE07CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 11501 - 11600
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE20C
IIS Server: RTCLIENT20C
SQL Server: RTNODE07CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 11601 - 11700
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE20D
IIS Server: RTCLIENT20D
SQL Server: RTNODE07CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 11701 - 11800
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE20E
IIS Server: RTCLIENT20E
SQL Server: RTNODE07CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 11801 - 11900
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE20F
IIS Server: RTCLIENT20F
SQL Server: RTNODE07CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 11901 - 12000
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE21A
IIS Server: RTCLIENT21A
SQL Server: RTNODE07CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 12001 - 12100
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE21B
IIS Server: RTCLIENT21B
SQL Server: RTNODE07CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 12101 - 12200
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000

District id: 1
Scale Down: No

Driver Engine: RTRTE21C
IIS Server: RTCLIENT21C
SQL Server: RTNODE07CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 12201 - 12300
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE21D
IIS Server: RTCLIENT21D
SQL Server: RTNODE07CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 12301 - 12400
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE21E
IIS Server: RTCLIENT21E
SQL Server: RTNODE07CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 12401 - 12500
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE21F
IIS Server: RTCLIENT21F
SQL Server: RTNODE07CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 12501 - 12600
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE22A
IIS Server: RTCLIENT22A
SQL Server: RTNODE08CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 12601 - 12700
w_id Min Warehouse: 1
w_id Max Warehouse: 28800

Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE22B
IIS Server: RTCLIENT22B
SQL Server: RTNODE08CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 12701 - 12800
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE22C
IIS Server: RTCLIENT22C
SQL Server: RTNODE08CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 12801 - 12900
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE22D
IIS Server: RTCLIENT22D
SQL Server: RTNODE08CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 12901 - 13000
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE22E
IIS Server: RTCLIENT22E
SQL Server: RTNODE08CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 13001 - 13100
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE22F
IIS Server: RTCLIENT22F
SQL Server: RTNODE08CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 13101 - 13200

w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE23A
IIS Server: RTCLIENT23A
SQL Server: RTNODE08CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 13201 - 13300
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE23B
IIS Server: RTCLIENT23B
SQL Server: RTNODE08CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 13301 - 13400
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE23C
IIS Server: RTCLIENT23C
SQL Server: RTNODE08CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 13401 - 13500
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE23D
IIS Server: RTCLIENT23D
SQL Server: RTNODE08CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 13501 - 13600
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE23E
IIS Server: RTCLIENT23E
SQL Server: RTNODE08CL
Database: tpcc
User: sa

Protocol: HTML
w_id Range: 13601 - 13700
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE23F
IIS Server: RTCLIENT23F
SQL Server: RTNODE08CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 13701 - 13800
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE24A
IIS Server: RTCLIENT24A
SQL Server: RTNODE08CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 13801 - 13900
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE24B
IIS Server: RTCLIENT24B
SQL Server: RTNODE08CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 13901 - 14000
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE24C
IIS Server: RTCLIENT24C
SQL Server: RTNODE08CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 14001 - 14100
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE24D
IIS Server: RTCLIENT24D
SQL Server: RTNODE08CL

Database: tpcc
User: sa
Protocol: HTML
w_id Range: 14101 - 14200
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE24E
IIS Server: RTCLIENT24E
SQL Server: RTNODE08CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 14201 - 14300
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE24F
IIS Server: RTCLIENT24F
SQL Server: RTNODE08CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 14301 - 14400
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE25A
IIS Server: RTCLIENT25A
SQL Server: RTNODE09CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 14401 - 14500
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE25B
IIS Server: RTCLIENT25B
SQL Server: RTNODE09CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 14501 - 14600
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE25C

IIS Server: RTCLIENT25C
SQL Server: RTNODE09CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 14601 - 14700
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE25D
IIS Server: RTCLIENT25D
SQL Server: RTNODE09CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 14701 - 14800
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE25E
IIS Server: RTCLIENT25E
SQL Server: RTNODE09CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 14801 - 14900
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE25F
IIS Server: RTCLIENT25F
SQL Server: RTNODE09CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 14901 - 15000
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE26A
IIS Server: RTCLIENT26A
SQL Server: RTNODE09CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 15001 - 15100
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE26B
IIS Server: RTCLIENT26B
SQL Server: RTNODE09CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 15101 - 15200
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE26C
IIS Server: RTCLIENT26C
SQL Server: RTNODE09CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 15201 - 15300
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE26D
IIS Server: RTCLIENT26D
SQL Server: RTNODE09CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 15301 - 15400
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE26E
IIS Server: RTCLIENT26E
SQL Server: RTNODE09CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 15401 - 15500
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE26F
IIS Server: RTCLIENT26F
SQL Server: RTNODE09CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 15501 - 15600
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000

District id: 1
Scale Down: No

Driver Engine: RTRTE27A
IIS Server: RTCLIENT27A
SQL Server: RTNODE09CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 15601 - 15700
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE27B
IIS Server: RTCLIENT27B
SQL Server: RTNODE09CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 15701 - 15800
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE27C
IIS Server: RTCLIENT27C
SQL Server: RTNODE09CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 15801 - 15900
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE27D
IIS Server: RTCLIENT27D
SQL Server: RTNODE09CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 15901 - 16000
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE27E
IIS Server: RTCLIENT27E
SQL Server: RTNODE09CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 16001 - 16100
w_id Min Warehouse: 1
w_id Max Warehouse: 28800

Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE27F
IIS Server: RTCLIENT27F
SQL Server: RTNODE09CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 16101 - 16200
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE28A
IIS Server: RTCLIENT28A
SQL Server: RTNODE10CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 16201 - 16300
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE28B
IIS Server: RTCLIENT28B
SQL Server: RTNODE10CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 16301 - 16400
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE28C
IIS Server: RTCLIENT28C
SQL Server: RTNODE10CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 16401 - 16500
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE28D
IIS Server: RTCLIENT28D
SQL Server: RTNODE10CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 16501 - 16600

w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE28E
IIS Server: RTCLIENT28E
SQL Server: RTNODE10CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 16601 - 16700
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE28F
IIS Server: RTCLIENT28F
SQL Server: RTNODE10CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 16701 - 16800
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE29A
IIS Server: RTCLIENT29A
SQL Server: RTNODE10CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 16801 - 16900
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE29B
IIS Server: RTCLIENT29B
SQL Server: RTNODE10CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 16901 - 17000
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE29C
IIS Server: RTCLIENT29C
SQL Server: RTNODE10CL
Database: tpcc
User: sa

Protocol: HTML
w_id Range: 17001 - 17100
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE29D
IIS Server: RTCLIENT29D
SQL Server: RTNODE10CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 17101 - 17200
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE29E
IIS Server: RTCLIENT29E
SQL Server: RTNODE10CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 17201 - 17300
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE29F
IIS Server: RTCLIENT29F
SQL Server: RTNODE10CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 17301 - 17400
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE30A
IIS Server: RTCLIENT30A
SQL Server: RTNODE10CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 17401 - 17500
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE30B
IIS Server: RTCLIENT30B
SQL Server: RTNODE10CL

Database: tpcc
User: sa
Protocol: HTML
w_id Range: 17501 - 17600
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE30C
IIS Server: RTCLIENT30C
SQL Server: RTNODE10CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 17601 - 17700
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE30D
IIS Server: RTCLIENT30D
SQL Server: RTNODE10CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 17701 - 17800
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE30E
IIS Server: RTCLIENT30E
SQL Server: RTNODE10CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 17801 - 17900
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE30F
IIS Server: RTCLIENT30F
SQL Server: RTNODE10CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 17901 - 18000
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE31A

IIS Server: RTCLIENT31A
SQL Server: RTNODE11CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 18001 - 18100
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE31B
IIS Server: RTCLIENT31B
SQL Server: RTNODE11CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 18101 - 18200
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE31C
IIS Server: RTCLIENT31C
SQL Server: RTNODE11CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 18201 - 18300
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE31D
IIS Server: RTCLIENT31D
SQL Server: RTNODE11CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 18301 - 18400
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE31E
IIS Server: RTCLIENT31E
SQL Server: RTNODE11CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 18401 - 18500
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE31F
IIS Server: RTCLIENT31F
SQL Server: RTNODE11CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 18501 - 18600
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE32A
IIS Server: RTCLIENT32A
SQL Server: RTNODE11CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 18601 - 18700
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE32B
IIS Server: RTCLIENT32B
SQL Server: RTNODE11CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 18701 - 18800
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE32C
IIS Server: RTCLIENT32C
SQL Server: RTNODE11CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 18801 - 18900
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE32D
IIS Server: RTCLIENT32D
SQL Server: RTNODE11CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 18901 - 19000
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000

District id: 1
Scale Down: No

Driver Engine: RTRTE32E
IIS Server: RTCLIENT32E
SQL Server: RTNODE11CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 19001 - 19100
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE32F
IIS Server: RTCLIENT32F
SQL Server: RTNODE11CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 19101 - 19200
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE33A
IIS Server: RTCLIENT33A
SQL Server: RTNODE11CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 19201 - 19300
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE33B
IIS Server: RTCLIENT33B
SQL Server: RTNODE11CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 19301 - 19400
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE33C
IIS Server: RTCLIENT33C
SQL Server: RTNODE11CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 19401 - 19500
w_id Min Warehouse: 1
w_id Max Warehouse: 28800

Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE33D
IIS Server: RTCLIENT33D
SQL Server: RTNODE11CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 19501 - 19600
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE33E
IIS Server: RTCLIENT33E
SQL Server: RTNODE11CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 19601 - 19700
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE33F
IIS Server: RTCLIENT33F
SQL Server: RTNODE11CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 19701 - 19800
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE34A
IIS Server: RTCLIENT34A
SQL Server: RTNODE12CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 19801 - 19900
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE34B
IIS Server: RTCLIENT34B
SQL Server: RTNODE12CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 19901 - 20000

w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE34C
IIS Server: RTCLIENT34C
SQL Server: RTNODE12CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 20001 - 20100
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE34D
IIS Server: RTCLIENT34D
SQL Server: RTNODE12CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 20101 - 20200
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE34E
IIS Server: RTCLIENT34E
SQL Server: RTNODE12CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 20201 - 20300
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE34F
IIS Server: RTCLIENT34F
SQL Server: RTNODE12CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 20301 - 20400
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE35A
IIS Server: RTCLIENT35A
SQL Server: RTNODE12CL
Database: tpcc
User: sa

Protocol: HTML
w_id Range: 20401 - 20500
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE35B
IIS Server: RTCLIENT35B
SQL Server: RTNODE12CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 20501 - 20600
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE35C
IIS Server: RTCLIENT35C
SQL Server: RTNODE12CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 20601 - 20700
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE35D
IIS Server: RTCLIENT35D
SQL Server: RTNODE12CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 20701 - 20800
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE35E
IIS Server: RTCLIENT35E
SQL Server: RTNODE12CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 20801 - 20900
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE35F
IIS Server: RTCLIENT35F
SQL Server: RTNODE12CL

Database: tpcc
User: sa
Protocol: HTML
w_id Range: 20901 - 21000
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE36A
IIS Server: RTCLIENT36A
SQL Server: RTNODE12CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 21001 - 21100
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE36B
IIS Server: RTCLIENT36B
SQL Server: RTNODE12CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 21101 - 21200
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE36C
IIS Server: RTCLIENT36C
SQL Server: RTNODE12CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 21201 - 21300
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE36D
IIS Server: RTCLIENT36D
SQL Server: RTNODE12CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 21301 - 21400
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE36E

IIS Server: RTCLIENT36E
SQL Server: RTNODE12CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 21401 - 21500
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE36F
IIS Server: RTCLIENT36F
SQL Server: RTNODE12CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 21501 - 21600
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE37A
IIS Server: RTCLIENT37A
SQL Server: RTNODE13CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 21601 - 21700
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE37B
IIS Server: RTCLIENT37B
SQL Server: RTNODE13CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 21701 - 21800
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE37C
IIS Server: RTCLIENT37C
SQL Server: RTNODE13CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 21801 - 21900
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE37D
IIS Server: RTCLIENT37D
SQL Server: RTNODE13CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 21901 - 22000
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE37E
IIS Server: RTCLIENT37E
SQL Server: RTNODE13CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 22001 - 22100
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE37F
IIS Server: RTCLIENT37F
SQL Server: RTNODE13CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 22101 - 22200
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE38A
IIS Server: RTCLIENT38A
SQL Server: RTNODE13CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 22201 - 22300
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE38B
IIS Server: RTCLIENT38B
SQL Server: RTNODE13CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 22301 - 22400
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000

District id: 1
Scale Down: No

Driver Engine: RTRTE38C
IIS Server: RTCLIENT38C
SQL Server: RTNODE13CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 22401 - 22500
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE38D
IIS Server: RTCLIENT38D
SQL Server: RTNODE13CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 22501 - 22600
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE38E
IIS Server: RTCLIENT38E
SQL Server: RTNODE13CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 22601 - 22700
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE38F
IIS Server: RTCLIENT38F
SQL Server: RTNODE13CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 22701 - 22800
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE39A
IIS Server: RTCLIENT39A
SQL Server: RTNODE13CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 22801 - 22900
w_id Min Warehouse: 1
w_id Max Warehouse: 28800

Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE39B
IIS Server: RTCLIENT39B
SQL Server: RTNODE13CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 22901 - 23000
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE39C
IIS Server: RTCLIENT39C
SQL Server: RTNODE13CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 23001 - 23100
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE39D
IIS Server: RTCLIENT39D
SQL Server: RTNODE13CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 23101 - 23200
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE39E
IIS Server: RTCLIENT39E
SQL Server: RTNODE13CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 23201 - 23300
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE39F
IIS Server: RTCLIENT39F
SQL Server: RTNODE13CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 23301 - 23400

w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE40A
IIS Server: RTCLIENT40A
SQL Server: RTNODE14CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 23401 - 23500
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE40B
IIS Server: RTCLIENT40B
SQL Server: RTNODE14CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 23501 - 23600
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE40C
IIS Server: RTCLIENT40C
SQL Server: RTNODE14CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 23601 - 23700
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE40D
IIS Server: RTCLIENT40D
SQL Server: RTNODE14CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 23701 - 23800
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE40E
IIS Server: RTCLIENT40E
SQL Server: RTNODE14CL
Database: tpcc
User: sa

Protocol: HTML
w_id Range: 23801 - 23900
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE40F
IIS Server: RTCLIENT40F
SQL Server: RTNODE14CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 23901 - 24000
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE41A
IIS Server: RTCLIENT41A
SQL Server: RTNODE14CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 24001 - 24100
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE41B
IIS Server: RTCLIENT41B
SQL Server: RTNODE14CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 24101 - 24200
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE41C
IIS Server: RTCLIENT41C
SQL Server: RTNODE14CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 24201 - 24300
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE41D
IIS Server: RTCLIENT41D
SQL Server: RTNODE14CL

Database: tpcc
User: sa
Protocol: HTML
w_id Range: 24301 - 24400
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE41E
IIS Server: RTCLIENT41E
SQL Server: RTNODE14CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 24401 - 24500
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE41F
IIS Server: RTCLIENT41F
SQL Server: RTNODE14CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 24501 - 24600
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE42A
IIS Server: RTCLIENT42A
SQL Server: RTNODE14CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 24601 - 24700
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE42B
IIS Server: RTCLIENT42B
SQL Server: RTNODE14CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 24701 - 24800
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE42C

IIS Server: RTCLIENT42C
SQL Server: RTNODE14CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 24801 - 24900
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE42D
IIS Server: RTCLIENT42D
SQL Server: RTNODE14CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 24901 - 25000
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE42E
IIS Server: RTCLIENT42E
SQL Server: RTNODE14CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 25001 - 25100
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE42F
IIS Server: RTCLIENT42F
SQL Server: RTNODE14CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 25101 - 25200
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE43A
IIS Server: RTCLIENT43A
SQL Server: RTNODE15CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 25201 - 25300
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE43B
IIS Server: RTCLIENT43B
SQL Server: RTNODE15CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 25301 - 25400
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE43C
IIS Server: RTCLIENT43C
SQL Server: RTNODE15CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 25401 - 25500
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE43D
IIS Server: RTCLIENT43D
SQL Server: RTNODE15CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 25501 - 25600
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE43E
IIS Server: RTCLIENT43E
SQL Server: RTNODE15CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 25601 - 25700
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE43F
IIS Server: RTCLIENT43F
SQL Server: RTNODE15CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 25701 - 25800
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000

District id: 1
Scale Down: No

Driver Engine: RTRTE44A
IIS Server: RTCLIENT44A
SQL Server: RTNODE15CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 25801 - 25900
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE44B
IIS Server: RTCLIENT44B
SQL Server: RTNODE15CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 25901 - 26000
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE44C
IIS Server: RTCLIENT44C
SQL Server: RTNODE15CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 26001 - 26100
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE44D
IIS Server: RTCLIENT44D
SQL Server: RTNODE15CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 26101 - 26200
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE44E
IIS Server: RTCLIENT44E
SQL Server: RTNODE15CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 26201 - 26300
w_id Min Warehouse: 1
w_id Max Warehouse: 28800

Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE44F
IIS Server: RTCLIENT44F
SQL Server: RTNODE15CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 26301 - 26400
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE45A
IIS Server: RTCLIENT45A
SQL Server: RTNODE15CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 26401 - 26500
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE45B
IIS Server: RTCLIENT45B
SQL Server: RTNODE15CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 26501 - 26600
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE45C
IIS Server: RTCLIENT45C
SQL Server: RTNODE15CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 26601 - 26700
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE45D
IIS Server: RTCLIENT45D
SQL Server: RTNODE15CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 26701 - 26800

w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE45E
IIS Server: RTCLIENT45E
SQL Server: RTNODE15CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 26801 - 26900
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE45F
IIS Server: RTCLIENT45F
SQL Server: RTNODE15CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 26901 - 27000
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE46A
IIS Server: RTCLIENT46A
SQL Server: RTNODE16CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 27001 - 27100
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE46B
IIS Server: RTCLIENT46B
SQL Server: RTNODE16CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 27101 - 27200
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE46C
IIS Server: RTCLIENT46C
SQL Server: RTNODE16CL
Database: tpcc
User: sa

Protocol: HTML
w_id Range: 27201 - 27300
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE46D
IIS Server: RTCLIENT46D
SQL Server: RTNODE16CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 27301 - 27400
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE46E
IIS Server: RTCLIENT46E
SQL Server: RTNODE16CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 27401 - 27500
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE46F
IIS Server: RTCLIENT46F
SQL Server: RTNODE16CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 27501 - 27600
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE47A
IIS Server: RTCLIENT47A
SQL Server: RTNODE16CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 27601 - 27700
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE47B
IIS Server: RTCLIENT47B
SQL Server: RTNODE16CL

Database: tpcc
User: sa
Protocol: HTML
w_id Range: 27701 - 27800
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE47C
IIS Server: RTCLIENT47C
SQL Server: RTNODE16CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 27801 - 27900
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE47D
IIS Server: RTCLIENT47D
SQL Server: RTNODE16CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 27901 - 28000
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE47E
IIS Server: RTCLIENT47E
SQL Server: RTNODE16CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 28001 - 28100
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE47F
IIS Server: RTCLIENT47F
SQL Server: RTNODE16CL
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 28101 - 28200
w_id Min Warehouse: 1
w_id Max Warehouse: 28800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Driver Engine: RTRTE48A

IIS Server: RTCLIENT48A
 SQL Server: RTNODE16CL
 Database: tpcc
 User: sa
 Protocol: HTML
 w_id Range: 28201 - 28300
 w_id Min Warehouse: 1
 w_id Max Warehouse: 28800
 Scale: Normal
 User Count: 1000
 District id: 1
 Scale Down: No

Driver Engine: RTRTE48B
 IIS Server: RTCLIENT48B
 SQL Server: RTNODE16CL
 Database: tpcc
 User: sa
 Protocol: HTML
 w_id Range: 28301 - 28400
 w_id Min Warehouse: 1
 w_id Max Warehouse: 28800
 Scale: Normal
 User Count: 1000
 District id: 1
 Scale Down: No

Driver Engine: RTRTE48C
 IIS Server: RTCLIENT48C
 SQL Server: RTNODE16CL
 Database: tpcc
 User: sa
 Protocol: HTML
 w_id Range: 28401 - 28500
 w_id Min Warehouse: 1
 w_id Max Warehouse: 28800
 Scale: Normal
 User Count: 1000
 District id: 1
 Scale Down: No

Driver Engine: RTRTE48D
 IIS Server: RTCLIENT48D
 SQL Server: RTNODE16CL
 Database: tpcc
 User: sa
 Protocol: HTML
 w_id Range: 28501 - 28600
 w_id Min Warehouse: 1
 w_id Max Warehouse: 28800
 Scale: Normal
 User Count: 1000
 District id: 1
 Scale Down: No

Driver Engine: RTRTE48E
 IIS Server: RTCLIENT48E
 SQL Server: RTNODE16CL
 Database: tpcc
 User: sa
 Protocol: HTML
 w_id Range: 28601 - 28700
 w_id Min Warehouse: 1
 w_id Max Warehouse: 28800
 Scale: Normal
 User Count: 1000
 District id: 1
 Scale Down: No

Driver Engine: RTRTE48F
 IIS Server: RTCLIENT48F
 SQL Server: RTNODE16CL
 Database: tpcc
 User: sa
 Protocol: HTML
 w_id Range: 28701 - 28800
 w_id Min Warehouse: 1
 w_id Max Warehouse: 28800
 Scale: Normal
 User Count: 1000
 District id: 1
 Scale Down: No

Number of Parameter Sets: 2

		~Default				
		Default Parameter Set				
		Txn	Think	Key	RT	RT
Menu	Delay	Weight	Time	Time	Delay	Fence
		New Order	10.00		12.05	
18.01	0.10	5.00	0.10			
		Payment	10.00	12.05	3.01	
0.10	5.00	0.10				
		Delivery	1.00	5.05	2.01	
0.10	5.00	0.10				
		Stock Level	1.00	5.05	2.01	
0.10	20.00	0.10				
		Order Status	1.00	10.05	2.01	
0.10	5.00	0.10				
		PARAM2				
		Editable Default Parameter Set				
		Txn	Think	Key	RT	RT
Menu	Delay	Weight	Time	Time	Delay	Fence
		New Order	10.00		12.05	
18.01	0.10	5.00	0.10			
		Payment	9.61	12.05	3.01	
0.10	5.00	0.10				
		Delivery	0.90	5.05	2.01	
0.10	5.00	0.10				
		Stock Level	0.90	5.05	2.01	
0.10	20.00	0.10				
		Order Status	0.90	10.05	2.01	
0.10	5.00	0.10				

Appendix D: 180-Day Space

TPC-C 180-Day Space Requirements						
Warehouses	28,800	tpmC	363,129.75	tpmC/W	12.61	
Numbers are shown in Kbytes unless otherwise specified.						
Table	Rows	Data	Index	5% Space	8H Space	Total Space
Warehouse	28,800	3,200	1,536	236.80		4,972.80
District	288,000	32,512	1,896	1,720.40		36,128.40
Item	100,000	304,896	2,176	7,731.20		162,355.20
New-Order	259,200,000	152,448	12,288		2,304,000	6,934,656.00
History	864,000,000	4,618,368	169,600		10,440,345.06	62,192,153.06
Orders	864,000,000	51,582,208	13,765,464		8,468,552.13	50,446,368.13
Customer	864,000,000	28,212,352	39,217,672	33,379,066.00		700,960,386.00
Order-Line	8,639,860,511	628,363,648	1,363,792		116,474,787.24	693,829,347.24
Stock	2,880,000,000	575,990,768	1,952,256	46,177,612.80		969,729,868.80
Totals		921,600,000	56,486,680	79,566,367.20	137,687,684.42	2,484,296,236
Segment	LogDev Cnt.	Segment Size	Needed	Overhead		Not Needed
misc	224	917,504,000	813,605,981	8,136,060		95,761,959.37
big	224	1,949,696,000	1,670,690,255	16,706,903		262,298,842.65
master, msdb,model	16	212,992	212,992			0.00
tpcc_root	16	131,072	131,072			0.00
tempdb	16	139,264	139,264			0.00
Totals		2,867,683,328.00	2,484,779,563.62	24,842,962.36		358,060,802.02
Dynamic Space	655,785,328.00	Sum of Data for Order, Order-Line and History				
Static Space	1,715,666,185.56	Data + Index + 5% Space + Overhead - Dynamic Space				
Free Space	138,171,012.42	Total Segment Size - Dynamic Space - Static Space - Not needed				
Daily Growth	132,297,312.34	(Dynamic Space/Wc * 62.5)* tpmC				
Daily Spread	(60,274,956.08)	Free Space - 1.5 * Daily Growth (Zero If Negative)				
180-Day Space (KB)	25,529,182,406.59	Static Space + 180 (Daily Growth + Daily Spread)				
180-Day Space (GB)	24,346.53	180-Day Space in GB (Excludes OS,Paging and RDBMS Logs)				
Available (GB)	28,474.68	Total storage configured and available for database, minus logs, in a RAID-1E configuration				
Log File Storage Requirements						
Log Used (KB)	119,236,208.46	Total				
% Log Used	11.9305	% Log File Storage Used During Entire Run				
Total N-O Txn	23,129,903.00	Total Count of New-Order Transactions Completed during Entire Run				
Log per N-O Txn	5.16	KB of Log Storage Used per New-Order Transaction				
8 Hour Log (GB)	856.91	8-Hours of Log in GB (excluding space for redundancy)				
Log Configured (GB)	953.13	59.57GB per Node (RAID-10)				
Disk Capacity	MB	GB				
9.1GB 10K rpm	8,678	8.47				
18.2GB 10K rpm	17,356	16.95				
Space Usage	GB Needed	Disks Priced		GB Priced	GB Usable	TB Usable
180-Day (RAID-1)	24,346.53	0	9.1GB 10K rpm	0.00	0.00	
		3,360	18.2GB 10K rpm	56,949.38	28,474.69	
Total DB		3,360		56,949.38	28,474.69	
8hr Log (RAID-1)	856.91	128	9.1GB 10K rpm	2,169.50	1,084.75	
OS, SQL (RAID-1)	128.00	32	18.2GB 10K rpm	271.19	135.59	
Total Space	25,331.44	3,520		59,390.06	29,695.03	57.99

Appendix E: Third-Party Quotations

Microsoft Corporation
One Microsoft Way
Redmond, WA 98052-6399

Tel 425 882 8080
Fax 425 936 7329
<http://www.microsoft.com/>

Microsoft

March 20, 2001

IBM
Tricia Thomas
D23U/B060/Office E127
3039 Cornwallis Road
Research Triangle Park, NC 27709

Tricia:

Here is the information you requested regarding pricing for several Microsoft products to be used in conjunction with your TPC-C V3.5 benchmark testing.

All pricing shown is in US Dollars (\$).

Part Number	Description	Unit Price	Quantity	Price
810-00945	SQL Server 2000 Enterprise Edition <i>Per processor licensing</i> <i>Discount schedule: Select B discount plan</i>	\$ 15,802	128	\$ 2,022,656
C11-00821	Windows 2000 Server <i>Server license only - No CALs</i> <i>Discount schedule: Open Program - No Level</i>	\$ 738	1	\$ 738
C10-00475	Windows 2000 Advanced Server <i>Server license only - No CALs</i> <i>Discount schedule: Open Program - No Level</i>	\$ 2,399	1	\$ 2,399
048-00317	Visual C++ Professional 6.0 Win32	\$ 549	1	\$ 549
	5-year maintenance for above software	\$ 2,095	1	\$ 167,600

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.

Reference ID: Ptpmf0120036478

Please include this Reference ID in any correspondence regarding this price quote.

Quote

Giganet, Inc.
580 Main Street
Bolton, MA 01740
(978) 779-7200
(978) 7797-203 Fax

Quote No.: 000000-00
Date: March 1, 2001

Ship to: IBM Corp.
Bill to: Chris King
3039 Corwallis Road
Dept. 23UA Bldg. 060/D133
Research Triangle Park, NC 27709

F.O.B.	Terms
Bolton, MA	Net 30

Qty	Part Number/ Description	Unit Price	Ext. Price
19	cLAN-1000 Giganet Host Adapter Card	\$795.00	\$15,105.00
19	cLAN-A1011 10 Meter Copper Cable	\$135.00	\$2,565.00
2	cLAN5300 30 port Cluster Switch	\$15,995.00	\$31,990.00
	5-Year Maintenance Price (7x24x4)	\$15,000.00	\$75,000.00
	3-Year Maintenance Price (7x24x4)	\$15,000.00	\$45,000.00
	Hardware TOTAL		\$49,660.00
	Actual shipping charges will be billed		

Prices are valid for 90 days from the date of this quote.

If you have any questions regarding this quote, please contact the GigaNet Sales Department at (978) 779-7400.



Cisco Systems, Inc.
 2300 Rexwoods Drive
 Suite 300
 Raleigh, NC 27607 USA
 Ph: Ph: 919-788-1208
 Fax: Fax: 919-788-1299

Price Quotation

Date: 3/19/01
To: Joe Jakubowski
 IBM

Quote Number: 4Z2-RPT
Total Price: \$56,152.50

Ph: (919) 543-6693
 Fax: (919) 486-2327

Product Number	Product Description	Qty	Unit List Price	Disc Price	Disc %	Extended Price
	Catalyst 6500 Test Lab Switch					
WS-C6506	Catalyst 6506 Chassis	1	\$7,995.00		25.000%	\$5,996.25
WS-CAC-1300W	Catalyst 6000 1300W AC Power Supply	1	\$3,995.00		25.000%	\$2,996.25
WS-CAC-1300W/2	Catalyst 6000 Second 1300W AC Power Supply	1	\$3,995.00		25.000%	\$2,996.25
CAB-7513AC	AC POWER CORD NORTH AMERICA	2	\$0.00		25.000%	\$0.00
SFC6K-SUP-5.3.5	Catalyst 6000 Supervisor Flash Image, Release 5.3(5)	1	\$0.00		25.000%	\$0.00
WS-X6K-SUP1A-PFC	Catalyst 6000 Supervisor Engine1-A, 2GE, plus PFC	1	\$17,495.00		25.000%	\$13,121.25
MEM-C6K-FLC16M	Catalyst 6000 Supervisor PCMCIA Flash Mem Card, 16MB Option	1	\$400.00		25.000%	\$300.00
WS-X6348-RJ-45	Catalyst 6000 48-port 10/100, Upgradable to Voice, Enh QoS	2	\$12,995.00		25.000%	\$19,492.50
WS-G5484	1000BASE-SX "Short Wavelength" GBIC (Multimode only)	2	\$500.00		25.000%	\$750.00
	Maintenance and support extended price is calculated for 5 years.					
CON-OSP-WS-C6506	24x7x4 OS Service,Catalyst 6506	1	\$10,500.00		0.000%	\$10,500.00

FOB Point: Origin
Ship Date:
Quote Valid Until: 06/19/2001

Payment Terms: Net 30
Installation: Available on Request and Billable
Warranty: 90 days

Signed:

Bob Blum

Notes:

Netfinity Lab Equipment

This price quotation does not constitute an offer by Cisco to sell products, but is instead an invitation to issue a purchase order to Cisco until the Quotation Valid date specified on this Price Quotation. Such a purchase order will be subject to Cisco's standard procedures, terms, and conditions for the acceptance of purchase orders. This order may be subject to sales tax, VAT, duty and freight charges even if not noted on this quote.

Software House International
2880 Zanker Blvd. #103
San Jose, CA 95134
Matthew Martin
National Account Executive
1-800-766-6357

Description	Part Number	Unit Price	Qty	Extended Price
8-Port 10/100Mbps Hub	DEH2924	23	39,600	\$910,800

Prices are valid for 90 days from March 16, 2001.