

TPC Benchmark™ C
Full Disclosure Report
for
IBM® @server® xSeries® 460
using
DB2® Universal Database 8.2
and
Microsoft® Windows® Server 2003
Enterprise x64 Edition

TPC-C Version 5.3

Submitted for Review
May 31, 2005



First Edition - May 2005

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 2005. All rights reserved.

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

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

Trademarks

IBM, the IBM logo, DB2, xSeries, ServeRAID, eServer and the eServer logo are trademarks or registered trademarks of International Business Machines Corporation.

The following terms used in this publication are trademarks of other companies as follows: TPC Benchmark, tpmC, and \$/tpmC trademark of Transaction Processing Performance Council; Intel and Xeon are trademarks or registered trademarks of Intel Corporation; Microsoft and Windows 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 GHz and MHz only measures microprocessor internal clock speed, not application performance. Many factors affect application performance.

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

Abstract

IBM Corporation conducted the TPC Benchmark™ C on the IBM® @server® xSeries® 460 configured as a client/server system. This report documents the full disclosure information required by the TPC Benchmark C Standard Specification, Revision 5.3, including the methodology used to achieve the reported results. All testing fully complied with this revision level.

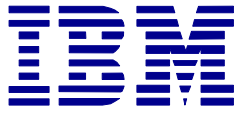
The software used on the xSeries 460 system includes Microsoft® Windows® Server 2003 Enterprise x64 Edition operating system and IBM DB2 Universal Database V8.2 Enterprise Edition database.

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

The benchmark results are summarized in the following table.

Hardware	Software	Total System Cost	tpmC	\$/tpmC	Total Solution Availability Date
IBM @server xSeries 460	DB2 UDB 8.2 Microsoft Windows Server 2003 Enterprise x64 Edition	\$1,440,290 USD	250,975	\$5.74 USD	November 30, 2005

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

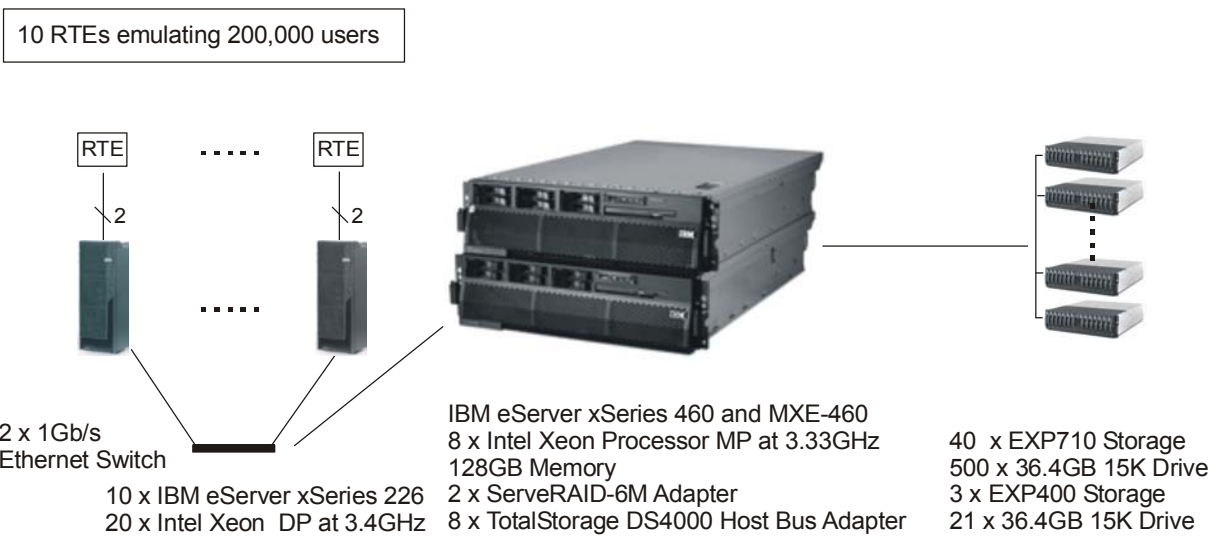


**IBM® eServer® xSeries® 460 c/s
and
DB2® UDB 8.2**

TPC-C Rev. 5.3

Report Date: May 31, 2005

Total System Cost	TPC-C Throughput	Price/Performance	Availability Date	
\$1,440,290 USD	250,975 tpmC	\$5.74 USD / tpmC	Nov. 30, 2005	
Database Server Processors/Cores/Threads	Database Manager	Operating System	Other Software	Number of Users
8/8/16 64-Bit Intel® Xeon™ Processor MP at 3.33GHz	DB2 UDB 8.2	Microsoft Windows® Server 2003 Enterprise x64 Edition	Microsoft Visual C++ 6.0 Microsoft COM+	200,000



System Component	Qty	Server:	Qty	Each of Ten Clients:
Processors/Cores/Threads	8/8/16	Intel Xeon Processor MP at 3.33GHz/8MB L3 Cache	2/2/4	Intel Xeon DP at 3.4GHz 2MB L2 Cache
Cache				
Memory	32	4GB ECC RDIMM		512MB 256MB
Disk Controllers	2	ServeRAID-6M Adapter	4	
Disk Drives	500	36.4GB 2Gbps FC (15K) 36.4GB Ultra320 (15K)	2 1	Ultra320 SCSI Interface 36.4GB (15000 rpm)
Total Storage	21	18600GB	1	

IBM Corporation	IBM @server xSeries 460 c/s with DB2 UDB 8.2			TPC-C Revision 5.3			
				Report Date: May 31, 2005			
Description	Part Number	Third Party Brand	Pricing	Unit Price	Quantity	Extended Price	3-Yr. Maint. Price
Server Hardware							
xSeries 460 with 2 x 3.33GHz/8MB L3 Cache Intel Xeon Processor MP	8872-3RU	IBM		1	24,999	1	24,999
xSeries 3.33GHz/8MB L3 Cache Intel Xeon Processor MP	13N0713	IBM		1	5,699	6	34,194
MXE-460 (0 Processors, 0 Memory, 2 Memory Cards)	8874-1RU	IBM		1	6,999	1	6,999
Scalability Cable 2.3.M	13M7414	IBM		1	299	2	598
8GB (2x4GB) PC2-3200 CL3 2RX4 ECC DDR2 SDRAM RDIMM	30R5145	IBM		1	17,879	16	286,064
Active Memory™ 4-Slot Memory Expansion Card	13M7409	IBM		1	499	4	1,996
ServeRAID-6M Ultra320 SCSI Adapter	32P0033	IBM		1	999	2	1,998
E54 15" (13.8" Viewable) Color Monitor	633147N	IBM		1	149	1	149
IBM Preferred Pro Full-Size Keyboard PS/2	31P7415	IBM		1	29	1	29
IBM Sleek 2-Button Mouse	28L3673	IBM		1	15	1	15
ServicePac for 3-Year 24x7x4 Support (x460 and MXE-460)	96P2253	IBM		1	3,390	2	6,780
ServicePac for 3-Year 24x7x4 Support (Monitor)	30L9183	IBM		1	90	1	90
Discount on xSeries Hardware (15%)						53,556	
Discount on ServicePacs (20%)							1,374
					Subtotal	303,485	5,496
Server Storage							
IBM TotalStorage DS4000 Host Bus Adapter	24P0960	IBM		1	1,485	8	11,880
IBM TotalStorage DS4500 Midrange Disk Subsystem	174290U	IBM		1	49,900	4	199,600
IBM DS4000 Mini Hub	19K1269	IBM		1	899	8	7,192
IBM Short Wave SFP Module	19K1271	IBM		1	499	168	83,832
IBM 1m LC-LC Fibre Channel Cable	19K1247	IBM		1	79	80	6,320
IBM 5m LC-LC Fibre Channel Cable	19K1248	IBM		1	129	8	1,032
IBM TotalStorage DS4000 EXP710 Storage Exp. Unit	1740710	IBM		1	6,000	40	240,000
2Gbps FC 36.4GB 15K Hot-Swap HDD	06P5772	IBM		1	1,115	500	557,500
IBM EXP400 Rack Storage Exp. Enclosure	17331RU	IBM		1	3,099	3	9,297
IBM 36.4GB 15K Ultra320 SCSI Drive	90P1318	IBM		1	349	21	7,329
2M SCSI cable	03K9310	IBM		1	75	3	225
IBM UPS 750TLV	21301TX	IBM		1	299	1	299
IBM S2 42U Standard Rack	93074SX	IBM		1	1,489	5	7,445
ServicePac for 3-Year 24x7x4 Support (EXP710)	41L2768	IBM		1	760	34	25,840
ServicePac for 3-Year 24x7x4 Support (DS4500)	96P2062	IBM		1	1,087	5	5,435
ServicePac for 3-Year 24x7x4 Support (Rack)	41L2760	IBM		1	300	5	1,500
Discount on Fibre Channel Disks (30%)						167,250	
Discount on Fibre Channel Storage (20%)						109,971	
Discount on xSeries Storage (15%)						3,689	
Discount on ServicePacs (20%)							6,555
					Subtotal	851,041	26,220
Server Software							
DB2 UDB ESE 8.2 for Windows Operating Systems on 64-Bit Extended Systems - SW License and Maintenance 12 Months		IBM		2	22,608	8	180,864
SW Maintenance Renewal - 1 Year		IBM		2	1,077	16	17,232
Microsoft Windows Server 2003 Enterprise x64 Edition*	P72-00264	Microsoft		3	2,334	1	2,334
Microsoft Problem Resolution Services		Microsoft		3	245	1	245
					Subtotal	183,198	17,477
Client Hardware							
x226 with 3.4GHz/2MB Xeon DP, 512MB (2x256MB) Memory	8648-6AU	IBM		1	1,939	10	19,390
3.4GHz/2MB Xeon DP Processor Upgrade	13N0674	IBM		1	1,049	10	10,490
1GB (2x512MB) PC-3200 DDR2 ECC SDRAM RDIMM	73P3522	IBM		1	399	20	7,980
36.4GB 10K Ultra320 SCSI Drive	90P1304	IBM		1	329	10	3,290
PRO/1000 MT Dual-Port Server Adapter	73P2701	Intel		1	249	10	2,490
E54 15" (13.8" Viewable) Color Monitor	633147N	IBM		1	149	10	1,490
ServicePac for 3-Year 24x7x4 Support (x226)	96P2250	IBM		1	750	10	7,500
ServicePac for 3-Year 24x7x4 Support (Monitor)	30L9183	IBM		1	90	10	900
Discount on xSeries Hardware (15%)						6,770	
Discount on ServicePacs (20%)							1,680
					Subtotal	38,361	6,720
Client Software							
Microsoft Windows 2000 Server with COM+*	C11-00821	Microsoft		3	738	10	7,380
Microsoft Visual C++ Professional 6.0	254-00170	Microsoft		3	109	1	109
					Subtotal	7,489	
Network Components							
DLink Gigabit Ethernet Switch (2 spares)				4	189	4	756
Ethernet Cable 14 Ft. (2 spares)				4	4	12	48
					Subtotal	804	
Discounts based on volume purchase; prices vary if purchased separately.							
					Total	1,384,377	55,913
For Pricing and Ordering Information: 1- IBM (1-888-SHOP-IBM, Keycode 7764) 2 - IBM DB2: Bernard Spang, Dir. Database Marketing Management, spang@us.ibm.com, 914-766-1491. 3 - Microsoft (*See Quote for Discounts); 4 - newegg.com Audited by Francois Raab, InfoSizing, Inc					Three-Year Cost of Ownership USD: \$1,440,290 tpmC: 250,975 \$ USD/tpmC: \$5.74		
Prices used in TPC benchmarks reflect the actual prices a customer would pay for a one-time purchase of the stated components. Individually negotiated discounts are not permitted. Special prices based on assumptions about past or future purchases are not permitted. All discounts reflect standard pricing policies for the listed components. For complete details, see the pricing sections of the TPC benchmark specifications. If you find that stated prices are not available according to these terms, please inform the TPC at pricing@tpc.org.							

Numerical Quantities Summary			
MQTh, Computed Maximum Qualified Throughput: 250,975 tpmC			
Response Times (in seconds)	90th Percentile	Average	Maximum
New-Order	0.34	0.21	1.80
Payment	0.40	0.32	3.34
Delivery	0.30	0.30	1.52
Stock Level	0.42	0.33	2.05
Order Status	0.42	0.24	1.94
Delivery (Deferred)	0.48	0.30	2.78
Menu	0.29	0.21	1.59
Response Time Delay Added for Emulated Components: 0.1 Seconds			
Transaction Mix (in percent of total transactions)			Percent
New-Order			44.96
Payment			43.02
Delivery			4.01
Stock-Level			4.01
Order Status			4.01
Keying/Think Times (in seconds)	Minimum	Average	Maximum
New Order	18.00 / 0.00	18.00 / 12.04	18.03 / 120.33
Payment	3.00 / 0.00	3.00 / 12.04	3.02 / 120.33
Delivery	2.00 / 0.00	2.00 / 5.03	2.02 / 50.31
Stock Level	2.00 / 0.00	2.00 / 5.04	2.02 / 50.31
Order Status	2.00 / 0.00	2.00 / 10.04	2.02 / 100.33
Test Duration			
Ramp-up time			57 minutes 40 seconds
Measurement interval			120 minutes
Number of checkpoints			NA
Checkpoint interval			NA
Number of transactions (all types) completed in measurement interval			66,992,300

Table of Contents

Abstract	3
Numerical Quantities Summary	5
Preface	12
General Items	13
Application Code Disclosure and Definition Statements	13
Benchmark Sponsor	13
Parameter Settings	13
Configuration Diagrams	13
Clause 1: Logical Database Design Related Items	15
Table Definitions	15
Physical Organization of the Database	15
Insert and Delete Operations	15
Horizontal or Vertical Partitioning	15
Replication	15
Table Attributes	15
Clause 2: Transaction and Terminal Profiles Related Items	16
Random Number Generation	16
Screen Layout	16
Terminal Verification	16
Intelligent Terminals	16
Transaction Profiles	16
Deferred Delivery Mechanism	17
Clause 3: Transaction and System Properties Related Items	18
Atomicity Requirements	18
Consistency Requirements	18
Isolation Requirements	19
Durability Requirements	19
Clause 4: Scaling and Database Population Related Items	21
Cardinality of Tables	21
Distribution of Tables and Logs	21
Database Model Implemented	36
Partitions/Replications Mapping	36
60-Day Space Requirement	37
Clause 5: Performance Metrics and Response Time Related Items	38
Measured tpmC	38
Response Times	38
Keying/Think Times	38
Response Time Frequency Distribution Curves	39
Performance Curve for Response Time vs. Throughput	41
New Order Think Time Distribution	42
Steady State Methodology	43
Work Performed during Steady State	43
Measurement Interval	44
Transaction Mix	44
Percentage of Total Mix	44
Number of Checkpoints	45
Clause 6: SUT, Driver and Communication Definition Related Items	46
Description of RTE	46
Emulated Components	46
Benchmarked and Targeted System Configuration Diagrams	46
Network Configuration	46
Network Bandwidth	46
Operator Intervention	46

Clause 7: Pricing Related Items	47
Hardware and Software Components	47
Availability Date	47
Measured tpmC	47
Country-Specific Pricing	47
Usage Pricing	47
System Pricing	48
Clause 9: Audit Related Items	49
Auditor	49
Availability of the Full Disclosure Report	49
<i>Attestation letter</i>	50
Appendix A: Client Server Code	52
A.1 Client/Terminal Handler Code	52
<i>makefile.config</i>	52
<i>include/tpccapp.h</i>	55
<i>include/tpccdbg.h</i>	55
<i>Src.Common/Makefile</i>	56
<i>Src.Common/tpccctx.sqc</i>	57
<i>Src.Common/tpccdbg.c</i>	58
<i>Src.Cli/Makefile</i>	62
<i>Src.Cli/tpcccli.sqc</i>	63
<i>nullDB/nullDB.cpp</i>	69
<i>nullDB/stdafx.h</i>	72
<i>nullDB/stdafx.cpp</i>	72
<i>tpccIsapi/htmlPhraser.h</i>	72
<i>tpccIsapi/resource.h</i>	73
<i>tpccIsapi/StdAfx.h</i>	73
<i>tpccIsapi/tpcc.h</i>	73
<i>tpccIsapi/tpccIsapi.def</i>	83
<i>tpccIsapi/tpccIsapi.hpp</i>	83
<i>tpccIsapi/htmlPhraser.cpp</i>	84
<i>tpccIsapi/StdAfx.cpp</i>	86
<i>tpccIsapi/tpccIsapi.cpp</i>	86
A.2 Client Transaction Code	111
<i>Makefile.config</i>	111
<i>tpccenv.bat</i>	112
<i>include/db2tpcc.h</i>	112
<i>include/lval.h</i>	114
<i>include/tpccapp.h</i>	114
<i>include/tpccdbg.h</i>	115
<i>Src.Common/Makefile</i>	115
<i>Src.Common/tpccctx.sqc</i>	116
<i>Src.Common/tpccdbg.c</i>	117
<i>Src.Common/tpccmisc.c</i>	121
<i>Src.Srv/Makefile</i>	121
<i>Src.Srv/cat-func.ddl</i>	123
<i>Src.Srv/cat-proc.ddl</i>	128
<i>Src.Srv/tpcc_all_sql.sqc</i>	128
<i>Src.Srv/unecat_func.ddl</i>	154
<i>Src.Srv/unecat_proc.ddl</i>	154
<i>Src.Srv/rpctpcc.def</i>	154
<i>utils/EXPLAIN.ddl</i>	154
<i>utils/UNEXPLAIN.ddl</i>	158
<i>tpccCom/comreg.h</i>	159
<i>tpccCom/dlldatax.h</i>	160

<i>tpccCom/Resource.h</i>	160
<i>tpccCom/stdafx.h</i>	160
<i>tpccCom/tpccCom.h</i>	160
<i>tpccCom/tpcc_com.h</i>	162
<i>tpccCom/tpccCom.def</i>	163
<i>tpccCom/tpccCom.idl</i>	163
<i>tpccCom/tpcc_com.rgs</i>	163
<i>tpccCom/comreg.cpp</i>	164
<i>tpccCom/stdafx.cpp</i>	164
<i>tpccCom/tpccCom.cpp</i>	164
<i>tpccCom/tpcc_com.cpp</i>	164
<i>TpccCom/dlldata.c</i>	168
<i>tpccCom/dlldatax.c</i>	168
<i>tpccCom/tpccCom_i.c</i>	169
<i>tpccCom/tpccCom_p.c</i>	169
<i>TpccDB2Glue/stdafx.h</i>	180
<i>tpccDB2Glue/tpccDB2glue.h</i>	180
<i>tpccDB2Glue/stdafx.cpp</i>	180
<i>tpccDB2Glue/tpccDB2glue.cpp</i>	180
<i>NullDB.cpp</i>	185
<i>NullDB.h</i>	188
<i>Stdafx.cpp</i>	188
<i>Stdafx.h</i>	188
<i>Stdafx.cpp</i>	189
<i>StdAfx.h</i>	189
<i>TpccComClient.cpp</i>	189
<i>HtmlPhraser.cpp</i>	190
<i>HtmlPhraser.h</i>	192
<i>Resource.h</i>	193
<i>StdAfx.cpp</i>	194
<i>StdAfx.h</i>	194
<i>Tpcc.h</i>	194
<i>TpccIsapi.cpp</i>	205
<i>TpccIsapi.def</i>	232
<i>TpccIsapi.hpp</i>	232
<i>TpccIsapi.rc</i>	234
Appendix B: Database Design Scripts	235
<i>create_tablespace.ddl</i>	235
<i>alter_tablespace.ddl</i>	247
<i>alter_bufferpool.ddl</i>	249
<i>create_bufferpool.ddl</i>	250
<i>create_database.ddl</i>	250
<i>alittbsp_pf_0.ddl</i>	250
<i>alittbsp_pf_1024.ddl</i>	252
<i>alittbsp_pf_4096.ddl</i>	255
<i>crconst_customer.ddl</i>	257
<i>crconst_district.ddl</i>	258
<i>crconst_history.ddl</i>	259
<i>crconst_new_ordera.ddl</i>	261
<i>crconst_new_orderb.ddl</i>	262
<i>crconst_order_line.ddl</i>	264
<i>crconst_orders.ddl</i>	266
<i>crconst_stock.ddl</i>	267
<i>crconst_warehouse.ddl</i>	268
<i>cridx_cust_idxb.ddl</i>	270

<i>cridx_ordr_idxb.ddl</i>	271
<i>crtb_customer.ddl</i>	272
<i>crtb_item.ddl</i>	279
<i>crtb_district.ddl</i>	279
<i>crtb_orders.ddl</i>	283
<i>crtb_order_line.ddl</i>	287
<i>crtb_new_ordera.ddl</i>	292
<i>crtb_new_orderb.ddl</i>	295
<i>crtb_stock.ddl</i>	298
<i>crtb_history.ddl</i>	304
<i>crtb_warehouse.ddl</i>	307
<i>crvw_customer.ddl</i>	311
<i>crvw_district.ddl</i>	311
<i>crvw_history.ddl</i>	311
<i>crvw_new_order.ddl</i>	312
<i>crvw_order_line.ddl</i>	312
<i>crvw_orders.ddl</i>	313
<i>crvw_stock.ddl</i>	313
<i>crvw_warehouse.ddl</i>	313
<i>gen_customer.bat</i>	314
<i>gen_district.bat</i>	314
<i>gen_history.bat</i>	314
<i>gen_item.bat</i>	316
<i>gen_new_order.bat</i>	316
<i>gen_orders.bat</i>	316
<i>gen_stock.bat</i>	317
<i>gen_warehouse.bat</i>	317
<i>load_customer_all.ddl</i>	318
<i>load_district_all.ddl</i>	319
<i>Load_history_all.ddl</i>	320
<i>connect to TPCC in share mode;</i>	320
<i>load_item_all.ddl</i>	322
<i>load_new_order_all.ddl</i>	322
<i>load_order_line_all.ddl</i>	323
<i>load_orders_all.ddl</i>	324
<i>load_stock_all.ddl</i>	326
<i>load_warehouse_all.ddl</i>	327
<i>rnst_customer.ddl</i>	328
<i>rnst_district.ddl</i>	329
<i>rnst_history.ddl</i>	330
<i>rnst_item.ddl</i>	331
<i>rnst_new_ordera.ddl</i>	331
<i>rnst_new_orderb.ddl</i>	332
<i>rnst_order_line.ddl</i>	333
<i>rnst_orders.ddl</i>	334
<i>rnst_stock.ddl</i>	334
<i>rnst_warehouse.ddl</i>	335
<i>DBGEN</i>	336
<i>dbgen\gendata.c</i>	336
<i>dbgen\makefile</i>	344
<i>dbgen\tpccrnd.c</i>	345
<i>dbgen\include\db2tpcc.h</i>	347
<i>dbgen\include\lval.h</i>	349
<i>dbgen\include\platform.h</i>	350
<i>dbgen\include\tpccrnd.h</i>	350

<i>dbgen\makefile.config</i>	350
<i>dbgen\Src.Common\makefile</i>	351
<i>dbgen\Src.Common\tpccmisc.c</i>	352
<i>dbgen\tpccenv.bat</i>	352
Appendix C: Tunable Parameters	354
IBM DB2 UDB	354
<i>Database Manager Configuration</i>	354
<i>Database Configuration</i>	355
<i>DB2set Parameters</i>	357
<i>Aff.cfg</i>	357
Microsoft Windows Server 2003 Enterprise x64 Edition	357
<i>Server Configuration Parameters</i>	357
<i>System Information Report</i>	357
ServeRAID-6M Disk Controller Configuration Parameters	441
DS4500 Disk Subsystem Configuration	446
Client Configuration	505
<i>Microsoft Windows 2000 Client System Information Report</i>	505
<i>Client Configuration Parameters</i>	540
<i>Microsoft Windows 2000 Client Registry Parameters</i>	540
RTE Input Parameters	542
Appendix D: 60-Day Space	544
Appendix E: Third-Party Quotations	545

Preface

The TPC Benchmark™ C was developed by the Transaction Processing Performance Council (TPC). The TPC was founded to define transaction processing benchmarks and to disseminate objective, verifiable performance data to the industry. This full disclosure report is based on the TPC Benchmark C Standard Specification Version 5.3.

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

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

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

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

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

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

General Items

Benchmark Sponsor

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

This benchmark was sponsored by International Business Machines Corporation.

Application Code Disclosure and Definition Statements

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

Appendix A contains all source code implemented in this benchmark.

Parameter Settings

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

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

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

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

Configuration Diagrams

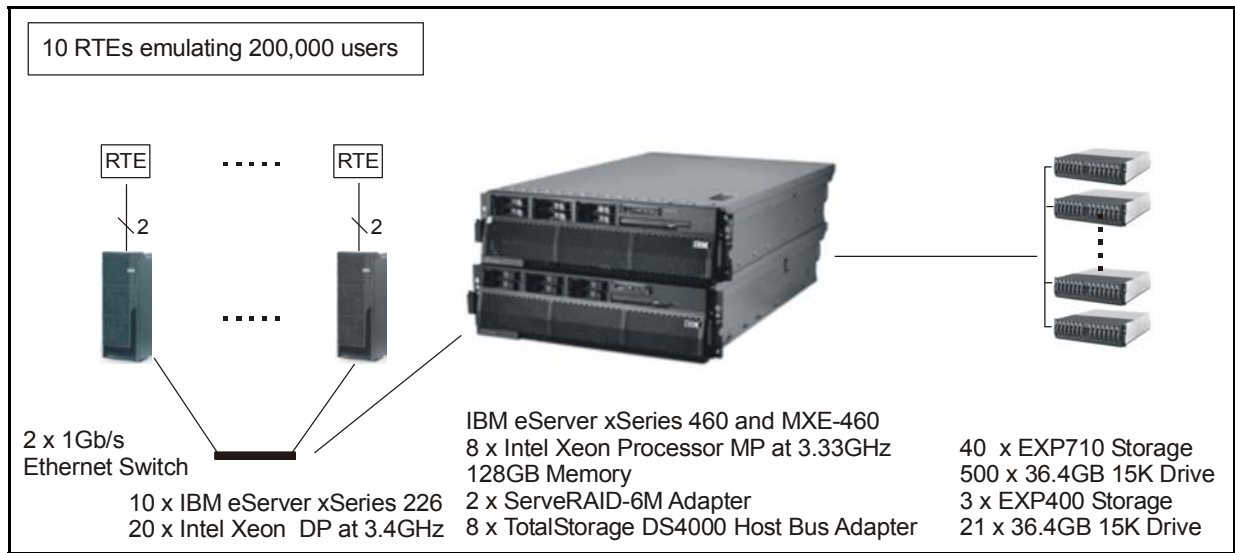
Diagrams of both measured and priced configurations must be provided, accompanied by a description of the differences.

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

The Remote Terminal Emulator (RTE) used for these TPC Benchmark C tests is an IBM proprietary RTE. Under Version 5.3, the components of the configuration being emulated by the RTE are the workstations and the Ethernet hubs. Appendix C contains a listing of the RTE scripts and inputs used in the benchmark testing.

The benchmarked configuration used 10 IBM eServer xSeries 226 systems, each configured with two 3.4GHz Intel Xeon processors, as the clients, which executed the terminal I/O and submitted transactions to COM+ servers, which are also running on the clients. These COM+ servers forwarded the transaction requests to the server, and returned the results to the RTE. DB2 UDB 8.2 is the DBMS executing on the server

Measured Configuration



The measured and priced configurations were identical. For the priced configuration, see the Executive Summary.

Clause 1: Logical Database Design Related Items

Table Definitions

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

Physical Organization of the Database

The physical organization of tables and indexes within the database must be disclosed. Physical space was allocated to DB2 UDB on the server disks as detailed in Figure 4-2.

Insert and Delete Operations

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

All insert and delete functions were fully operational during the running of the benchmark. The space required for an additional 5 percent of the initial table cardinality was allocated to DB2 UDB and priced as static space.

Horizontal or Vertical Partitioning

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

All tables except Item table were horizontally partitioned into 25 tables of 800 warehouses each. For each partitioned table, a view was created over all table partitions to provide full transparency of data manipulation.

Replication

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

Table Attributes

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

No additional attributes were used in this benchmark.

Clause 2: Transaction and Terminal Profiles Related Items

Random Number Generation

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

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

Screen Layout

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

All screen layouts followed the TPC Benchmark C Standard Specification.

Terminal Verification

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

The auditor verified terminal features by direct experimentation. The benchmarked configuration uses Microsoft Internet Explorer 6.0 SP1 and HTML scripts as the terminal interface.

Intelligent Terminals

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

The terminals emulated in the priced configuration are IBM PC desktop computer systems. All processing of the input/output screens was handled by the xSeries 226 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 460 database server.

Transaction Profiles

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

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

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

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

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

Table 2-1. Transaction Statistics

New Order	Value (%)
Home warehouse order lines	99.00
Remote warehouse order lines	1.00
Rolled back transactions	1.00
Average number of items per order	10.00
Payment	
Home warehouse payment transactions	0.85
Remote warehouse payment transactions	0.15
Non-Primary Key Access	
Payment transactions using C_LAST	60.00
Order-Status transactions using C_LAST	59.99
Delivery	
Delivery transactions skipped	0
Transaction Mix	
New-Order	44.96
Payment	43.02
Delivery	4.01
Stock Level	4.01
Order Status	4.01

Deferred Delivery Mechanism

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

The Delivery transaction was submitted to an ISAPI queue that is separate from the COM+ queue that the other transactions used. This queue is serviced by a variable amount of threads that are separate from the worker threads inside the web server. Web server threads are able to complete the on-line part of the Delivery transaction and immediately return successful queuing responses to the drivers. The threads servicing the queue are responsible for completing the deferred part of the transaction asynchronously.

The source code is listed in Appendix A.

Clause 3: Transaction and System Properties Related Items

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

Atomicity Requirements

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

All ACID tests were conducted according to specification.

Completed Transactions

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

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

Aborted Transactions

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

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

Consistency Requirements

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

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

Isolation Requirements

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

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

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

Case A was followed for Isolation test seven.

Durability Requirements

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

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

Loss of Data Test

The DS4500 Disk Subsystem contains two RAID controller blades, which provide RAID functionality to the attached disks. Each RAID controller blade contains a Read/Write cache. Write caching was enabled. The attached disks contained a portion of each of the tables in the tpcc database. During steady state one RAID controller blade was desinserted causing DB2 to report errors accessing that device.

The following steps were successfully performed to pass the Durability test of failure of a DS4500 controller with database tables:

1. The contents of the database were backed up to several database dump devices during the initial database load.
2. The current count of the total number of orders was determined by the sum of D_NEXT_O_ID for all rows in the district table giving SUM1.
3. A test was started with 60,800 users submitting transactions.
4. A DS4500 controller blade containing a portion of each of the tables in the tpcc database was pulled out causing DB2 to report errors accessing that device.
5. The run was aborted and DB2 was stopped.
6. The failed DS4500 controller blade was reinserted and was recovered.
7. The database was recovered and restored from the backup dump devices. Afterwards, the transaction log was rolled forward to the database.
8. Step 2 was repeated to obtain the current count of the total number of orders giving SUM2.
9. It was verified that the sum of D_NEXT_O_ID after the database is recovered is greater than or equal to the sum of D_NEXT_O_ID before the run, plus all new order transactions completed during the run minus any rollback transactions.
10. Consistency Condition 3 was verified.

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

1. The current count of the total number of orders was determined by the sum of D_NEXT_O_ID for all rows in the district table giving SUM1.
2. This test was executed on a full scale benchmark run with 200,000 users.

3. The test continued to run for 5 minutes after all users were connected to the server.
4. One disk from the log array was removed. Since the disk was RAID-1 mirrored, DB2 continued to process transactions without interruption.
5. The test continued to run for another 5 minutes.
6. The server under test was powered off, which removed power from the system and the memory.
7. The server was powered on again.
8. DB2 was allowed to recover.
9. Step 1 was repeated to obtain the current count of the total number of orders giving SUM2.
10. It was verified that the sum of D_NEXT_O_ID after the database recovered was greater than or equal to the sum of D_NEXT_O_ID before the run, plus all new order transactions completed during the run minus any rollback transactions.

Clause 4: Scaling and Database Population Related Items

Cardinality of Tables

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

The database was built with 20,000 warehouses and the audited run used 20,000 warehouses.

Table 4-1. Initial Cardinality of Tables

Table Name	Rows
Warehouse	20,000
District	200,000
Item	100,000
New Order	180,000,000
History	600,000,000
Orders	600,000,000
Customer	600,000,000
Order Line	6,000,180,579
Stock	2,000,000,000
Inactive Warehouses	0

Distribution of Tables and Logs

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

The logs are configured as a RAID-10 disk array with 20 physical disks of 36.4GB each attached via an internal ServeRAID-6M adapter.

For the database tables, there is a total of 500 disk drives. Each physical disk has a capacity of 36.4GB. A total of eight Fibre Channel storage adapters connect these 500 disks.

Each disk volume, as seen by the Windows operating system, is configured as a RAID-0 disk array with 10 physical disks. There is a total of 50 such disk volumes. Each disk volume is further partitioned into 12 partitions. Each partition corresponds to a DB2 container.

Each partition within a partitioned table is made of two DB2 containers; thus, there are 25 partitioned tables with a total of 50 DB2 containers so that the corresponding view spans all eight adapters connected to the 500 disks.

The ITEM table, which is not partitioned, is made up of 50 DB2 containers and also span all eight adapters.

A total of 60 drives held six RAID-5 arrays, which were used to store the backup image only. These 60 drives were not priced.

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

Figure 4-2. Data Distribution for the Benchmarked Configuration

Disk #	Drives	Partition	Size	Use
0	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\001 C:\Containers\WAR\001 C:\Containers\DIS\001 C:\Containers\CSTI\001 C:\Containers\NEWA\001 C:\Containers\OLN\001 C:\Containers\STK\001 C:\Containers\CST\001 C:\Containers\ORDI\001 C:\Containers\ORD\001 C:\Containers\HST\001 C:\Containers\NEWB\001	334.00GB	Database Backup Files
1	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\002 C:\Containers\WAR\002 C:\Containers\DIS\002 C:\Containers\CSTI\002 C:\Containers\NEWA\002 C:\Containers\OLN\002 C:\Containers\STK\002 C:\Containers\CST\002 C:\Containers\ORDI\002 C:\Containers\ORD\002 C:\Containers\HST\002 C:\Containers\NEWB\002	334.00GB	Database Backup Files
2	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\003 C:\Containers\WAR\003 C:\Containers\DIS\003 C:\Containers\CSTI\003 C:\Containers\NEWA\003 C:\Containers\OLN\003 C:\Containers\STK\003 C:\Containers\CST\003 C:\Containers\ORDI\003 C:\Containers\ORD\003 C:\Containers\HST\003 C:\Containers\NEWB\003	334.00GB	Database Backup Files
3	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\004 C:\Containers\WAR\004 C:\Containers\DIS\004 C:\Containers\CSTI\004 C:\Containers\NEWA\004 C:\Containers\OLN\004 C:\Containers\STK\004 C:\Containers\CST\004 C:\Containers\ORDI\004 C:\Containers\ORD\004 C:\Containers\HST\004 C:\Containers\NEWB\004	334.00GB	Database Backup Files

4	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\005 C:\Containers\WAR\005 C:\Containers\DIS\005 C:\Containers\CSTI\005 C:\Containers\NEWA\005 C:\Containers\OLN\005 C:\Containers\STK\005 C:\Containers\CST\005 C:\Containers\ORDI\005 C:\Containers\ORD\005 C:\Containers\HST\005 C:\Containers\NEWB\005	334.00GB	Database Backup Files
5	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\006 C:\Containers\WAR\006 C:\Containers\DIS\006 C:\Containers\CSTI\006 C:\Containers\NEWA\006 C:\Containers\OLN\006 C:\Containers\STK\006 C:\Containers\CST\006 C:\Containers\ORDI\006 C:\Containers\ORD\006 C:\Containers\HST\006 C:\Containers\NEWB\006	334.00GB	Database Backup Files
6	10 - 36.4GB EXP710 Enclosure	C:\Backup	300.60GB	Backup Files

Disk #	Drives	Partition	Size	Use
7	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\007 C:\Containers\WAR\007 C:\Containers\DIS\007 C:\Containers\CSTI\007 C:\Containers\NEWA\007 C:\Containers\OLN\007 C:\Containers\STK\007 C:\Containers\CST\007 C:\Containers\ORDI\007 C:\Containers\ORD\007 C:\Containers\HST\007 C:\Containers\NEWB\007	334.00GB	Database Backup Files
8	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\008 C:\Containers\WAR\008 C:\Containers\DIS\008 C:\Containers\CSTI\008 C:\Containers\NEWA\008 C:\Containers\OLN\008 C:\Containers\STK\008 C:\Containers\CST\008 C:\Containers\ORDI\008 C:\Containers\ORD\008 C:\Containers\HST\008 C:\Containers\NEWB\008	334.00GB	Database Backup Files
9	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\009 C:\Containers\WAR\009 C:\Containers\DIS\009 C:\Containers\CSTI\009 C:\Containers\NEWA\009 C:\Containers\OLN\009 C:\Containers\STK\009 C:\Containers\CST\009 C:\Containers\ORDI\009 C:\Containers\ORD\009 C:\Containers\HST\009 C:\Containers\NEWB\009	334.00GB	Database Backup Files
10	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\010 C:\Containers\WAR\010 C:\Containers\DIS\010 C:\Containers\CSTI\010 C:\Containers\NEWA\010 C:\Containers\OLN\010 C:\Containers\STK\010 C:\Containers\CST\010 C:\Containers\ORDI\010 C:\Containers\ORD\010 C:\Containers\HST\010 C:\Containers\NEWB\010	334.00GB	Database Backup Files

11	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\011 C:\Containers\WAR\011 C:\Containers\DIS\011 C:\Containers\CST\011 C:\Containers\NEWA\011 C:\Containers\OLN\011 C:\Containers\STK\011 C:\Containers\CST\011 C:\Containers\ORD\011 C:\Containers\ORD\011 C:\Containers\HST\011 C:\Containers\NEWB\011	334.00GB	Database Backup Files
12	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\012 C:\Containers\WAR\012 C:\Containers\DIS\012 C:\Containers\CST\012 C:\Containers\NEWA\012 C:\Containers\OLN\012 C:\Containers\STK\012 C:\Containers\CST\012 C:\Containers\ORD\012 C:\Containers\ORD\012 C:\Containers\HST\012 C:\Containers\NEWB\012	334.00GB	Database Backup Files
13	10 - 36.4GB EXP710 Enclosure	C:\Backup	300.60GB	Backup Files

Disk #	Drives	Partition	Size	Use
14	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\013 C:\Containers\WAR\013 C:\Containers\DIS\013 C:\Containers\CSTI\013 C:\Containers\NEWA\013 C:\Containers\OLN\013 C:\Containers\STK\013 C:\Containers\CST\013 C:\Containers\ORDI\013 C:\Containers\ORD\013 C:\Containers\HST\013 C:\Containers\NEWB\013	334.00GB	Database Backup Files
15	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\014 C:\Containers\WAR\014 C:\Containers\DIS\014 C:\Containers\CSTI\014 C:\Containers\NEWA\014 C:\Containers\OLN\014 C:\Containers\STK\014 C:\Containers\CST\014 C:\Containers\ORDI\014 C:\Containers\ORD\014 C:\Containers\HST\014 C:\Containers\NEWB\014	334.00GB	Database Backup Files
16	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\015 C:\Containers\WAR\015 C:\Containers\DIS\015 C:\Containers\CSTI\015 C:\Containers\NEWA\015 C:\Containers\OLN\015 C:\Containers\STK\015 C:\Containers\CST\015 C:\Containers\ORDI\015 C:\Containers\ORD\015 C:\Containers\HST\015 C:\Containers\NEWB\015	334.00GB	Database Backup Files
17	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\016 C:\Containers\WAR\016 C:\Containers\DIS\016 C:\Containers\CSTI\016 C:\Containers\NEWA\016 C:\Containers\OLN\016 C:\Containers\STK\016 C:\Containers\CST\016 C:\Containers\ORDI\016 C:\Containers\ORD\016 C:\Containers\HST\016 C:\Containers\NEWB\016	334.00GB	Database Backup Files

18	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\017 C:\Containers\WAR\017 C:\Containers\DIS\017 C:\Containers\CST\017 C:\Containers\NEWA\017 C:\Containers\OLN\017 C:\Containers\STK\017 C:\Containers\CST\017 C:\Containers\ORD\017 C:\Containers\ORD\017 C:\Containers\HST\017 C:\Containers\NEWB\017	334.00GB	Database Backup Files
19	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\018 C:\Containers\WAR\018 C:\Containers\DIS\018 C:\Containers\CST\018 C:\Containers\NEWA\018 C:\Containers\OLN\018 C:\Containers\STK\018 C:\Containers\CST\018 C:\Containers\ORD\018 C:\Containers\ORD\018 C:\Containers\HST\018 C:\Containers\NEWB\018	334.00GB	Database Backup Files
20	10 - 36.4GB EXP710 Enclosure	C:\Backup	300.60GB	Backup Files

Disk #	Drives	Partition	Size	Use
21	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\019 C:\Containers\WAR\019 C:\Containers\DIS\019 C:\Containers\CSTI\019 C:\Containers\NEWA\019 C:\Containers\OLN\019 C:\Containers\STK\019 C:\Containers\CST\019 C:\Containers\ORD\019 C:\Containers\ORD\019 C:\Containers\HST\019 C:\Containers\NEWB\019	334.00GB	Database Backup Files
22	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\020 C:\Containers\WAR\020 C:\Containers\DIS\020 C:\Containers\CSTI\020 C:\Containers\NEWA\020 C:\Containers\OLN\020 C:\Containers\STK\020 C:\Containers\CST\020 C:\Containers\ORD\020 C:\Containers\ORD\020 C:\Containers\HST\020 C:\Containers\NEWB\020	334.00GB	Database Backup Files
23	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\021 C:\Containers\WAR\021 C:\Containers\DIS\021 C:\Containers\CSTI\021 C:\Containers\NEWA\021 C:\Containers\OLN\021 C:\Containers\STK\021 C:\Containers\CST\021 C:\Containers\ORD\021 C:\Containers\ORD\021 C:\Containers\HST\021 C:\Containers\NEWB\021	334.00GB	Database Backup Files
24	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\022 C:\Containers\WAR\022 C:\Containers\DIS\022 C:\Containers\CSTI\022 C:\Containers\NEWA\022 C:\Containers\OLN\022 C:\Containers\STK\022 C:\Containers\CST\022 C:\Containers\ORD\022 C:\Containers\ORD\022 C:\Containers\HST\022 C:\Containers\NEWB\022	334.00GB	Database Backup Files

25	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\023 C:\Containers\WAR\023 C:\Containers\DIS\023 C:\Containers\CSTI\023 C:\Containers\NEWA\023 C:\Containers\OLN\023 C:\Containers\STK\023 C:\Containers\CST\023 C:\Containers\ORDI\023 C:\Containers\ORD\023 C:\Containers\HST\023 C:\Containers\NEWB\023	334.00GB	Database Backup Files
26	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\024 C:\Containers\WAR\024 C:\Containers\DIS\024 C:\Containers\CSTI\024 C:\Containers\NEWA\024 C:\Containers\OLN\024 C:\Containers\STK\024 C:\Containers\CST\024 C:\Containers\ORDI\024 C:\Containers\ORD\024 C:\Containers\HST\024 C:\Containers\NEWB\024	334.00GB	Database Backup Files
27	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\025 C:\Containers\WAR\025 C:\Containers\DIS\025 C:\Containers\CSTI\025 C:\Containers\NEWA\025 C:\Containers\OLN\025 C:\Containers\STK\025 C:\Containers\CST\025 C:\Containers\ORDI\025 C:\Containers\ORD\025 C:\Containers\HST\025 C:\Containers\NEWB\025	334.00GB	Database Backup Files
28	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\026 C:\Containers\WAR\026 C:\Containers\DIS\026 C:\Containers\CSTI\026 C:\Containers\NEWA\026 C:\Containers\OLN\026 C:\Containers\STK\026 C:\Containers\CST\026 C:\Containers\ORDI\026 C:\Containers\ORD\026 C:\Containers\HST\026 C:\Containers\NEWB\026	334.00GB	Database Backup Files
29	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\027 C:\Containers\WAR\027 C:\Containers\DIS\027 C:\Containers\CSTI\027 C:\Containers\NEWA\027 C:\Containers\OLN\027 C:\Containers\STK\027 C:\Containers\CST\027 C:\Containers\ORDI\027 C:\Containers\ORD\027 C:\Containers\HST\027 C:\Containers\NEWB\027	334.00GB	Database Backup Files

30	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\028 C:\Containers\WAR\028 C:\Containers\DIS\028 C:\Containers\CSTI\028 C:\Containers\NEWA\028 C:\Containers\OLN\028 C:\Containers\STK\028 C:\Containers\CST\028 C:\Containers\ORDI\028 C:\Containers\ORD\028 C:\Containers\HST\028 C:\Containers\NEWB\028	334.00GB	Database Backup Files
31	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\029 C:\Containers\WAR\029 C:\Containers\DIS\029 C:\Containers\CSTI\029 C:\Containers\NEWA\029 C:\Containers\OLN\029 C:\Containers\STK\029 C:\Containers\CST\029 C:\Containers\ORDI\029 C:\Containers\ORD\029 C:\Containers\HST\029 C:\Containers\NEWB\029	334.00GB	Database Backup Files
32	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\030 C:\Containers\WAR\030 C:\Containers\DIS\030 C:\Containers\CSTI\030 C:\Containers\NEWA\030 C:\Containers\OLN\030 C:\Containers\STK\030 C:\Containers\CST\030 C:\Containers\ORDI\030 C:\Containers\ORD\030 C:\Containers\HST\030 C:\Containers\NEWB\030	334.00GB	Database Backup Files
33	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\031 C:\Containers\WAR\031 C:\Containers\DIS\031 C:\Containers\CSTI\031 C:\Containers\NEWA\031 C:\Containers\OLN\031 C:\Containers\STK\031 C:\Containers\CST\031 C:\Containers\ORDI\031 C:\Containers\ORD\031 C:\Containers\HST\031 C:\Containers\NEWB\031	334.00GB	Database Backup Files
34	10 - 36.4GB EXP710 Enclosure	C:\Backup	300.60GB	Backup Files

Disk #	Drives	Partition	Size	Use
35	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\032 C:\Containers\WAR\032 C:\Containers\DIS\032 C:\Containers\CST\032 C:\Containers\NEWA\032 C:\Containers\OLN\032 C:\Containers\STK\032 C:\Containers\CST\032 C:\Containers\ORD\032 C:\Containers\ORD\032 C:\Containers\HST\032 C:\Containers\NEWB\032	334.00GB	Database Backup Files
36	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\033 C:\Containers\WAR\033 C:\Containers\DIS\033 C:\Containers\CST\033 C:\Containers\NEWA\033 C:\Containers\OLN\033 C:\Containers\STK\033 C:\Containers\CST\033 C:\Containers\ORD\033 C:\Containers\ORD\033 C:\Containers\HST\033 C:\Containers\NEWB\033	334.00GB	Database Backup Files
37	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\034 C:\Containers\WAR\034 C:\Containers\DIS\034 C:\Containers\CST\034 C:\Containers\NEWA\034 C:\Containers\OLN\034 C:\Containers\STK\034 C:\Containers\CST\034 C:\Containers\ORD\034 C:\Containers\ORD\034 C:\Containers\HST\034 C:\Containers\NEWB\034	334.00GB	Database Backup Files
38	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\035 C:\Containers\WAR\035 C:\Containers\DIS\035 C:\Containers\CST\035 C:\Containers\NEWA\035 C:\Containers\OLN\035 C:\Containers\STK\035 C:\Containers\CST\035 C:\Containers\ORD\035 C:\Containers\ORD\035 C:\Containers\HST\035 C:\Containers\NEWB\035	334.00GB	Database Backup Files

39	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\036 C:\Containers\WAR\036 C:\Containers\DIS\036 C:\Containers\CSTI\036 C:\Containers\NEWA\036 C:\Containers\OLN\036 C:\Containers\STK\036 C:\Containers\CST\036 C:\Containers\ORDI\036 C:\Containers\ORD\036 C:\Containers\HST\036 C:\Containers\NEWB\036	334.00GB	Database Backup Files
40	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\037 C:\Containers\WAR\037 C:\Containers\DIS\037 C:\Containers\CSTI\037 C:\Containers\NEWA\037 C:\Containers\OLN\037 C:\Containers\STK\037 C:\Containers\CST\037 C:\Containers\ORDI\037 C:\Containers\ORD\037 C:\Containers\HST\037 C:\Containers\NEWB\037	334.00GB	Database Backup Files
41	10 - 36.4GB EXP710 Enclosure	C:\Backup	300.60GB	Backup Files

Disk #	Drives	Partition	Size	Use
42	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\038 C:\Containers\WAR\038 C:\Containers\DIS\038 C:\Containers\CSTI\038 C:\Containers\NEWA\038 C:\Containers\OLN\038 C:\Containers\STK\038 C:\Containers\CST\038 C:\Containers\ORDI\038 C:\Containers\ORD\038 C:\Containers\HST\038 C:\Containers\NEWB\038	334.00GB	Database Backup Files
43	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\039 C:\Containers\WAR\039 C:\Containers\DIS\039 C:\Containers\CSTI\039 C:\Containers\NEWA\039 C:\Containers\OLN\039 C:\Containers\STK\039 C:\Containers\CST\039 C:\Containers\ORDI\039 C:\Containers\ORD\039 C:\Containers\HST\039 C:\Containers\NEWB\039	334.00GB	Database Backup Files
44	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\040 C:\Containers\WAR\040 C:\Containers\DIS\040 C:\Containers\CSTI\040 C:\Containers\NEWA\040 C:\Containers\OLN\040 C:\Containers\STK\040 C:\Containers\CST\040 C:\Containers\ORDI\040 C:\Containers\ORD\040 C:\Containers\HST\040 C:\Containers\NEWB\040	334.00GB	Database Backup Files
45	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\041 C:\Containers\WAR\041 C:\Containers\DIS\041 C:\Containers\CSTI\041 C:\Containers\NEWA\041 C:\Containers\OLN\041 C:\Containers\STK\041 C:\Containers\CST\041 C:\Containers\ORDI\041 C:\Containers\ORD\041 C:\Containers\HST\041 C:\Containers\NEWB\041	334.00GB	Database Backup Files

46	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\042 C:\Containers\WAR\042 C:\Containers\DIS\042 C:\Containers\CSTI\042 C:\Containers\NEWA\042 C:\Containers\OLN\042 C:\Containers\STK\042 C:\Containers\CST\042 C:\Containers\ORDI\042 C:\Containers\ORD\042 C:\Containers\HST\042 C:\Containers\NEWB\042	334.00GB	Database Backup Files
47	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\043 C:\Containers\WAR\043 C:\Containers\DIS\043 C:\Containers\CSTI\043 C:\Containers\NEWA\043 C:\Containers\OLN\043 C:\Containers\STK\043 C:\Containers\CST\043 C:\Containers\ORDI\043 C:\Containers\ORD\043 C:\Containers\HST\043 C:\Containers\NEWB\043	334.00GB	Database Backup Files
48	10 - 36.4GB EXP710 Enclosure	C:\Backup	300.60GB	Backup Files

Disk #	Drives	Partition	Size	Use
49	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\044 C:\Containers\WAR\044 C:\Containers\DIS\044 C:\Containers\CSTI\044 C:\Containers\NEWA\044 C:\Containers\OLN\044 C:\Containers\STK\044 C:\Containers\CST\044 C:\Containers\ORDI\044 C:\Containers\ORD\044 C:\Containers\HST\044 C:\Containers\NEWB\044	334.00GB	Database Backup Files
50	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\045 C:\Containers\WAR\045 C:\Containers\DIS\045 C:\Containers\CSTI\045 C:\Containers\NEWA\045 C:\Containers\OLN\045 C:\Containers\STK\045 C:\Containers\CST\045 C:\Containers\ORDI\045 C:\Containers\ORD\045 C:\Containers\HST\045 C:\Containers\NEWB\045	334.00GB	Database Backup Files
51	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\046 C:\Containers\WAR\046 C:\Containers\DIS\046 C:\Containers\CSTI\046 C:\Containers\NEWA\046 C:\Containers\OLN\046 C:\Containers\STK\046 C:\Containers\CST\046 C:\Containers\ORDI\046 C:\Containers\ORD\046 C:\Containers\HST\046 C:\Containers\NEWB\046	334.00GB	Database Backup Files
52	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\047 C:\Containers\WAR\047 C:\Containers\DIS\047 C:\Containers\CSTI\047 C:\Containers\NEWA\047 C:\Containers\OLN\047 C:\Containers\STK\047 C:\Containers\CST\047 C:\Containers\ORDI\047 C:\Containers\ORD\047 C:\Containers\HST\047 C:\Containers\NEWB\047	334.00GB	Database Backup Files

53	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\048 C:\Containers\WAR\048 C:\Containers\DIS\048 C:\Containers\CSTI\048 C:\Containers\NEWA\048 C:\Containers\OLN\048 C:\Containers\STK\048 C:\Containers\CST\048 C:\Containers\ORDI\048 C:\Containers\ORD\048 C:\Containers\HST\048 C:\Containers\NEWB\048	334.00GB	Database Backup Files
54	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\049 C:\Containers\WAR\049 C:\Containers\DIS\049 C:\Containers\CSTI\049 C:\Containers\NEWA\049 C:\Containers\OLN\049 C:\Containers\STK\049 C:\Containers\CST\049 C:\Containers\ORDI\049 C:\Containers\ORD\049 C:\Containers\HST\049 C:\Containers\NEWB\049	334.00GB	Database Backup Files
55	10 - 36.4GB EXP710 Enclosure	C:\Containers\ITM\050 C:\Containers\WAR\050 C:\Containers\DIS\050 C:\Containers\CSTI\050 C:\Containers\NEWA\050 C:\Containers\OLN\050 C:\Containers\STK\050 C:\Containers\CST\050 C:\Containers\ORDI\050 C:\Containers\ORD\050 C:\Containers\HST\050 C:\Containers\NEWB\050	334.00GB	Database Backup Files
56	1 - 36.4GB EXP400 Enclosure	C:	33.9GB	OS Drive
57	20 - 36.4GB EXP400 Enclosure	M:	339.01GB	Database Log Drive

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.

The database manager used for this testing was DB2 UDB 8.2, which is a relational database. DB2 remote stored procedures and embedded SQL statements were used. The DB2 stored procedures were invoked via SQL CALL statements. Both the client application and stored procedures were written in embedded C code.

Partitions/Replications Mapping

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

Except the Item table, all other tables were horizontally partitioned into multiple tables. The specifics of the distribution of partitioned and non-partitioned tables across the physical media are describe in Table 4-2. The database was not replicated.

60-Day Space Requirement

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

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

Clause 5: Performance Metrics and Response Time Related Items

Measured tpmC

Measured tpmC must be reported.

Measured tpmC: 250,975 tpmC

Price per tpmC: \$5.74 USD per tpmC

Response Times

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

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

Table 5-1. Response Times in Seconds

Transaction Type	Average	Maximum	90 %-tile
New-Order	0.21	1.80	0.34
Payment	0.32	3.34	0.40
Delivery	0.30	1.52	0.30
Stock Level	0.33	2.05	0.42
Order Status	0.24	1.94	0.42
Delivery (Deferred)	0.30	2.78	0.48
Menu	0.21	1.59	0.29

Keying/Think Times

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

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

Table 5-2. Keying/Think Times

Transaction Type	Average	Minimum	Maximum
New-Order	18.00 / 12.04	18.00 / 0.00	18.03/ 120.33
Payment	3.00 / 12.04	3.00 / 0.00	3.02/ 120.33
Delivery	2.00 / 5.03	2.00 / 0.00	2.02/ 50.31
Stock Level	2.00 / 5.04	2.00 / 0.00	2.02/ 50.31
Order Status	2.00 / 10.04	2.00 / 0.00	2.02/ 100.33

Response Time Frequency Distribution Curves

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

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

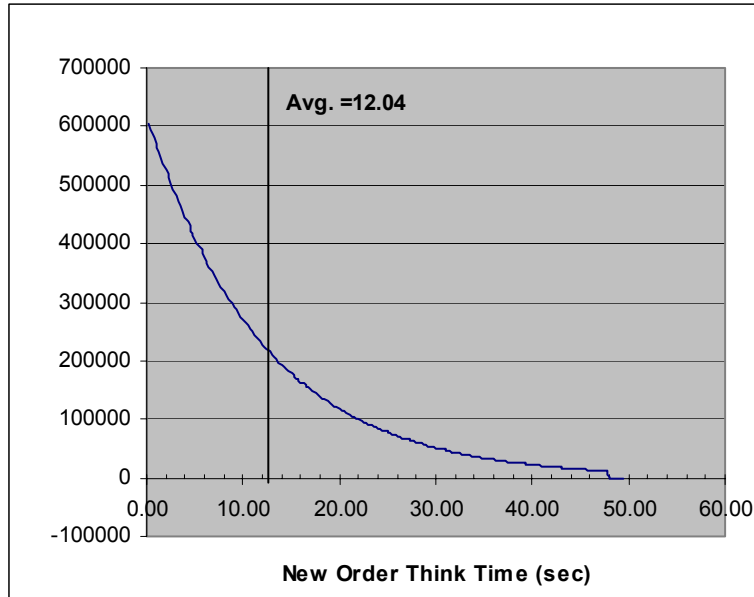


Figure 5-2. Payment Transaction - Response Time Frequency Distribution

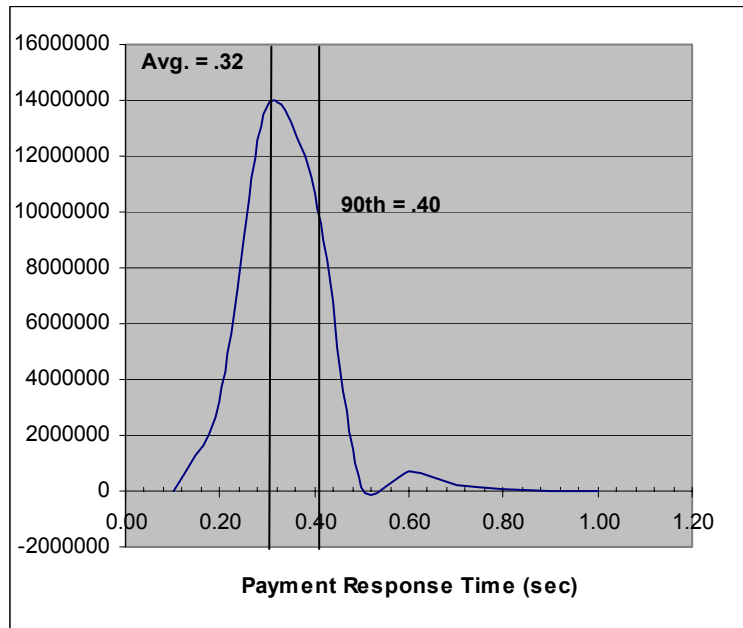


Figure 5-3. Order-Status Transaction - Response Time Frequency Distribution

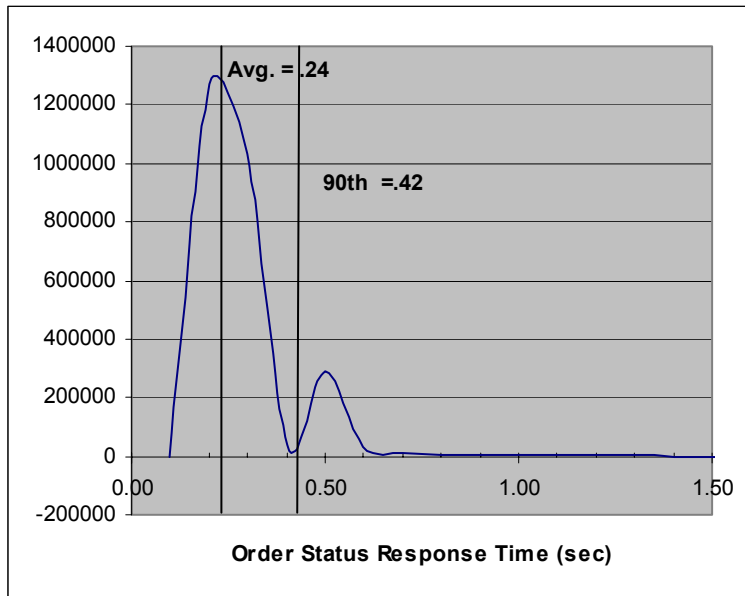
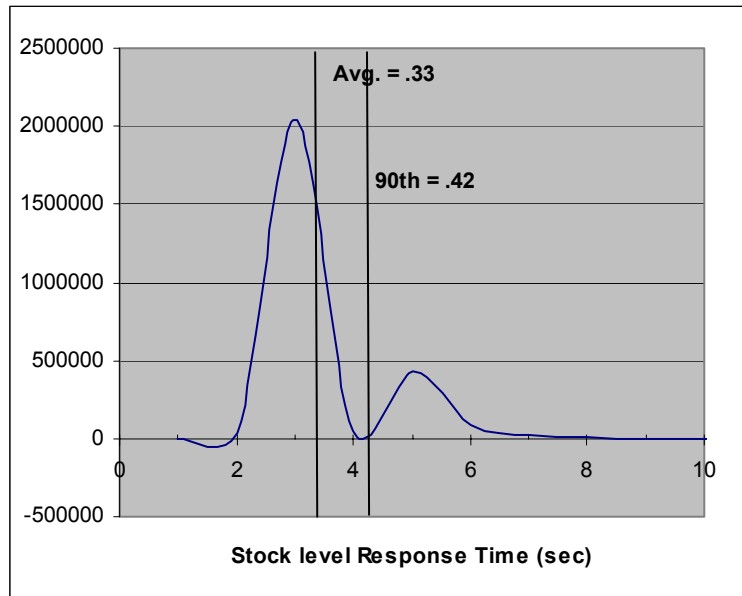


Figure 5-4. Delivery Transaction - Response Time Frequency Distribution



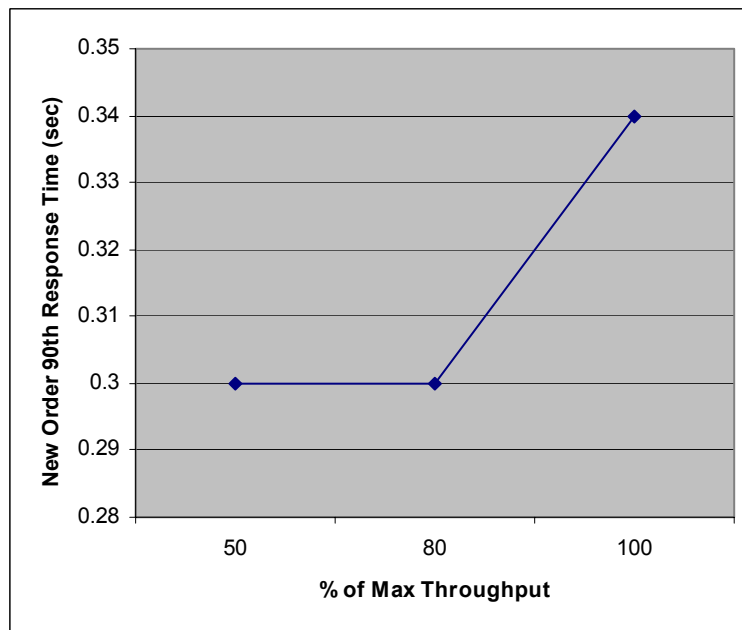
Figure 5-5. Stock-Level Transaction - Response Time Frequency Distribution



Performance Curve for Response Time vs. Throughput

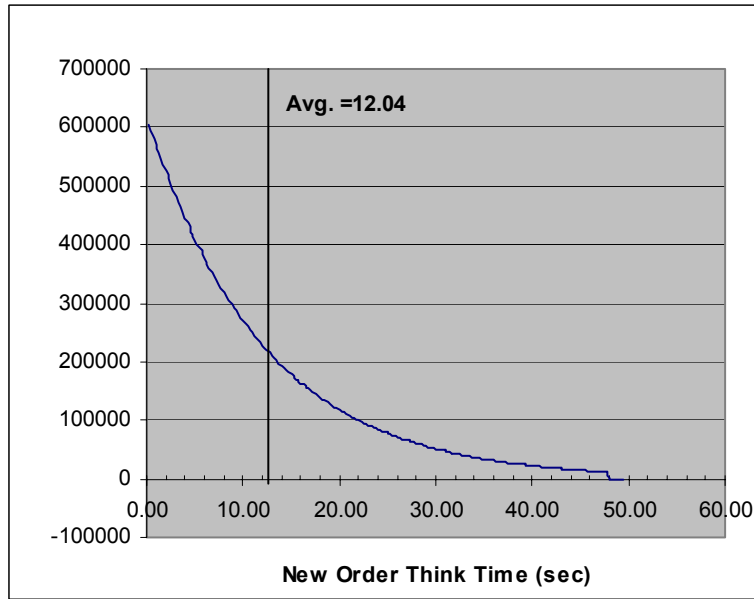
The performance curve for response time vs. throughput (see Clause 5.6.2) must be reported for the New-Order transaction.

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



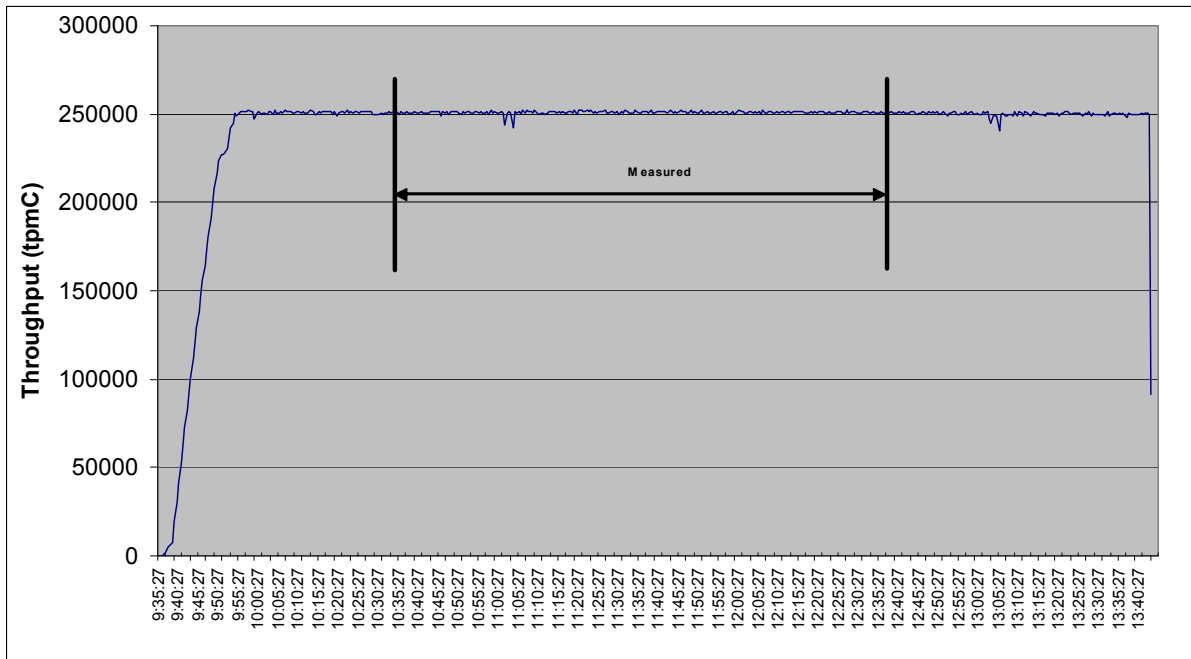
New Order Think Time Distribution

Figure 5-7. New-Order Think Time Distribution



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

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



Steady State Methodology

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

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

Work Performed during Steady State

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

Transaction Flow

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

The RTE transmissions were sent to application processes running on the client machines through Ethernet LANs. Each of the 4 (non-delivery) transactions is serviced by two individual programs, Internet Information Service 5.1 (IIS) and a Microsoft COM+ 1.0 Queued Component Server, used as the transaction manager (COM+). Both programs are running on the client system:

- The initial HTML 1.0 request is serviced by an ISAPI custom-written handler running on Internet Information Service 5.1(IIS). IIS is responsible for handling all HTML requests. The web server communicates to the COM+ server through a Microsoft COM+ api interface.
- COM+ communicates with the Server system over Ethernet and handles all database operations, using DB2 embedded SQL calls. When the COM+ server boots up, it creates a configurable amount of connections to the Server (listed in application settings). COM+ routes the transaction and balances the load according to the options defined in the Component Services GUI for the COM+ server application and settings in the Windows 2000 Registry. The configuration file and registry variables are listed in Appendix B.2. At the beginning, each TPC-C user sends a pair of HTML 1.0 requests submitting the its unique warehouse and district to the IIS ISAPI handler. Upon successful validation of the user's login, IIS the displays an HTML form that encapsulates the TPC-C transaction menu.

The transaction flow is described below:

1. The TPC-C user requests the transaction type's HTML form and proceeds to generate (fill in) a GET request with the required files for the transaction.
2. IIS accepts the filled-in GET request, parses, and validates all values entered by the user.
3. It then proceeds to transmit those values to the COM+ server through an transaction type-specific COM+ api interface.
4. The COM+ Pool Manager receives the request and first decides if there is a connection object in the pool available to service it. If so, the connection is used to send the transaction request to the Server. If no connection is available, the request will enter a COM+ internal queue and will be serviced by the next available connection.
5. Once the connection is available to be used, a COM+ pool thread receives the transaction and calls a TPC-C back-end DB2 client api to execute all database operations related to the transaction type. (All the transaction information entered on the HTML form is available in a data structure provided by the ISAPI caller.)
6. The transaction is committed and the DB2 back-end client returns control back to the COM pool thread.

7. COM pool thread returns control to the ISAPI caller. (All transaction results are inside the data structure that the ISAPI caller provided to the COM+ api in the parameter list.)
- 8 ISAPI caller returns control to the "screen application" by doing a PUT request.

Measurement Interval

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

The measurement interval was 120 minutes.

Transaction Mix

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

The RTE was given a weighted random distribution, which was not adjusted during the run. See Table 5-3.

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	0.85
Remote warehouse payment transactions	0.15
Non-Primary Key Access	
Payment transactions using C_LAST	60.00
Order-Status transactions using C_LAST	59.99
Delivery	
Delivery transactions skipped	0
Transaction Mix	
New-Order	44.96
Payment	43.02
Delivery	4.01
Stock Level	4.01
Order-Status	4.01

Number of Checkpoints

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

DB2 UDB uses a write-ahead-logging protocol to guarantee recovery. This protocol uses “soft” checkpoint to write least-recently-used database pages to disk independent of transaction commit. However, enough log information to redo/undo the change to a database pages is committed to disk before the database page itself is written. This protocol, therefore, renders checkpoint unnecessary for DB2 UDB.

For a more detailed description of the general principles of the write-ahead-logging protocol, see the IBM research paper, “ARIES: A Transaction Recovery Method Supporting Fine Granularity Locking and Partial Rollbacks Using Write-Ahead Logging,” by C. Mohan, Database Technology Institute, IBM Almaden Research Center.

<http://portal.acm.org/citation.cfm?id=128770&coll=portal&dl=ACM&CFID=10343790&CFTOKEN=42047146>

Clause 6: SUT, Driver and Communication Definition Related Items

Description of RTE

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

The RTE used is IBM-developed proprietary software. The RTE input is listed in Appendix C.

Emulated Components

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

No components were emulated.

Benchmarked and Targeted System Configuration Diagrams

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

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

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

Network Configuration

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

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

Network Bandwidth

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

The Ethernet used in the LAN connecting the clients and driver RTEs complies with the IEEE.802.3 standard. The Ethernet LAN had a bandwidth of 1Gbps. The LAN that connected the clients to the server complies with the IEEE.802.3 standard. The Ethernet LAN had a bandwidth of 1Gbps.

Operator Intervention

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

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

Clause 7: Pricing Related Items

Hardware and Software Components

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

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

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

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

Availability Date

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

The planned availability date for the 8-way x460 is June 17, 2005. The planned availability date that the 4GB Memory DIMM will be supported in and available for the x460 is November 30, 2005. DB2 UDB is generally available now. The total solution as priced will be generally available November 30, 2005.

Measured tpmC

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

- Maximum Qualified Throughput: 250,975 tpmC
- Price per tpmC: \$5.74 USD per tpmC
- Three-year cost of ownership: \$1,440,290 USD

Country-Specific Pricing

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

The configuration is priced for the United States of America.

Usage Pricing

For any usage pricing, the sponsor must disclose:

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

The component pricing based on usage is shown below:

- 1 Microsoft Windows Server 2003 Enterprise x64 Edition
- 10 Microsoft Windows 2000 Server
- 8 DB2 UDB 8.2 (based on per-processor price)

- 3-year support for hardware components (except for components for which a minimum of 2 or 10 percent spares are provided)

System Pricing

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

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

Clause 9: Audit Related Items

Auditor

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

This implementation of the TPC-C benchmark was audited by Francois Raab 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.

The TPC Benchmark C Full Disclosure Report can be obtained from www.tpc.org.

Benchmark Sponsor: Celia Schreiber
 Manager, xSeries Performance
 IBM Systems and Technology Group
 3039 Cornwallis Road
 Research Triangle Park, NC 27709

May 26, 2005

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

Platform: IBM eServer xSeries 460 c/s
 Operating system: Microsoft Windows Server 2003 Enterprise x64 Edition
 Database Manager: IBM DB2 UDB 8.2
 Transaction Manager: Microsoft COM+

The results were:

CPU's Speed	Memory	Disks	NewOrder 90% Response Time	tpmC
Server: IBM eServer xSeries 460				
8 x Xeon MP (3.33GHz)	128 GB (8 MB L3)	500 x 36.4 GB 2Gbps 21 x 36.4 GB Ultra320	0.34 Seconds	250,975.35
Ten Client: IBM eServer xSeries 226 (each with)				
2 x Xeon DP (3.4 GHz)	2.5 GB (2 MB cache)	1 x 36.4 GB U320 SCSI	n/a	n/a

In my opinion, these performance results were produced in compliance with the TPC requirements for the benchmark.

The following verification items were given special attention:

- The transactions were correctly implemented
- The database records were the proper size

- The database was properly scaled and populated
- The ACID properties were met
- 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.
- At least 90% of all delivery transactions met the 80 Second completion time limit
- All 90% response times were under the specified maximums
- The measurement interval was representative of steady state conditions
- The reported measurement interval was 120 minutes
- Write-ahead-logging was active during the measurement interval
- The 60 day storage requirement was correctly computed
- The system pricing was verified for major components and maintenance

Additional Audit Notes:

None.

Respectfully Yours,

A handwritten signature in black ink, appearing to read "François Raab". The signature is fluid and cursive, with a long horizontal stroke extending to the right.

François Raab, President

Appendix A: Client Server Code

A.1 Client/Terminal Handler Code

makefile.config

```
#####  
#####  
## Licensed Materials - Property of IBM  
##  
## Governed under the terms of the International  
## License Agreement for Non-Warranted Sample Code.  
##  
## (C) COPYRIGHT International Business Machines Corp. 1996 - 2005  
## All Rights Reserved.  
##  
## US Government Users Restricted Rights - Use, duplication or  
## disclosure restricted by GSA ADP Schedule Contract with IBM Corp.  
#####  
#####  
  
#  
# Makefile.config - NT/Win2000 Makefile Configuration  
#  
  
# Make Configuration (MSVC)  
MAKE=nmake.exe  
  
# Compiler Configuration (MSVC).  
# CFLAGS_DEBUG may be set to "-Zi -Od", "-DDEBUGIT" "-Zi -Od  
-DDEBUGIT" or left blank  
CC=cl.exe  
CFLAGS_OS=-DSQLWINT -MT -DWIN32 -J -Zp8 -DREG_KIT_METHOD  
CFLAGS_OUT=/Fo  
CFLAGS_DEBUG=  
  
# Linker Configuration (MSVC)  
LD_EXEC=link.exe  
LD_STORP=link.exe  
LDFLAGS_EXEC=  
LDFLAGS_SHLIB=/DLL  
LDFLAGS_STORP=$(LDFLAGS_SHLIB) /DEF:rpctpc.def  
LDFLAGS_LIB=/LIBPATH:$(TPCC_SQLLIB)\lib /LIBPATH:"C:\Program  
Files\Microsoft Visual Studio\VC98\Lib" db2api.lib winmm.lib  
LDFLAGS_OUT=/OUT:  
  
# Library Configuration  
AR=lib.exe  
ARFLAGS=  
ARFLAGS_LIB=  
ARFLAGS_OUT=/OUT:  
  
# OS Commands  
ERASE=del /F  
ERASEDIR=rmdir /S  
MOVE=MOVE  
COPY=COPY  
  
# OS File Extensions & Path Separator  
OBJEXT=.obj  
LIBEXT=.lib  
SHLIBEXT=.dll  
BINEXT=.exe  
SLASH=\  
CMDSEP=&
```

tpccenv.bat

```
@REM  
*****  
*****  
@REM Licensed Materials - Property of IBM  
@REM  
@REM Governed under the terms of the International  
@REM License Agreement for Non-Warranted Sample Code.  
@REM  
@REM (C) COPYRIGHT International Business Machines Corp. 1996 - 2005  
@REM All Rights Reserved.  
@REM  
@REM US Government Users Restricted Rights - Use, duplication or  
@REM disclosure restricted by GSA ADP Schedule Contract with IBM Corp.  
@REM  
*****  
*****  
@REM  
@REM tpccenv.bat - Windows Environment Setup  
@REM  
  
@REM The Kit Version  
set TPCC_VERSION=CK041012  
  
@REM The DB2 Instance Name (for DB2)  
set DB2INSTANCE=DB2  
  
@REM The OS being used (i.e. "UNIX", "WINDOWS")  
set PLATFORM=WINDOWS  
  
@REM The type of make command and slash used by the OS  
@REM (i.e. UNIX - "/", WINDOWS - "\")  
@REM These are referenced all over the kit.  
set SLASH=  
set MAKE=nmake  
  
set TPCC_SPTYPE=SPGENERAL  
  
set DB2VERSION=v8  
  
@REM The schema name is typically the SQL authorization ID (or username).  
@REM This is required for runstats and EEE.  
set TPCC_SCHEMA=%USERNAME%  
  
@REM DB2 EE/EEE Configuration  
set DB2EDITION=EE  
set DB2NODE=0  
set DB2NODES=1  
  
set HOME=C:  
set TPCC_DBNAME=TPCC  
set TPCC_ROOT=c:\tpcckit\tpc-c.ibm  
set TPCC_SQLLIB=c:\SQLLIB  
set TPCC_RUNDATA=c:\tpcckit\tpccdata  
  
set TPCC_DEBUGDIR=c:\temp  
  
@REM Specifies where stored procedures should be placed and if they should  
@REM be fenced.  
set TPCC_SPDIR=%TPCC_SQLLIB%\function  
set TPCC_FENCED=NO  
include/db2tpcc.h  
/*****  
*****  
** Licensed Materials - Property of IBM  
**  
** Governed under the terms of the International
```

```

** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2005
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****
***/

/*
 * db2tpcc.h - Macros and Miscellany
 */

#ifndef __DB2TPCC_H
#define __DB2TPCC_H

#include <sys/types.h>
typedef __int16 int16_t;
typedef __int32 int32_t;
typedef __int64 int64_t;

#include "lval.h"

/*
*****
***** */
/* Transaction Return Codes (s_transtatus) */
/*
*****
***** */

#define INVALID_ITEM 100
#define TRAN_OK 0
#define FATAL_SQLERROR -1

/*
*****
***** */
/* Definition of Unused and Bad Items */
/*
*****
***** */
/* Define unused item ID to be 0. This allows the SUT to determine the
 * number of items in the order as required by 2.4.1.3 and 2.4.2.2 since
 * the assumption that any item with OL_I_ID = 0 is unused will be true.
 * This in turn requires that the value used for an invalid item is
 * equal to ITEMS + 1.
 */
/*
*****
***** */

#define INVALID_ITEM_ID (2 * ITEMS) + 1
#define UNUSED_ITEM_ID 0

#define MIN_WAREHOUSE 1
#define MAX_WAREHOUSE WAREHOUSES

/*
*****
***** */
/* NURand Constants */
/* C_C_LAST_RUN and C_C_LAST_LOAD must adhere to clause 2.1.6.
 */
/*
*****
***** */
#define C_C_LAST_RUN 88
#define C_C_LAST_LOAD 173
#define C_C_ID 319

```

```

#define C_OL_I_ID 3849
#define A_C_LAST 255
#define A_C_ID 1023
#define A_OL_I_ID 8191

/*****
*****/
/* Transaction Type Identifiers */
/*****
*****/

#define CLIENT_SQL 0
#define NEWORD_SQL 1
#define PAYMENT_SQL 2
#define ORDSTAT_SQL 3
#define DELIVERY_SQL 4
#define STOCKLEV_SQL 5

#define SPGENERAL_PAD 3
#define SPGENERAL_ADJUST sizeof(int16_t)

struct in_neword_struct {
    int16_t len;
    int16_t pad[SPGENERAL_PAD];
    struct in_items_struct {
        int32_t s_OL_I_ID;
        int32_t s_OL_SUPPLY_W_ID;
        int16_t s_OL_QUANTITY;
        int16_t pad1[3];
    } in_item[15];
    int64_t s_O_ENTRY_D_time; /* init by SUT */
    int32_t s_C_ID;
    int32_t s_W_ID;
    int16_t s_D_ID;
    int16_t s_O_OL_CNT; /* init by SUT */
    int16_t s_all_local;
    int16_t duplicate_items;
};

struct out_neword_struct {
    int16_t len;
    int16_t pad[SPGENERAL_PAD];
    struct items_struct {
        int32_t s_I_PRICE;
        int32_t s_OL_AMOUNT;
        int16_t s_S_QUANTITY;
        int16_t pad2;
        char s_I_NAME[25];
        char s_brand_generic;
    } item[15];
    int64_t s_O_ENTRY_D_time;
    int32_t s_W_TAX;
    int32_t s_D_TAX;
    int32_t s_C_DISCOUNT;
    int32_t s_total_amount;
    int32_t s_O_ID;
    int16_t s_O_OL_CNT;
    int16_t s_transtatus;
    int16_t deadlocks;
    char s_C_LAST[17];
    char s_C_CREDIT[3];
};

struct in_payment_struct {
    int16_t len;
    int16_t pad[SPGENERAL_PAD];
    int64_t s_H_DATE_time; /* init by SUT */
    int64_t s_H_AMOUNT;

```

```

int32_t s_W_ID;
int32_t s_C_W_ID;
int32_t s_C_ID;
int16_t s_C_D_ID;
int16_t s_D_ID;
char s_C_LAST[17];
};

struct out_payment_struct {
int16_t len;
int16_t pad[SPGENERAL_PAD];
int64_t s_H_DATE_time;
int64_t s_C_SINCE_time;
int64_t s_C_CREDIT_LIM;
int64_t s_C_BALANCE;
int32_t s_C_DISCOUNT;
int32_t s_C_ID;
int16_t s_transtatus;
int16_t deadlocks;
char s_W_STREET_1[21];
char s_W_STREET_2[21];
char s_W_CITY[21];
char s_W_STATE[3];
char s_W_ZIP[10];
char s_D_STREET_1[21];
char s_D_STREET_2[21];
char s_D_CITY[21];
char s_D_STATE[3];
char s_D_ZIP[10];
char s_C_FIRST[17];
char s_C_MIDDLE[3];
char s_C_LAST[17];
char s_C_STREET_1[21];
char s_C_STREET_2[21];
char s_C_CITY[21];
char s_C_STATE[3];
char s_C_ZIP[10];
char s_C_PHONE[17];
char s_C_CREDIT[3];
char s_C_DATA[201];
};

struct in_ordstat_struct {
int16_t len;
int16_t pad[SPGENERAL_PAD];
int32_t s_C_ID;
int32_t s_W_ID;
int16_t s_D_ID;
int16_t pad1[3];
char s_C_LAST[17];
};

struct out_ordstat_struct {
int16_t len;
int16_t pad[SPGENERAL_PAD];
int64_t s_C_BALANCE;
int64_t s_O_ENTRY_D_time;
int32_t s_C_ID;
int32_t s_O_ID;
int16_t s_O_CARRIER_ID;
int16_t s_ol_cnt;
int16_t pad1[2];
struct oitems_struct {
int64_t s_OL_DELIVERY_D_time;
int32_t s_OL_AMOUNT;
int32_t s_OL_I_ID;
int32_t s_OL_SUPPLY_W_ID;
int16_t s_OL_QUANTITY;
int16_t pad2;
} item[15];
int16_t s_transtatus;
int16_t deadlocks;
char s_C_FIRST[17];
char s_C_MIDDLE[3];
char s_C_LAST[17];
};

struct in_delivery_struct {
int16_t len;
int16_t pad[SPGENERAL_PAD];
int64_t s_O_DELIVERY_D_time; /* init by SUT */
int32_t s_W_ID;
int16_t s_O_CARRIER_ID;
};

struct out_delivery_struct {
int16_t len;
int16_t pad[SPGENERAL_PAD];
int32_t s_O_ID[10];
int16_t s_transtatus;
int16_t deadlocks;
};

struct in_stocklev_struct {
int16_t len;
int16_t pad[SPGENERAL_PAD];
int32_t s_threshold;
int32_t s_W_ID;
int16_t s_D_ID;
};

struct out_stocklev_struct {
int16_t len;
int16_t pad[SPGENERAL_PAD];
int32_t s_low_stock;
int16_t s_transtatus;
int16_t deadlocks;
};

/*
*****
***** */
/* Transaction Prototypes */
/*
*****
***** */

#ifdef __cplusplus
extern "C" {
#endif

extern int neword_sql(struct in_neword_struct*, struct out_neword_struct*);
extern int payment_sql(struct in_payment_struct*, struct out_payment_struct*);
extern int ordstat_sql(struct in_ordstat_struct*, struct out_ordstat_struct*);
extern int delivery_sql(struct in_delivery_struct*, struct out_delivery_struct*);
extern int stocklev_sql(struct in_stocklev_struct*, struct out_stocklev_struct*);

#ifdef __cplusplus
}
#endif

/*
*****
***** */
/* DB2 Connect/Disconnect & Thread Context Wrappers */

```

```

/*
*****
***** */

#ifdef __cplusplus
extern "C" {
#endif

extern int connect_to_TM(char*);
extern int connect_to_TM_auth(char*, char*, char*);
extern int disconnect_from_TM(void);

extern int create_context(void);
extern int destroy_context(void);
extern int get_context(void*);
extern int attach_context(void*);
extern int detach_context(void*);

#ifdef __cplusplus
}
#endif

#endif // __DB2TPCC_H

include/tpccapp.h

** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2005
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****

/*
* tpccapp.h - Application Macros
*/

#ifndef __TPCCAPP_H
#define __TPCCAPP_H

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <time.h>

#include "sqlenv.h"
#define daricall __stdcall

#include "sqlca.h"
#include "sqlcodes.h"

#ifdef SWAP_ENDIAN
#define SWAP_BYTE(Var) SwapEndian((void*)&Var, sizeof(Var))

/*
*****
***** */
FUNCTION: SwapEndian
PURPOSE: Swap the byte order of a structure
EXAMPLE: int I=0x12345678; SWAP_BYTE(I); I => 0x78563412;
IMPLEMENTATION: Fold Addr in half, swap header & tail by XOR op
e.g.: *a = 0x12 [ Addr + 0];

```

```

*b = 0x78 [ Addr + 4 - 0 - 1 = Addr+3];
*a ^= *b; // sets *a to 0x6A
*b ^= *a; // sets *b to 0x12
*a ^= *b; // sets *a to 0x78

Now *a => 0x78 && *b => 0x12
*****
*****/

void SwapEndian(void *Addr, int nb)
{
int i;
for (i=0; i<nb/2; i++)
{
char *a = (char*)Addr+i;
char *b = (char*)Addr+(nb-i-1);

*a ^= *b;
*b ^= *a;
*a ^= *b;
}
}
#endif //SWAP_ENDIAN

/*
*****
*****/
/* SQLCODE Macros */
/*
*****
*****/

#define DLCHK(a) \
if (sqlca.sqlcode == SQL_RC_E911) { goto a; }

#define NACOMPCHK(last) \
if (sqlca.sqlcode != SQL_RC_E1339) { last = -1; } \
else { int a = ((sqlca.sqlerrmc[4] == 0x20) ? 0 : sqlca.sqlerrmc[4]-0x30); \
int b = ((sqlca.sqlerrmc[5] == 0x20) ? 0 : sqlca.sqlerrmc[5]-0x30); \
if (b == 0) { last = a; } else { last = a * 10 + b; } \
}

#endif // __TPCCAPP_H

include/tpccdbg.h

/*
*****
***** */
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2005
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****

/*
* tpccdbg.h - Debugging Macros
*/

#ifndef __TPCCDBG_H
#define __TPCCDBG_H

#ifdef __cplusplus

```

```

extern "C" {
#endif

extern void sqlerror (int tranType, char *msg, char *file, int line,
    SQL_STRUCTURE sqlca *psqlca);

extern void new_debug (struct out_neword_struct *neword_ptr,
    struct in_neword_struct *in_neword_ptr,
    char *msg);
extern void pay_debug (struct out_payment_struct *payment_ptr,
    struct in_payment_struct *in_payment_ptr,
    char *msg);
extern void ord_debug (struct out_ordstat_struct *ordstat_ptr,
    struct in_ordstat_struct *in_ordstat_ptr,
    char *msg);
extern void del_debug (struct out_delivery_struct *delivery_ptr,
    struct in_delivery_struct *in_delivery_ptr,
    char *msg);
extern void stk_debug (struct out_stocklev_struct *stocklev_ptr,
    struct in_stocklev_struct *in_stocklev_ptr,
    char *msg);

extern void new_print (struct out_neword_struct *neword_ptr,
    struct in_neword_struct *in_neword_ptr,
    char *filename,
    char *msg);
extern void pay_print (struct out_payment_struct *payment_ptr,
    struct in_payment_struct *in_payment_ptr,
    char *filename,
    char *msg);
extern void ord_print (struct out_ordstat_struct *ordstat_ptr,
    struct in_ordstat_struct *in_ordstat_ptr,
    char *filename,
    char *msg);
extern void del_print (struct out_delivery_struct *delivery_ptr,
    struct in_delivery_struct *in_delivery_ptr,
    char *filename,
    char *msg);
extern void stk_print (struct out_stocklev_struct *stocklev_ptr,
    struct in_stocklev_struct *in_stocklev_ptr,
    char *filename,
    char *msg);

#ifdef __cplusplus
}
#endif

#endif // __TPCCDBG_H
Src.Common/Makefile

#####
#####
## Licensed Materials - Property of IBM
##
## Governed under the terms of the International
## License Agreement for Non-Warranted Sample Code.
##
## (C) COPYRIGHT International Business Machines Corp. 1996 - 2004
## All Rights Reserved.
##
## US Government Users Restricted Rights - Use, duplication or
## disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
#####
#####
#
# Makefile - Makefile for Src.Common
#
!include $(TPCC_ROOT)/Makefile.config

```

```

#
#####
#####
# Preprocessor, Compiler and Linker Flags
#
#####
BND_OPTS =          GRANT PUBLIC \
                    MESSAGES $*.bnd.msg
PRP_OPTS =          BINDFILE \
                    OPTLEVEL 1 \
                    ISOLATION RR \
                    MESSAGES $*.prep.msg \
                    LEVEL $(TPCC_VERSION) \
                    NOLINEMACRO
INCLUDES =          -I$(TPCC_SQLLIB)$(SLASH)include
                   -I$(TPCC_ROOT)$(SLASH)include
CFLAGS =            $(CFLAGS_OS) $(CFLAGS_DEBUG) $(INCLUDES) \
                   -DSQLA_NOLINES -D$(DB2EDITION)
                   -D$(DB2VERSION) \
                   -D$(TPCC_SPTYPE)
UTIL_OBJ =          tpcctdbg$(OBJEXT) tpcctcx$(OBJEXT)
#
#####
#####
# User Targets
#
#####
all:                connect $(UTIL_OBJ) disconnect
clean:              - $(ERASE) *$(OBJEXT) *.bnd *.msg tpcctcx.c
#
#####
#####
# Helper Targets
#
#####
connect:            - db2 connect to $(TPCC_DBNAME)
disconnect:        - db2 connect reset
                  - db2 terminate

rebind:            db2 bind tpcctcx.bnd $(BND_OPTS)
#
#####
#####
# Build Rules
#
#####
.SUFFIXES:
.SUFFIXES: $(OBJEXT) .c .sqc
.sqc.c:
    @echo "Prepping $*.sqc"
    -db2 prep $*.sqc $(PRP_OPTS)
    @echo "Binding $*.bnd"
    db2 bind $*.bnd $(BND_OPTS)
#
#####
#####
# Dependencies
#
#####
#####

```



```
# Source
tpccdbg$(OBJEXT): tpccdbg.c
tpccctx$(OBJEXT): tpccctx.c
tpccmisc$(OBJEXT): tpccmisc.c
# Headers
tpccdbg.c: $(TPCC_ROOT)/include/db2tpcc.h
```

Src.Common/tpccctx.sqc

```

/*****
*****
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2004
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****/
/*****/
/*
 * tpccctx.sqc - TPCC context code
 */
#include <stdlib.h>
#include <stdio.h>
#include <sqlutil.h>
#include "db2tpcc.h"
int connect_to_TM(char *in_dbname);
int connect_to_TM_auth(char *in_dbname, char *in_username, char
*in_password);
int disconnect_from_TM(void);
int create_context();
int destroy_context();
int attach_context(void*);
int detach_context(void*);
int get_context(void**);
int connect_to_TM(char *in_dbname)
{
    return connect_to_TM_auth(in_dbname, "", "");
}
int connect_to_TM_auth(char *in_dbname, char *in_username, char
*in_password)
{
    SQL_STRUCTURE sqlca sqlca;
    int ConnectSQLCODE = 0;
    EXEC SQL BEGIN DECLARE SECTION;
    char dbname[9];
    char username[129];
    char password[15];
    EXEC SQL END DECLARE SECTION;
    SQLCODE = create_context();
    if (SQLCODE != 0) { return SQLCODE; }
    strncpy(dbname, in_dbname, 8);
    if (strcmp(in_username, "") == 0)
    {
        EXEC SQL CONNECT TO :dbname IN SHARE MODE;
    } else {
        strncpy(username, in_username, 128);
        strncpy(password, in_password, 14);
        EXEC SQL CONNECT TO :dbname IN SHARE MODE USER :username
USING :password;
    }
    ConnectSQLCODE = SQLCODE;
    if (ConnectSQLCODE != 0)

```

```

{
    sqlerror( CLIENT_SQL, "CONNECT", __FILE__, __LINE__, &sqlca);
    SQLCODE = destroy_context();
    if (SQLCODE != 0) { return SQLCODE; }
    return ConnectSQLCODE;
}
return 0;
}
int disconnect_from_TM(void)
{
    SQL_STRUCTURE sqlca sqlca;
    int DisconnectSQLCODE = 0;
    EXEC SQL CONNECT RESET;
    DisconnectSQLCODE = SQLCODE;
    if (DisconnectSQLCODE != 0) {
        sqlerror( CLIENT_SQL, "DISCONNECT", __FILE__, __LINE__, &sqlca);
    }
    SQLCODE = destroy_context();

    if (SQLCODE != 0) { return SQLCODE; }
    if (DisconnectSQLCODE) {
        return DisconnectSQLCODE;
    }
    return 0;
}
int create_context(void)
{
    SQL_STRUCTURE sqlca sqlca;
    void *ctx;
    sqlcSetTypeCtx(SQL_CTX_MULTI_MANUAL);
    sqlcBeginCtx(&ctx, SQL_CTX_BEGIN_ALL, NULL, &sqlca);
    if (SQLCODE != 0) {
        sqlerror( CLIENT_SQL, "CREATE", __FILE__, __LINE__, &sqlca);
        return SQLCODE;
    }
    return 0;
}
int attach_context(void *ctx)
{
    SQL_STRUCTURE sqlca sqlca;
    sqlcAttachToCtx(ctx, NULL, &sqlca);
    if (SQLCODE != 0) {
        sqlerror( CLIENT_SQL, "ATTACH", __FILE__, __LINE__, &sqlca);
        return SQLCODE;
    }
    return 0;
}
int detach_context(void *ctx)
{
    SQL_STRUCTURE sqlca sqlca;

    sqlcDetachFromCtx(ctx, NULL, &sqlca);
    if (SQLCODE != 0) {
        sqlerror( CLIENT_SQL, "DETACH", __FILE__, __LINE__, &sqlca);
        return SQLCODE;
    }
    return 0;
}
int destroy_context(void)
{
    SQL_STRUCTURE sqlca sqlca;
    void *ctx;
    SQLCODE = get_context(&ctx);
    if (SQLCODE) { return SQLCODE; }
    sqlcEndCtx(&ctx, SQL_CTX_END_ALL, NULL, &sqlca);
    if (SQLCODE != 0) {

```

```

    sqlerror( CLIENT_SQL, "DESTROY", __FILE__, __LINE__, &sqlca);
    return SQLCODE;
}
return 0;
}
int get_context(void **ctx)
{
    SQL_STRUCTURE sqlca sqlca;
    sqlGetCurrentCtx(ctx, NULL, &sqlca);
    if (SQLCODE != 0) {
        sqlerror( CLIENT_SQL, "GETCTX", __FILE__, __LINE__, &sqlca);
        return SQLCODE;
    }
    return 0;
}

```

Src.Common/tpccdbg.c

```

/*****
*****
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2004
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****/
/*
 * tcdbg.c - Debugging Routines
 */
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <ctype.h>
#include <time.h>
#include "sqlca.h"
#include "sql.h"
#include "db2tpcc.h"
#include "tpccdbg.h"
#define DEBUG_FILENAME_SZ 128
#define DEBUG_PATH_SIZE 128
void del_print();
void new_print();
void ord_print();
void pay_print();
void stk_print();
void current_tmstamp(char *buf);
static int debugInit = 0;
static char debugPath[DEBUG_PATH_SIZE] = "";
/*-----*/
/* InitializeDebug */
/*-----*/
inline void InitializeDebug(void) {
    if (debugInit == 0) {
        char *p = getenv("TPCC_DEBUGDIR");
        if (p) {
            strncpy(debugPath, p, DEBUG_PATH_SIZE);
        } else {
            strcpy(debugPath, "C:\\temp");
        }
        strcat(debugPath, "\\");
    }
    debugInit = 1;
}

```

```

}
/*-----*/
/* sqlerror */
/*-----*/
void sqlerror(int tranType, char *msg, char *file, int line, SQL_STRUCTURE
sqlca *psqlca)
{
    FILE *err_fp = NULL;
    char err_fn[DEBUG_PATH_SIZE + DEBUG_FILENAME_SZ];
    char tranName[16];
    int j,k;
    char timeStamp[27];
    char errStr[512] = "";
    InitializeDebug();
    strncpy(err_fn, debugPath, DEBUG_PATH_SIZE);
    current_tmstamp(&timeStamp[0]);
    timeStamp[19] = (char)NULL;
    switch(tranType)
    {
        case NEWORD_SQL:
            // sprintf(err_fn, "%d.err.out", getpid());
            strcat(err_fn, "new.err.out");
            strcpy(tranName, "NEW_ORDER");
            break;
        case DELIVERY_SQL:
            // sprintf(err_fn, "%d.err.out", getpid());
            strcat(err_fn, "del.err.out");
            strcpy(tranName, "DELIVERY");
            break;
        case PAYMENT_SQL:
            // sprintf(err_fn, "%d.err.out", getpid());
            strcat(err_fn, "pay.err.out");
            strcpy(tranName, "PAYMENT");
            break;
        case ORDSTAT_SQL:
            // sprintf(err_fn, "%d.err.out", getpid());
            strcat(err_fn, "ord.err.out");
            strcpy(tranName, "ORDER_STAT");
            break;
        case STOCKLEV_SQL:
            // sprintf(err_fn, "%d.err.out", getpid());
            strcat(err_fn, "stk.err.out");
            strcpy(tranName, "STOCK_LVL");
            break;
        case 0:
            strcat(err_fn, "cli.err.out");
            strcpy(tranName, "CLIENT");
            break;
        default:
            return;
    }
    /* Generate Formatted Error Message */
    sqlaintp(errStr, 512, 78, psqlca);
    err_fp = fopen(err_fn, "a+");
    fprintf(err_fp, "-----\n");
    fprintf(err_fp, "Transaction: %s (%s)\n", tranName, msg);
    fprintf(err_fp, "FILE %s (%u)\n", file, line);
    fprintf(err_fp, "SQLCODE %d ", psqlca->sqlcode);
    fprintf(err_fp, "TIME %s\n", timeStamp);
    fprintf(err_fp, "-----\n");
    fprintf(err_fp, "%s", errStr);
    fprintf(err_fp, "-----\n");
    if (psqlca->sqlerrmc[0] != ' ' || psqlca->sqlerrmc[1] != ' ')
    {
        fprintf(err_fp, "slerrmc: ");
        for(j = 0; j < 5; j++)
        {
            for(k = 0; k < 16; k++) {

```

```

int pos = j * 16 + k;
if (pos < 70) fprintf(err_fp, "%02x ", psqlca->sqlerrmc[pos]);
else fprintf(err_fp, " ");
}
fprintf(err_fp, " |");
for(k = 0; k < 16; k++) {
int pos = j * 16 + k;
char c = ' ';
if (pos < 70) {
c = psqlca->sqlerrmc[pos];
if (!isprint(c)) c = ' ';
}
fprintf(err_fp, "%c", c);
}
fprintf(err_fp, "\n");
if (j < 4) fprintf(err_fp, "      ");
}
}
fprintf(err_fp, "sqlerrp: ");
for(j = 0; j < 8; j++)
fprintf(err_fp, "%c", psqlca->sqlerrp[j]);
fprintf(err_fp, "\n");
fprintf(err_fp, "sqlerrd: ");
for(j = 0; j < 6; j++)
fprintf(err_fp, " %d", psqlca->sqlerrd[j]);
fprintf(err_fp, "\n");
if (psqlca->sqlwarn[0] != ' ')
{
fprintf(err_fp, "sqlwarn: ");
for(j = 0; j < 8; j++)
fprintf(err_fp, "%c ", psqlca->sqlwarn[j]);
fprintf(err_fp, "\n");
}
fprintf(err_fp, "\n");
fclose(err_fp);
}

/*-----*/
/* del_debug */
/*-----*/
void del_debug (struct out_delivery_struct *delivery_ptr,
struct in_delivery_struct *in_delivery,
char *msg)
{
char debug_fn[DEBUG_PATH_SIZE + DEBUG_FILENAME_SZ];

InitializeDebug();
strncpy(debug_fn, debugPath, DEBUG_PATH_SIZE);
strcat(debug_fn, "del.debug.out");
del_print(delivery_ptr, in_delivery, debug_fn, msg);
}
/*-----*/
/* del_print */
/*-----*/
void del_print (struct out_delivery_struct *delivery_ptr,
struct in_delivery_struct *in_delivery,
char *filename,
char *msg)
{
FILE *debug_fp;
char timeStamp[27];
int j;
current_tmstamp(&timeStamp[0]);
timeStamp[19] = (char)NULL;
if ((debug_fp = fopen(filename, "a+")) == NULL)
{
return;
}
}

```

```

fprintf(debug_fp, "Delivery debug information follows %s (%s)\n",
timeStamp, msg);

fprintf(debug_fp, "\n=====
=====\\n");
fprintf(debug_fp, "in_delivery_struct {\n");
fprintf(debug_fp, "ts_W_ID = %d (%X)\n",
in_delivery->s_W_ID, in_delivery->s_W_ID);
fprintf(debug_fp, "ts_O_CARRIER_ID = %d (%X)\n",
in_delivery->s_O_CARRIER_ID, in_delivery->s_O_CARRIER_ID);
fprintf(debug_fp, "ts_O_DELIVERY_D = %lld (%lX)\n",
in_delivery->s_O_DELIVERY_D_time,
in_delivery->s_O_DELIVERY_D_time);
fprintf(debug_fp, "}\n");
fprintf(debug_fp, "out_delivery_struct {\n");
fprintf(debug_fp, "ts_transtatus = %d (%X)\n",
delivery_ptr->s_transtatus, delivery_ptr->s_transtatus);
fprintf(debug_fp, "tdeadlocks = %d (%X)\n",
delivery_ptr->deadlocks, delivery_ptr->deadlocks);
for (j = 0; j < 10; j++) {
fprintf(debug_fp, "\tts_O_ID[%d] = %d\n",
j, delivery_ptr->s_O_ID[j]);
}
fprintf(debug_fp, "\t}\n");
fclose(debug_fp);
}

/*-----*/
/* new_debug */
/*-----*/
void new_debug (struct out_neword_struct *neword_ptr,
struct in_neword_struct *in_neword,
char *msg)
{
char debug_fn[DEBUG_PATH_SIZE + DEBUG_FILENAME_SZ];

InitializeDebug();
strncpy(debug_fn, debugPath, DEBUG_PATH_SIZE);
strcat(debug_fn, "new.debug.out");
new_print(neword_ptr, in_neword, debug_fn, msg);
}
/*-----*/
/* new_print */
/*-----*/
void new_print (struct out_neword_struct *neword_ptr,
struct in_neword_struct *in_neword,
char *filename,
char *msg)
{
FILE *debug_fp;
char timeStamp[27];
int j, items;
current_tmstamp(&timeStamp[0]);
timeStamp[19] = (char)NULL;
if ((debug_fp = fopen(filename, "a+")) == NULL)
{
return;
}
fprintf(debug_fp, "New order debug information follows %s (%s)\n",
timeStamp, msg);

fprintf(debug_fp, "\n=====
=====\\n");
fprintf(debug_fp, "in_neword_struct {\n");
fprintf(debug_fp, "ts_C_ID = %d (%X)\n",
in_neword->s_C_ID, in_neword->s_C_ID);
fprintf(debug_fp, "ts_W_ID = %d (%X)\n",
in_neword->s_W_ID, in_neword->s_W_ID);
}

```

```

fprintf(debug_fp, "ts_D_ID = %d (%X)\n",
        in_neword->s_D_ID, in_neword->s_D_ID);
fprintf(debug_fp, "ts_O_OL_CNT = %d (%X)\n",
        in_neword->s_O_OL_CNT, in_neword->s_O_OL_CNT);
fprintf(debug_fp, "ts_all_local = %d (%X)\n",
        in_neword->s_all_local, in_neword->s_all_local);
fprintf(debug_fp, "ts_O_ENTRY_D = %lld (%lX)\n",
        in_neword->s_O_ENTRY_D_time, in_neword->s_O_ENTRY_D_time);
// fprintf(debug_fp, "ts_transtatus = %d (%X)\n",
//         in_neword->s_transtatus, in_neword->s_transtatus);
// fprintf(debug_fp, "tduplicate_items= %d (%X)\n",
//         in_neword->duplicate_items, in_neword->duplicate_items);
fprintf(debug_fp, "titems {\n");
items = in_neword->s_O_OL_CNT;
for (j=0; j<items; j++) {
    if(j != 0)
        fprintf(debug_fp, "\n");
    fprintf(debug_fp, "ts_OL_I_ID[%d] = %d (%X)\n",
            j, in_neword->in_item[j].s_OL_I_ID,
            in_neword->in_item[j].s_OL_I_ID);
    fprintf(debug_fp, "ts_OL_SUPPLY_W_ID[%d] = %d (%X)\n",
            j, in_neword->in_item[j].s_OL_SUPPLY_W_ID,
            in_neword->in_item[j].s_OL_SUPPLY_W_ID);
    fprintf(debug_fp, "ts_OL_QUANTITY[%d] = %d (%X)\n",
            j, in_neword->in_item[j].s_OL_QUANTITY,
            in_neword->in_item[j].s_OL_QUANTITY);
}
fprintf(debug_fp, "t}\n\n");
fprintf(debug_fp, "out_neword_struct {\n");
fprintf(debug_fp, "ts_C_LAST = %s\n",
        neword_ptr->s_C_LAST);
fprintf(debug_fp, "ts_C_CREDIT = %s\n",
        neword_ptr->s_C_CREDIT);
fprintf(debug_fp, "ts_W_TAX = %d\n",
        neword_ptr->s_W_TAX);
fprintf(debug_fp, "ts_D_TAX = %d\n",
        neword_ptr->s_D_TAX);
fprintf(debug_fp, "ts_C_DISCOUNT = %d\n",
        neword_ptr->s_C_DISCOUNT);
fprintf(debug_fp, "ts_O_ID = %d (%X)\n",
        neword_ptr->s_O_ID, neword_ptr->s_O_ID);
fprintf(debug_fp, "ts_O_OL_CNT = %d (%X)\n",
        neword_ptr->s_O_OL_CNT, neword_ptr->s_O_OL_CNT);
fprintf(debug_fp, "ts_O_ENTRY_D = %lld (%lX)\n",
        neword_ptr->s_O_ENTRY_D_time,
        neword_ptr->s_O_ENTRY_D_time);
neword_ptr->s_O_ENTRY_D_time);
fprintf(debug_fp, "ts_total_amount = %d\n",
        neword_ptr->s_total_amount);
fprintf(debug_fp, "ts_transtatus = %d (%X)\n",
        neword_ptr->s_transtatus, neword_ptr->s_transtatus);
fprintf(debug_fp, "tdeadlocks = %d (%X)\n",
        neword_ptr->deadlocks, neword_ptr->deadlocks);
// fprintf(debug_fp, "ts_W_ID = %d (%X)\n",
//         neword_ptr->s_W_ID, neword_ptr->s_W_ID);
// fprintf(debug_fp, "ts_D_ID = %d (%X)\n",
//         neword_ptr->s_D_ID, neword_ptr->s_D_ID);
// fprintf(debug_fp, "ts_all_local = %d (%X)\n",
//         neword_ptr->s_all_local, neword_ptr->s_all_local);
// fprintf(debug_fp, "tduplicate_items= %d (%X)\n",
//         neword_ptr->duplicate_items, neword_ptr->duplicate_items);
fprintf(debug_fp, "titems {\n");
items = neword_ptr->s_O_OL_CNT;
for (j=0; j<items; j++) {
    if(j != 0)
        fprintf(debug_fp, "\n");
    fprintf(debug_fp, "ts_I_NAME[%d] = %s\n",
            j, neword_ptr->item[j].s_I_NAME);

```

```

fprintf(debug_fp, "ts_I_PRICE[%d] = %d\n",
        j, neword_ptr->item[j].s_I_PRICE);
fprintf(debug_fp, "ts_OL_AMOUNT[%d] = %d\n",
        j, neword_ptr->item[j].s_OL_AMOUNT);
fprintf(debug_fp, "ts_S_QUANTITY[%d] = %d (%X)\n",
        j, neword_ptr->item[j].s_S_QUANTITY,
        neword_ptr->item[j].s_S_QUANTITY);
fprintf(debug_fp, "ts_brand_generic[%d] = %c\n",
        j, neword_ptr->item[j].s_brand_generic);
}
fprintf(debug_fp, "t}\n\n");
fclose(debug_fp);
}

/*-----*/
/* ord_debug */
/*-----*/
void ord_debug (struct out_ordstat_struct *ordstat_ptr,
                struct in_ordstat_struct *in_ordstat,
                char *msg)
{
    char debug_fn[DEBUG_PATH_SIZE + DEBUG_FILENAME_SZ];

    InitializeDebug();
    strncpy(debug_fn, debugPath, DEBUG_PATH_SIZE);
    strcat(debug_fn, "ord.debug.out");
    ord_print(ordstat_ptr, in_ordstat, debug_fn, msg);
}

/*-----*/
/* ord_print */
/*-----*/
void ord_print (struct out_ordstat_struct *ordstat_ptr,
                struct in_ordstat_struct *in_ordstat,
                char *filename,
                char *msg)
{
    FILE *debug_fp;
    char timeStamp[27];
    int j, items;
    current_tmstamp(&timeStamp[0]);
    timeStamp[19] = (char)NULL;
    if ((debug_fp = fopen(filename, "a+")) == NULL)
    {
        return;
    }
    fprintf(debug_fp, "Order status debug information follows %s (%s)\n",
            timeStamp, msg);

    fprintf(debug_fp, "\n=====");
    fprintf(debug_fp, "in_ordstat_struct {\n");
    fprintf(debug_fp, "ts_W_ID = %d (%X)\n",
            in_ordstat->s_W_ID, in_ordstat->s_W_ID);
    fprintf(debug_fp, "ts_D_ID = %d (%X)\n",
            in_ordstat->s_D_ID, in_ordstat->s_D_ID);
    fprintf(debug_fp, "ts_C_ID = %d (%X)\n",
            in_ordstat->s_C_ID, in_ordstat->s_C_ID);
    fprintf(debug_fp, "ts_C_LAST = %s\n",
            in_ordstat->s_C_LAST);

    fprintf(debug_fp, "}\n\n");
    fprintf(debug_fp, "out_ordstat_struct {\n");
    fprintf(debug_fp, "ts_C_ID = %d (%X)\n",
            ordstat_ptr->s_C_ID, ordstat_ptr->s_C_ID);
    fprintf(debug_fp, "ts_C_FIRST = %s\n",
            ordstat_ptr->s_C_FIRST);
    fprintf(debug_fp, "ts_C_MIDDLE = %s\n",

```

```

    ordstat_ptr->s_C_MIDDLE);
fprintf(debug_fp, "\ts_C_LAST = %s\n",
        ordstat_ptr->s_C_LAST);
fprintf(debug_fp, "\ts_C_BALANCE = %lld\n",
        ordstat_ptr->s_C_BALANCE);
fprintf(debug_fp, "\ts_O_ID = %d (%X)\n",
        ordstat_ptr->s_O_ID, ordstat_ptr->s_O_ID);
fprintf(debug_fp, "\ts_O_ENTRY_D = %lld (%lX)\n",
        ordstat_ptr->s_O_ENTRY_D_time, ordstat_ptr->s_O_ENTRY_D_time);
fprintf(debug_fp, "\ts_O_CARRIER_ID = %d (%X)\n",
        ordstat_ptr->s_O_CARRIER_ID, ordstat_ptr->s_O_CARRIER_ID);
fprintf(debug_fp, "\ts_ol_cnt = %d (%X)\n",
        ordstat_ptr->s_ol_cnt, ordstat_ptr->s_ol_cnt);
fprintf(debug_fp, "\ts_transtatus = %d (%X)\n",
        ordstat_ptr->s_transtatus, ordstat_ptr->s_transtatus);
fprintf(debug_fp, "\tdeadlocks = %d (%X)\n",
        ordstat_ptr->deadlocks, ordstat_ptr->deadlocks);
fprintf(debug_fp, "\titems {\n");
items = ordstat_ptr->s_ol_cnt;
for (j = 0; j < items; j++) {
    if (j != 0)
        fprintf(debug_fp, "\n");
    fprintf(debug_fp, "\ts_OL_SUPPLY_W_ID[%d] = %d (%X)\n",
            j, ordstat_ptr->item[j].s_OL_SUPPLY_W_ID,
            ordstat_ptr->item[j].s_OL_SUPPLY_W_ID);
    fprintf(debug_fp, "\ts_OL_I_ID[%d] = %d (%X)\n",
            j, ordstat_ptr->item[j].s_OL_I_ID, ordstat_ptr->item[j].s_OL_I_ID);
    fprintf(debug_fp, "\ts_OL_QUANTITY[%d] = %d (%X)\n",
            j, ordstat_ptr->item[j].s_OL_QUANTITY,
            ordstat_ptr->item[j].s_OL_QUANTITY);
    fprintf(debug_fp, "\ts_OL_AMOUNT[%d] = %d\n",
            j, ordstat_ptr->item[j].s_OL_AMOUNT);
    fprintf(debug_fp, "\ts_OL_DELIVERY_D[%d] = %lld (%lX)\n",
            j, ordstat_ptr->item[j].s_OL_DELIVERY_D_time,
            ordstat_ptr->item[j].s_OL_DELIVERY_D_time);
}
fprintf(debug_fp, "\t}\n");
fclose(debug_fp);
}

/*-----*/
/* pay_debug */
/*-----*/
void pay_debug (struct out_payment_struct *payment_ptr,
               struct in_payment_struct *in_payment,
               char *msg)
{
    char debug_fn[DEBUG_PATH_SIZE + DEBUG_FILENAME_SZ];

    InitializeDebug();
    strncpy(debug_fn, debugPath, DEBUG_PATH_SIZE);
    strcat(debug_fn, "pay.debug.out");
    pay_print(payment_ptr, in_payment, debug_fn, msg);
}

/*-----*/
/* pay_print */
/*-----*/
void pay_print (struct out_payment_struct *payment_ptr,
               struct in_payment_struct *in_payment,
               char *filename,
               char *msg)
{
    FILE *debug_fp;
    char timeStamp[27];
    current_tmstamp(&timeStamp[0]);
    timeStamp[19] = (char)NULL;

```

```

    if ((debug_fp = fopen(filename, "a+")) == NULL)
    {
        return;
    }
    fprintf(debug_fp, "Payment debug information follows %s (%s)\n",
            timeStamp, msg);

    fprintf(debug_fp, "\n=====
=====");
    fprintf(debug_fp, "in_payment_struct {\n");
    fprintf(debug_fp, "\ts_H_AMOUNT = %lld (%lX)\n",
            in_payment->s_H_AMOUNT, in_payment->s_H_AMOUNT);
    fprintf(debug_fp, "\ts_C_ID = %d (%X)\n",
            in_payment->s_C_ID, in_payment->s_C_ID);
    fprintf(debug_fp, "\ts_W_ID = %d (%X)\n",
            in_payment->s_W_ID, in_payment->s_W_ID);
    fprintf(debug_fp, "\ts_D_ID = %d (%X)\n",
            in_payment->s_D_ID, in_payment->s_D_ID);
    fprintf(debug_fp, "\ts_C_D_ID = %d (%X)\n",
            in_payment->s_C_D_ID, in_payment->s_C_D_ID);
    fprintf(debug_fp, "\ts_C_W_ID = %d (%X)\n",
            in_payment->s_C_W_ID, in_payment->s_C_W_ID);
    fprintf(debug_fp, "\ts_C_LAST = %s\n",
            in_payment->s_C_LAST);
    fprintf(debug_fp, "\ts_H_DATE = %lld (%lX)\n",
            in_payment->s_H_DATE_time, in_payment->s_H_DATE_time);
    fprintf(debug_fp, "\n}\n");
    fprintf(debug_fp, "out_payment_struct {\n");
    fprintf(debug_fp, "\ts_H_DATE = %lld (%lX)\n",
            in_payment->s_H_DATE_time, in_payment->s_H_DATE_time);
    fprintf(debug_fp, "\ts_C_CREDIT_LIM = %lld\n",
            payment_ptr->s_C_CREDIT_LIM);
    fprintf(debug_fp, "\ts_C_DISCOUNT = %d\n",
            payment_ptr->s_C_DISCOUNT);
    fprintf(debug_fp, "\ts_C_BALANCE = %lld\n",
            payment_ptr->s_C_BALANCE);
    fprintf(debug_fp, "\ts_C_ID = %d (%X)\n",
            payment_ptr->s_C_ID, payment_ptr->s_C_ID);
    fprintf(debug_fp, "\ts_W_STREET_1 = %s\n",
            payment_ptr->s_W_STREET_1);
    fprintf(debug_fp, "\ts_W_STREET_2 = %s\n",
            payment_ptr->s_W_STREET_2);
    fprintf(debug_fp, "\ts_W_CITY = %s\n",
            payment_ptr->s_W_CITY);
    fprintf(debug_fp, "\ts_W_STATE = %s\n",
            payment_ptr->s_W_STATE);
    fprintf(debug_fp, "\ts_W_ZIP = %s\n",
            payment_ptr->s_W_ZIP);
    fprintf(debug_fp, "\ts_D_STREET_1 = %s\n",
            payment_ptr->s_D_STREET_1);
    fprintf(debug_fp, "\ts_D_STREET_2 = %s\n",
            payment_ptr->s_D_STREET_2);
    fprintf(debug_fp, "\ts_D_CITY = %s\n",
            payment_ptr->s_D_CITY);
    fprintf(debug_fp, "\ts_D_STATE = %s\n",
            payment_ptr->s_D_STATE);
    fprintf(debug_fp, "\ts_D_ZIP = %s\n",
            payment_ptr->s_D_ZIP);
    fprintf(debug_fp, "\ts_C_FIRST = %s\n",
            payment_ptr->s_C_FIRST);
    fprintf(debug_fp, "\ts_C_MIDDLE = %s\n",
            payment_ptr->s_C_MIDDLE);
    fprintf(debug_fp, "\ts_C_LAST = %s\n",
            payment_ptr->s_C_LAST);
    fprintf(debug_fp, "\ts_C_STREET_1 = %s\n",
            payment_ptr->s_C_STREET_1);
    fprintf(debug_fp, "\ts_C_STREET_2 = %s\n",
            payment_ptr->s_C_STREET_2);

```

```

fprintf(debug_fp, "ts_C_CITY    = %s\n",
        payment_ptr->s_C_CITY);
fprintf(debug_fp, "ts_C_STATE   = %s\n",
        payment_ptr->s_C_STATE);
fprintf(debug_fp, "ts_C_ZIP    = %s\n",
        payment_ptr->s_C_ZIP);
fprintf(debug_fp, "ts_C_PHONE  = %s\n",
        payment_ptr->s_C_PHONE);
fprintf(debug_fp, "ts_C_SINCE  = %lld (%lX)\n",
        payment_ptr->s_C_SINCE_time, payment_ptr->s_C_SINCE_time);
fprintf(debug_fp, "ts_C_CREDIT = %s\n",
        payment_ptr->s_C_CREDIT);
fprintf(debug_fp, "ts_C_DATA   = %s\n",
        payment_ptr->s_C_DATA);
fprintf(debug_fp, "ts_transtatus = %d (%X)\n",
        payment_ptr->s_transtatus, payment_ptr->s_transtatus);
fprintf(debug_fp, "tdeadlocks  = %d (%X)\n",
        payment_ptr->deadlocks, payment_ptr->deadlocks);
fprintf(debug_fp, "\n\n");
fclose(debug_fp);
}

```

```

/*-----*/
/* stk_debug          */
/*-----*/

```

```

void stk_debug (struct out_stocklev_struct *stocklev,
               struct in_stocklev_struct *in_stocklev,
               char *msg)
{
    char debug_fn[DEBUG_PATH_SIZE + DEBUG_FILENAME_SZ];

```

```

    InitializeDebug();
    strncpy(debug_fn, debugPath, DEBUG_PATH_SIZE);
    strcat(debug_fn, "stk.debug.out");
    stk_print(stocklev, in_stocklev, debug_fn, msg);
}

```

```

/*-----*/
/* stk_print          */
/*-----*/

```

```

void stk_print (struct out_stocklev_struct *stocklev,
               struct in_stocklev_struct *in_stocklev,
               char *filename,
               char *msg)
{
    FILE *debug_fp;
    char timeStamp[27];
    current_tmstamp(&timeStamp[0]);
    timeStamp[19] = (char)NULL;
    if ((debug_fp = fopen(filename, "a+")) == NULL)
    {
        return;
    }
    fprintf(debug_fp, "Stock level debug information follows %s (%s)\n",
            timeStamp, msg);
}

```

```

fprintf(debug_fp, "\n=====
\n");
fprintf(debug_fp, "in_stocklev_struct {\n");
fprintf(debug_fp, "ts_W_ID    = %d (%X)\n",
        in_stocklev->s_W_ID, in_stocklev->s_W_ID);
fprintf(debug_fp, "ts_D_ID    = %d (%X)\n",
        in_stocklev->s_D_ID, in_stocklev->s_D_ID);
fprintf(debug_fp, "ts_threshold = %d (%X)\n",
        in_stocklev->s_threshold, in_stocklev->s_threshold);
fprintf(debug_fp, "}\n\n");
fprintf(debug_fp, "out_stocklev_struct {\n");
fprintf(debug_fp, "ts_transtatus = %d (%X)\n",
        stocklev->s_transtatus, stocklev->s_transtatus);

```

```

fprintf(debug_fp, "tdeadlocks  = %d (%X)\n",
        stocklev->deadlocks, stocklev->deadlocks);
fprintf(debug_fp, "ts_low_stock = %d (%X)\n",
        stocklev->s_low_stock, stocklev->s_low_stock);
fprintf(debug_fp, "}\n\n");
fclose(debug_fp);
}

```

```

void current_tmstamp(char *buf)
{
    time_t t = time(NULL);
    strncpy(buf, ctime(&t), 19);
}

```

Src.Cli/Makefile

```

#####
#####
## Licensed Materials - Property of IBM
##
## Governed under the terms of the International
## License Agreement for Non-Warranted Sample Code.
##
## (C) COPYRIGHT International Business Machines Corp. 1996 - 2004
## All Rights Reserved.
##
## US Government Users Restricted Rights - Use, duplication or
## disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
#####
#
# Makefile - Makefile for Src.Cli (RTE/Driver Interface)
#
!include $(TPCC_ROOT)/Makefile.config
#
#####
# Preprocessor, Compiler and Linker Flags
#
#####
BND_OPTS =      GRANT PUBLIC \
                MESSAGES $*.bnd.msg
PRP_OPTS =      BINDFILE \
                ISOLATION RR \
                EXPLAIN ALL \
                MESSAGES $*.prep.msg \
                LEVEL $(TPCC_VERSION) \
                NOLINEMACRO
INCLUDES =      -I$(TPCC_SQLLIB)/include -I$(TPCC_ROOT)/include
CFLAGS =        $(CFLAGS_OS) $(INCLUDES) $(CFLAGS_DEBUG) \
                $(UOPTS) -D$(DB2EDITION) -D$(DB2VERSION)
-D$(TPCC_SPTYPE)
OBJS =          $(TPCC_ROOT)/Src.Common/tpccmisc$(OBJEXT) \
                $(TPCC_ROOT)/Src.Common/tpccdbg$(OBJEXT) \
                $(TPCC_ROOT)/Src.Common/tpccctx$(OBJEXT) \
                tpcccli$(OBJEXT)
LIBS =          tpcccli$(LIBEXT)
#
#####
# User Targets
#
#####
all:            connect $(OBJS) plan $(LIBS) disconnect
                $(AR) $(ARFLAGS) $(ARFLAGS_OUT)tpcccli$(LIBEXT)
$(OBJS) $(ARFLAGS_LIB)
                @echo "-----"
                @echo "Please copy lval.h, db2tpcc.h, and tpcccli$(LIBEXT) to"

```

```

        @echo "a place where they can be #included and linked with the"
        @echo "RTE/driver code."
        @echo "-----"
clean:
    - $(ERASE) *.msg *.bnd *.plan *$(OBJEXT) *$(LIBEXT) tpccli.c
#
#####
#####
# Helper Targets
#
#####
#####
connect:
    - db2 connect to $(TPCC_DBNAME)
disconnect:
    - db2 connect reset
    - db2 terminate
plan:
    - db2exfmt -d $(TPCC_DBNAME) -e $(TPCC_SCHEMA) -s
$(TPCC_SCHEMA) -w -l -n TPCCLLI -g -# 0 -o TPCCLLI.exfmt.plan
    - db2expln -d $(TPCC_DBNAME) -c $(TPCC_SCHEMA) -p
TPCCCLI -s 0 -g -o TPCCLLI.expln.plan
rebind:    connect
           db2 bind tpccli.bnd $(BND_OPTS) QUERYOPT 7
#
#####
#####
# Build Rules
#
#####
#####
.SUFFIXES:
.SUFFIXES: $(OBJEXT) .c .sqc
tpccli.c:
    @echo "Prepping $*.sqc"
    -db2 prep $*.sqc $(PRP_OPTS) ISOLATION RR
    @echo "Binding $*.bnd"
    db2 bind $*.bnd $(BND_OPTS) QUERYOPT 7
#
#####
#####
# Dependencies
#
#####
#####
# Client Library:
tpccli$(LIBEXT):  $(OBJS)
# Source
tpcc_all_sql$(OBJEXT):          tpcc_all_sql.c
# Headers
tpcc_all_sql.c:                  $(TPCC_ROOT)/include/db2tpcc.h
$(TPCC_ROOT)/include/lval.h

```

Src.Cli/tpccli.sqc

```

/*****
*****
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2005
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

```

```

*****
*****/
/*
 * tpccli.sqc - Client/Server code for TPCC
 */

#include <stdlib.h>
#include <errno.h>
#include "db2tpcc.h"
#include "tpccapp.h"
#include "tpccdbg.h"

#include "sqlca.h"
#include "sql.h"

// -----
// New Order CLIENT
// -----

static int itemComparison ( const void * a , const void * b )
{
    struct in_items_struct * one = (struct in_items_struct *) a ;
    struct in_items_struct * two = (struct in_items_struct *) b ;

    // If diff item id then sort on that.
    // If real/quasi dup, then sort on warehouse id.

    if ( one->s_OL_I_ID != two->s_OL_I_ID )
    {
        return ( one->s_OL_I_ID - two->s_OL_I_ID ) ;
    }
    else
    {
        return ( one->s_OL_SUPPLY_W_ID - two->s_OL_SUPPLY_W_ID ) ;
    }
}

int neword_sql ( struct in_neword_struct * in_neword
                , struct out_neword_struct * neword )
{
    struct sqlca sqlca ;

    EXEC SQL BEGIN DECLARE SECTION;

    struct vc_new_in
    {
        short len;
        char data[ 270 ] ;
    } * pHostvarInput ;

    struct vc_new_out
    {
        short len;
        char data[ 662 ] ;
    } * pHostvarOutput ;

    EXEC SQL END DECLARE SECTION;

    int clientRc = TRAN_OK ;

    int itemIndex = 0 ;
    int actualItemIndex = 0 ;

    /* Create Timestamp */
    in_neword->s_O_ENTRY_D_time = time(NULL) ;

    // Sort the item list

```

```

in_neword->s_all_local = 1 ;

for ( itemIndex = 0 ;
    itemIndex < 15 && in_neword->in_item[ itemIndex ].s_OL_I_ID !=
UNUSED_ITEM_ID ;
    itemIndex++
)
{
    if ( in_neword->in_item[ itemIndex ].s_OL_SUPPLY_W_ID !=
in_neword->s_W_ID )
    {
        in_neword->s_all_local = 0 ;
    }
}

in_neword->s_O_OL_CNT = itemIndex ;

// Sort the original array
qsort( in_neword->in_item, in_neword->s_O_OL_CNT
, sizeof( in_neword->in_item[ 0 ] )
, itemComparison
);

actualItemIndex = -1 ;

for ( itemIndex = 0
    ; itemIndex < in_neword->s_O_OL_CNT
    ; itemIndex++ )
{
    actualItemIndex ++ ;
    in_neword->in_item[ actualItemIndex ].s_OL_I_ID =
in_neword->in_item[ itemIndex ].s_OL_I_ID ;
    in_neword->in_item[ actualItemIndex ].s_OL_SUPPLY_W_ID =
in_neword->in_item[ itemIndex ].s_OL_SUPPLY_W_ID ;
    in_neword->in_item[ actualItemIndex ].s_OL_QUANTITY =
in_neword->in_item[ itemIndex ].s_OL_QUANTITY ;
}

in_neword->s_O_OL_CNT = actualItemIndex + 1 ;

pHostvarInput = (struct vc_new_in *) in_neword ;
pHostvarInput->len = sizeof(struct in_neword_struct) -
SPGENERAL_ADJUST ;

pHostvarOutput = (struct vc_new_out *) neword ;
pHostvarOutput->len = sizeof(struct out_neword_struct) -
SPGENERAL_ADJUST ;

#ifdef DEBUGIT
new_debug(neword, in_neword, "Client before SP call");
#endif /* DEBUGIT */

#ifdef SWAP_ENDIAN
for (itemIndex=0; itemIndex<in_neword->s_O_OL_CNT; itemIndex++)
{
    SWAP_BYTE(in_neword->in_item[ itemIndex ].s_OL_I_ID);
    SWAP_BYTE(in_neword->in_item[ itemIndex ].s_OL_SUPPLY_W_ID);
    SWAP_BYTE(in_neword->in_item[ itemIndex ].s_OL_QUANTITY);
}
SWAP_BYTE(in_neword->s_O_ENTRY_D_time);
SWAP_BYTE(in_neword->s_C_ID);
SWAP_BYTE(in_neword->s_W_ID);
SWAP_BYTE(in_neword->s_D_ID);
SWAP_BYTE(in_neword->s_O_OL_CNT);
SWAP_BYTE(in_neword->s_all_local);
SWAP_BYTE(in_neword->duplicate_items);

```

```

#endif //SWAP_ENDIAN

EXEC SQL CALL news ( :*pHostvarInput, :*pHostvarOutput );

#ifdef SWAP_ENDIAN
SWAP_BYTE(in_neword->s_O_ENTRY_D_time);
SWAP_BYTE(in_neword->s_C_ID);
SWAP_BYTE(in_neword->s_W_ID);
SWAP_BYTE(in_neword->s_D_ID);
SWAP_BYTE(in_neword->s_O_OL_CNT);
SWAP_BYTE(in_neword->s_all_local);
SWAP_BYTE(in_neword->duplicate_items);
for (itemIndex=0; itemIndex<in_neword->s_O_OL_CNT; itemIndex++)
{
    SWAP_BYTE(in_neword->in_item[ itemIndex ].s_OL_I_ID);
    SWAP_BYTE(in_neword->in_item[ itemIndex ].s_OL_SUPPLY_W_ID);
    SWAP_BYTE(in_neword->in_item[ itemIndex ].s_OL_QUANTITY);
}
SWAP_BYTE(neword->s_O_ENTRY_D_time);
SWAP_BYTE(neword->s_W_TAX);
SWAP_BYTE(neword->s_D_TAX);
SWAP_BYTE(neword->s_C_DISCOUNT);
SWAP_BYTE(neword->s_total_amount);
SWAP_BYTE(neword->s_O_ID);
SWAP_BYTE(neword->s_O_OL_CNT);
SWAP_BYTE(neword->s_transtatus);
SWAP_BYTE(neword->deadlocks);
for (itemIndex=0; itemIndex<in_neword->s_O_OL_CNT; itemIndex++)
{
    SWAP_BYTE(neword->item[ itemIndex ].s_I_PRICE);
    SWAP_BYTE(neword->item[ itemIndex ].s_OL_AMOUNT);
    SWAP_BYTE(neword->item[ itemIndex ].s_S_QUANTITY);
}
#endif //SWAP_ENDIAN

if ( sqlca.sqlcode == 0 )
{
    double wtax = neword->s_W_TAX / 10000.0 ;
    double dtax = neword->s_D_TAX / 10000.0 ;
    double cdisc = neword->s_C_DISCOUNT / 10000.0 ;
    double factor = (1.0 - cdisc) * (1.0 + wtax + dtax) ;

    // Post process the item set, detecting any bad items , and set or count from
that.
    // Anything that could be deferred from the SP to the client has been.

    neword->s_total_amount = 0 ;

    for ( itemIndex = 0 ;
        itemIndex < in_neword->s_O_OL_CNT ; // from input , not output
        itemIndex++
    )
    {
        if ( neword->item[ itemIndex ].s_I_PRICE > 0 ) // A zero price signifies a
bad item
        {
            neword->item[ itemIndex ].s_OL_AMOUNT = neword->item[
itemIndex ].s_I_PRICE *
in_neword->in_item[ itemIndex
].s_OL_QUANTITY ; // reference input value

            neword->s_total_amount += neword->item[ itemIndex
].s_OL_AMOUNT ;
        }
    }

    // s_total_amount gets cast implicitly to a double to do the arithmetic,

```



```

// and then cast back to a sqlint32.
neword->s_total_amount *= factor;
}
else
{
sqlerror( NEWORD_SQL, "NEW", __FILE__, __LINE__, &sqlca );
neword->s_transtatus = FATAL_SQLERROR;
clientRc = FATAL_SQLERROR;
}

/* Update Output Structure with Timestamp */
neword->s_O_ENTRY_D_time = in_neword->s_O_ENTRY_D_time;

#ifdef DEBUGIT
new_debug(neword, in_neword, "Client after SP call");
#endif /* DEBUGIT */

if(neword->s_transtatus <= FATAL_SQLERROR)
{
new_debug(neword, in_neword, "NEW failed");
clientRc = FATAL_SQLERROR;
}

if(neword->s_transtatus == INVALID_ITEM)
{
clientRc = INVALID_ITEM;
}

return ( clientRc );
}

// -----
// Payment CLIENT
// -----

int payment_sql ( struct in_payment_struct * in_payment
, struct out_payment_struct * payment )
{
struct sqlca sqlca;

int clientRc = TRAN_OK;

EXEC SQL BEGIN DECLARE SECTION;

// Inputs

sqlint64 h_amount;
sqlint32 in_c_id;

struct s_data_type { short len; char data[ 16 ]; } c_last_input;

sqlint32 w_id;
sqlint32 c_w_id;
short d_id;
short c_d_id;
sqlint64 h_date;

// Outputs

sqlint32 c_id;

sqlint64 c_credit_lim;
sqlint32 c_discount;
sqlint64 c_balance;

char w_street_1 [ 20 ], w_street_2 [ 20 ];
char w_city [ 20 ], w_state [ 2 ], w_zip [ 9 ];

```

```

char d_street_1 [ 20 ], d_street_2 [ 20 ], d_city [ 20 ];
char d_state [ 2 ], d_zip [ 9 ], c_first [ 16 ];

char c_last [ 16 ];

char c_middle [ 2 ], c_street_1 [ 20 ];
char c_street_2 [ 20 ], c_city [ 20 ], c_state [ 2 ];
char c_zip [ 9 ], c_phone [ 16 ];

char c_credit [ 2 ];

sqlint64 c_since;

char c_data [ 200 ];
short c_data_indicator = 0;

struct c_data_prefix_c_last_type { short len; char data[ 28 ]; }
c_data_prefix_c_last;
struct c_data_prefix_c_id_type { short len; char data[ 34 ]; }
c_data_prefix_c_id;

EXEC SQL END DECLARE SECTION;

// Input redirects

#define h_amount in_payment->s_H_AMOUNT
#define in_c_id in_payment->s_C_ID

#define w_id in_payment->s_W_ID
#define d_id in_payment->s_D_ID

#define c_d_id in_payment->s_C_D_ID
#define c_w_id in_payment->s_C_W_ID
#define h_date in_payment->s_H_DATE_time

// Output redirects

#define c_credit_lim payment->s_C_CREDIT_LIM
#define c_discount payment->s_C_DISCOUNT
#define c_balance payment->s_C_BALANCE

#define c_id payment->s_C_ID
#define c_last payment->s_C_LAST

#define c_first payment->s_C_FIRST
#define c_middle payment->s_C_MIDDLE
#define c_street_1 payment->s_C_STREET_1
#define c_street_2 payment->s_C_STREET_2
#define c_city payment->s_C_CITY
#define c_state payment->s_C_STATE
#define c_zip payment->s_C_ZIP
#define c_phone payment->s_C_PHONE
#define c_credit payment->s_C_CREDIT
#define c_since payment->s_C_SINCE_time
#define c_data payment->s_C_DATA

#define w_street_1 payment->s_W_STREET_1
#define w_street_2 payment->s_W_STREET_2
#define w_city payment->s_W_CITY
#define w_state payment->s_W_STATE
#define w_zip payment->s_W_ZIP

#define d_street_1 payment->s_D_STREET_1
#define d_street_2 payment->s_D_STREET_2
#define d_city payment->s_D_CITY
#define d_state payment->s_D_STATE

```

```

#define d_zip      payment->s_D_ZIP

/* Create Timestamp */
in_payment->s_H_DATE_time = (sqlint64) time( NULL );

payment->deadlocks = -1 ;
payment->s_transtatus = TRAN_OK ;

if(c_w_id == 0) { c_w_id = w_id; }
if(c_d_id == 0) { c_d_id = d_id; }

#ifdef DEBUGIT
  pay_debug(payment, in_payment, "Client before SQL call");
#endif /* DEBUGIT */

// Create c_data_prefix strings and copy some elements from
// in -> out struct outside of retry_tran loop
// (defect 273748)

if( in_c_id == 0 )
{
  c_data_prefix_c_last.len = sprintf( c_data_prefix_c_last.data, "%2.2d
%6.6d %2.2d %6.6d %04.4d.%02.2d", c_d_id, c_w_id, d_id, w_id,
(int)(h_amount / 100), (int)(h_amount % 100) ); //@d273748mte

  // Setup the input c_last varchar
  c_last_input.len = strlen( in_payment->s_C_LAST );
  memcpy( c_last_input.data, in_payment->s_C_LAST, c_last_input.len );

  // Copy to the output structure
  memcpy( payment->s_C_LAST, in_payment->s_C_LAST, sizeof(
payment->s_C_LAST ) );
} else {

  // Copy c_id to the output structure
  c_id = in_c_id;

  c_data_prefix_c_id.len = sprintf( c_data_prefix_c_id.data, "%5.5d %2.2d
%6.6d %2.2d %6.6d %04.4d.%02.2d", c_d_id, c_w_id, d_id, w_id,
(int)(h_amount / 100), (int)(h_amount % 100) ); //@d273748mte

}

retry_tran:

payment->deadlocks ++;

if( in_c_id == 0 )
{
  EXEC SQL BEGIN COMPOUND NOT ATOMIC STATIC

  SELECT 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_ID, 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, C_DATA

  INTO :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_id, :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, :c_data :c_data_indicator

  FROM TABLE ( PAY_C_LAST( :w_id
, :d_id

```

```

, :c_w_id
, :c_d_id
, :c_last_input
, :h_date
, :h_amount
, :c_data_prefix_c_last
)

) AS T( 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_ID, 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, C_DATA
)
;

COMMIT ;

END COMPOUND ;

}
else
{
  EXEC SQL BEGIN COMPOUND NOT ATOMIC STATIC

  SELECT 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_LAST, 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, C_DATA

  INTO :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_last, :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, :c_data :c_data_indicator

  FROM TABLE ( PAY_C_ID( :w_id
, :d_id
, :c_w_id
, :c_d_id
, :in_c_id
, :h_date
, :h_amount
, :c_data_prefix_c_id
)

) AS T( 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_LAST, 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, C_DATA
)
;

COMMIT ;

```

```

        END COMPOUND ;
    }

    /* Update Output Structure with Timestamp */
    payment->s_H_DATE_time = in_payment->s_H_DATE_time ;

#ifdef DEBUGIT
    pay_debug(payment, in_payment, "Client after SQL call");
#endif /* DEBUGIT */

    if ( sqlca.sqlcode != 0 )
    {
        DLCHK( retry_tran );

        sqlerror( PAYMENT_SQL, "PAY", __FILE__, __LINE__, &sqlca );
        payment->s_transtatus = FATAL_SQLERROR ;
        clientRc = FATAL_SQLERROR ;

        pay_debug( payment, in_payment, "PAY failed" );

        EXEC SQL ROLLBACK WORK ;

        if ( sqlca.sqlcode != 0 )
        {
            sqlerror( PAYMENT_SQL, "ROLLBACK FAILED", __FILE__,
                __LINE__, &sqlca );
        }
    }

    return ( clientRc );
}

// -----
// Order Status CLIENT
// -----

int ordstat_sql ( struct in_ordstat_struct * in_ordstat
    , struct out_ordstat_struct * ordstat )
{
    struct sqlca sqlca ;

    EXEC SQL BEGIN DECLARE SECTION;

    struct vc_ord_in
    {
        short len ;
        char data[ 42 ] ;
    } * in_ord ;

    struct vc_ord_out
    {
        short len ;
        char data[ 446 ] ;
    } * out_ord ;

    EXEC SQL END DECLARE SECTION;

    int clientRc = TRAN_OK ;
    int itemIndex = 0 ;

    in_ord = (struct vc_ord_in *) in_ordstat ;
    in_ord->len = sizeof(struct in_ordstat_struct) - SPGENERAL_ADJUST ;

    out_ord = (struct vc_ord_out *) ordstat ;
    out_ord->len = sizeof(struct out_ordstat_struct) - SPGENERAL_ADJUST ;

```

```

#ifdef DEBUGIT
    ord_debug(ordstat, in_ordstat, "Client before SP call");
#endif /* DEBUGIT */

#ifdef SWAP_ENDIAN
    SWAP_BYTE(in_ordstat->s_C_ID);
    SWAP_BYTE(in_ordstat->s_W_ID);
    SWAP_BYTE(in_ordstat->s_D_ID);
#endif //SWAP_ENDIAN

    EXEC SQL CALL ords ( :*in_ord, :*out_ord ) ;

#ifdef SWAP_ENDIAN
    SWAP_BYTE(in_ordstat->s_C_ID);
    SWAP_BYTE(in_ordstat->s_W_ID);
    SWAP_BYTE(in_ordstat->s_D_ID);

    SWAP_BYTE(ordstat->s_C_BALANCE);
    SWAP_BYTE(ordstat->s_O_ENTRY_D_time);
    SWAP_BYTE(ordstat->s_C_ID);
    SWAP_BYTE(ordstat->s_O_ID);
    SWAP_BYTE(ordstat->s_O_CARRIER_ID);
    SWAP_BYTE(ordstat->s_ol_cnt);
    SWAP_BYTE(ordstat->s_transtatus);
    SWAP_BYTE(ordstat->deadlocks);
    for ( itemIndex=0; itemIndex<ordstat->s_ol_cnt; itemIndex++)
    {
        SWAP_BYTE(ordstat->item[ itemIndex ].s_OL_DELIVERY_D_time);
        SWAP_BYTE(ordstat->item[ itemIndex ].s_OL_AMOUNT);
        SWAP_BYTE(ordstat->item[ itemIndex ].s_OL_I_ID);
        SWAP_BYTE(ordstat->item[ itemIndex ].s_OL_SUPPLY_W_ID);
        SWAP_BYTE(ordstat->item[ itemIndex ].s_OL_QUANTITY);
    }
#endif //SWAP_ENDIAN

    if ( sqlca.sqlcode == 0 )
    {
        // Propogate the field we already knew into the output structure
        // 60% of the time, we already new c_last (input c_id is 0)

        if ( in_ordstat->s_C_ID == 0 )
        {
            memcpy( ordstat->s_C_LAST, in_ordstat->s_C_LAST, sizeof(
                ordstat->s_C_LAST ) );
        }
        else
        {
            ordstat->s_C_ID = in_ordstat->s_C_ID ;
        }
    }
    else
    {
        sqlerror( ORDSTAT_SQL, "ORD", __FILE__, __LINE__, &sqlca );
        ordstat->s_transtatus = FATAL_SQLERROR ;
        clientRc = FATAL_SQLERROR ;
    }

#ifdef DEBUGIT
    ord_debug(ordstat, in_ordstat, "Client after SP call");
#endif /* DEBUGIT */

    if ( ordstat->s_transtatus <= FATAL_SQLERROR )
    {
        ord_debug(ordstat, in_ordstat, "ORD failed");
        clientRc = FATAL_SQLERROR ;
    }

    return ( clientRc );

```

```

}

// -----
// Delivery CLIENT
// -----

int delivery_sql ( struct in_delivery_struct * in_delivery
                 , struct out_delivery_struct * delivery )
{
    struct sqlca sqlca ;

    EXEC SQL BEGIN DECLARE SECTION;

    struct vc_del_in
    {
        short len ;
        char data[ 22 ] ;
    } * in_del ;

    struct vc_del_out
    {
        short len ;
        char data[ 50 ] ;
    } * out_del ;

    EXEC SQL END DECLARE SECTION;

    int clientRc = TRAN_OK ;
    int orderIndex = 0 ;

    /* Create Timestamp */

    in_delivery->s_O_DELIVERY_D_time = (sqlint64) time( NULL ) ;

    in_del = (struct vc_del_in *) in_delivery ;
    in_del->len = sizeof(struct in_delivery_struct) - SPGENERAL_ADJUST;

    out_del = (struct vc_del_out *) delivery ;
    out_del->len = sizeof(struct out_delivery_struct) - SPGENERAL_ADJUST;

#ifdef DEBUGIT
    del_debug(delivery, in_delivery, "Client before SP call");
#endif /* DEBUGIT */

#ifdef SWAP_ENDIAN
    SWAP_BYTE(in_delivery->s_O_DELIVERY_D_time);
    SWAP_BYTE(in_delivery->s_W_ID);
    SWAP_BYTE(in_delivery->s_O_CARRIER_ID);
#endif //SWAP_ENDIAN

    EXEC SQL CALL dels ( :*in_del, :*out_del ) ;

#ifdef SWAP_ENDIAN
    SWAP_BYTE(in_delivery->s_O_DELIVERY_D_time);
    SWAP_BYTE(in_delivery->s_W_ID);
    SWAP_BYTE(in_delivery->s_O_CARRIER_ID);

    for (orderIndex=0; orderIndex<10; orderIndex++) {
        SWAP_BYTE(delivery->s_O_ID[ orderIndex ]);
    }
    SWAP_BYTE(delivery->s_transtatus);
    SWAP_BYTE(delivery->deadlocks);
#endif //SWAP_ENDIAN

#ifdef DEBUGIT
    del_debug(delivery, in_delivery, "Client after SP call");
#endif /* DEBUGIT */

```

```

if ( sqlca.sqlcode != 0 )
{
    sqlerror( DELIVERY_SQL, "DEL", __FILE__, __LINE__, &sqlca ) ;
    delivery->s_transtatus = FATAL_SQLERROR ;
    clientRc = FATAL_SQLERROR ;
}

if ( delivery->s_transtatus <= FATAL_SQLERROR )
{
    del_debug(delivery, in_delivery, "DEL failed");
    clientRc = FATAL_SQLERROR ;
}

return ( clientRc ) ;
}

// -----
// Stock CLIENT
// -----

#undef w_id
#undef d_id

int stocklev_sql ( struct in_stocklev_struct * in_stocklev
                 , struct out_stocklev_struct * stocklev )
{
    struct sqlca sqlca ;

    int clientRc = TRAN_OK ;

    EXEC SQL BEGIN DECLARE SECTION;

    // input

    sqlint32  threshold ;

    // output

    sqlint32  low_stock ;

    EXEC SQL END DECLARE SECTION;

#define w_id  in_stocklev->s_W_ID
#define d_id  in_stocklev->s_D_ID
#define threshold in_stocklev->s_threshold
#define low_stock stocklev->s_low_stock

    stocklev->deadlocks = -1 ;
    stocklev->s_transtatus = TRAN_OK ;

#ifdef DEBUGIT
    stk_debug(stocklev, in_stocklev, "Client before SQL call");
#endif /* DEBUGIT */

    retry_tran:

    stocklev->deadlocks ++ ;

    EXEC SQL BEGIN COMPOUND NOT ATOMIC STATIC

        SELECT COUNT( S_I_ID ) INTO :low_stock

        FROM ( SELECT DISTINCT S_I_ID

                FROM ORDER_LINE , STOCK , DISTRICT

                WHERE D_W_ID = :w_id
                  AND D_ID = :d_id

```

```

        AND OL_O_ID < d_next_o_id
        AND OL_O_ID >= ( d_next_o_id - 20 )
        AND OL_W_ID = D_W_ID
        AND OL_D_ID = D_ID
        AND S_I_ID = OL_I_ID
        AND S_W_ID = OL_W_ID
        AND S_QUANTITY < :threshold

    ) OLS

WITH CS
;

COMMIT ;

END COMPOUND ;

#ifdef DEBUGIT
    stk_debug(stocklev, in_stocklev, "Client after SQL call");
#endif /* DEBUGIT */

if ( sqlca.sqlcode != 0 )
{
    DLCHK( retry_tran );

    sqlerror( STOCKLEV_SQL, "STK", __FILE__, __LINE__, &sqlca);
    stocklev->s_transtatus = FATAL_SQLERROR ;
    clientRc = FATAL_SQLERROR ;

    stk_debug( stocklev, in_stocklev, "STK failed" );

    EXEC SQL ROLLBACK WORK ;

    if ( sqlca.sqlcode != 0 )
    {
        sqlerror( STOCKLEV_SQL, "ROLLBACK FAILED", __FILE__,
        __LINE__, &sqlca );
    }
}

return ( clientRc );
}

```

NULLDB/NULLDB.cpp

```

// NULLDB.cpp : Defines the entry point for the DLL application.
//
#include "stdafx.h"
#include "NULLDB.h"
#include "..\tpcc\sapi\tpcc.h"
BOOL WINAPIENTRY DllMain( HANDLE hModule,
        DWORD ul_reason_for_call,
        LPVOID lpReserved
        )
{
    switch (ul_reason_for_call)
    {
        case DLL_PROCESS_ATTACH:
        case DLL_THREAD_ATTACH:
        case DLL_THREAD_DETACH:
        case DLL_PROCESS_DETACH:
            break;
    }
    return TRUE;
}
// This is an example of an exported variable

```

```

NULLDB_API int dataSet = 0;
extern "C" NULLDB_API int connect_db(char *dbName,void **ctx)
{
    return OK;
}
extern "C" NULLDB_API int disconnect_db(void *ctx)
{
    return OK;
}
extern "C" NULLDB_API int do_nord(struct nord_wrapper *nord,void *ctx)
{
    nord->out_nord.s_transtatus = 0;
    if (dataSet == 0)
    {
        strcpy(nord->out_nord.s_C_LAST,"NOYOLA");
        strcpy(nord->out_nord.s_C_CREDIT,"GC");
        nord->out_nord.s_W_TAX = 1694;
        nord->out_nord.s_D_TAX = 967;
        nord->out_nord.s_C_DISCOUNT = 1024;
        nord->out_nord.s_O_ID = 3013;
        nord->out_nord.s_O_OL_CNT = 4;
        nord->out_nord.s_total_amount = 32345;
        nord->out_nord.s_O_ENTRY_D_time = 1234567890;
        strcpy(nord->out_nord.item[0].s_I_NAME,"98 Toyota Supra Turbo");
        nord->in_nord.in_item[0].s_OL_I_ID = 1;
        nord->in_nord.in_item[0].s_OL_QUANTITY = 1;
        nord->in_nord.in_item[0].s_OL_SUPPLY_W_ID = 1;
        nord->out_nord.item[0].s_I_PRICE = 42000;
        nord->out_nord.item[0].s_OL_AMOUNT = 554000;
        nord->out_nord.item[0].s_S_QUANTITY = 31;
        nord->out_nord.item[0].s_brand_generic = 'G';
        strcpy(nord->out_nord.item[1].s_I_NAME,"HKS Turbo Timer");
        nord->in_nord.in_item[1].s_OL_I_ID = 1;
        nord->in_nord.in_item[1].s_OL_QUANTITY = 1;
        nord->in_nord.in_item[1].s_OL_SUPPLY_W_ID = 1;
        nord->out_nord.item[1].s_I_PRICE = 4500;
        nord->out_nord.item[1].s_OL_AMOUNT = 438100;
        nord->out_nord.item[1].s_S_QUANTITY = 57;
        nord->out_nord.item[1].s_brand_generic = 'G';
        strcpy(nord->out_nord.item[2].s_I_NAME,"TRD GEN2 Exhaust");
        nord->in_nord.in_item[2].s_OL_I_ID = 1;
        nord->in_nord.in_item[2].s_OL_QUANTITY = 1;
        nord->in_nord.in_item[2].s_OL_SUPPLY_W_ID = 1;
        nord->out_nord.item[2].s_I_PRICE = 6734;
        nord->out_nord.item[2].s_OL_AMOUNT = 47173;
        nord->out_nord.item[2].s_S_QUANTITY = 42;
        nord->out_nord.item[2].s_brand_generic = 'G';
        strcpy(nord->out_nord.item[3].s_I_NAME,"BLITZ DUAL-SOLENOID");
        nord->in_nord.in_item[3].s_OL_I_ID = 1;
        nord->in_nord.in_item[3].s_OL_QUANTITY = 1;
        nord->in_nord.in_item[3].s_OL_SUPPLY_W_ID = 1;
        nord->out_nord.item[3].s_I_PRICE = 35000;
        nord->out_nord.item[3].s_OL_AMOUNT = 12096;
        nord->out_nord.item[3].s_S_QUANTITY = 84;
        nord->out_nord.item[3].s_brand_generic = 'G';
        dataSet = 1;
    }
}
else
{
    strcpy(nord->out_nord.s_C_LAST,"SIMPSON");
    strcpy(nord->out_nord.s_C_CREDIT,"GC");
    nord->out_nord.s_W_TAX = 913;
    nord->out_nord.s_D_TAX = 1519;
    nord->out_nord.s_C_DISCOUNT = 958;
    nord->out_nord.s_O_ID = 1410;
    nord->out_nord.s_O_OL_CNT = 9;
    nord->out_nord.s_total_amount = 12345;
    nord->out_nord.s_O_ENTRY_D_time = 1234567890;
}

```



```

strcpy(pyomt->out_paym.s_C_LAST,"Williams");
strcpy(pyomt->out_paym.s_C_STREET_1,"North Rab Road");
strcpy(pyomt->out_paym.s_C_STREET_2,"Apt 343");
strcpy(pyomt->out_paym.s_C_CITY,"La Fiera");
strcpy(pyomt->out_paym.s_C_STATE,"TX");
strcpy(pyomt->out_paym.s_C_ZIP,"785585432");
strcpy(pyomt->out_paym.s_C_PHONE,"1234567890123456");
pyomt->out_paym.s_C_SINCE_time = 0;
strcpy(pyomt->out_paym.s_C_CREDIT,"GC");
strcpy(pyomt->out_paym.s_C_DATA,"Great Ebay");
dataSet = 0;
}
return OK;
}
extern "C" NULLDB_API int do_orcls(struct orcls_wrapper *orcls,void *ctx)
{
orcls->out_orcls.s_transtatus = 0;
if (dataSet == 0)
{
orcls->out_orcls.s_C_BALANCE = 100000;
orcls->out_orcls.s_C_ID = 3;
orcls->out_orcls.s_O_ID = 1696;
orcls->out_orcls.s_O_CARRIER_ID = 9;
orcls->out_orcls.s_ol_cnt = 6;
orcls->out_orcls.s_O_ENTRY_D_time = 1234567890;
strcpy(orcls->out_orcls.s_C_FIRST,"Homer");
strcpy(orcls->out_orcls.s_C_MIDDLE,"J");
strcpy(orcls->out_orcls.s_C_LAST,"Simpson");
orcls->out_orcls.item[0].s_OL_AMOUNT = 30000;
orcls->out_orcls.item[0].s_OL_I_ID = 23492;
orcls->out_orcls.item[0].s_OL_SUPPLY_W_ID = 9;
orcls->out_orcls.item[0].s_OL_QUANTITY = 5;
orcls->out_orcls.item[0].s_OL_DELIVERY_D_time = 1234567890;
orcls->out_orcls.item[1].s_OL_AMOUNT = 12300;
orcls->out_orcls.item[1].s_OL_I_ID = 18860;
orcls->out_orcls.item[1].s_OL_SUPPLY_W_ID = 9;
orcls->out_orcls.item[1].s_OL_QUANTITY = 5;
orcls->out_orcls.item[1].s_OL_DELIVERY_D_time = 1234567890;
orcls->out_orcls.item[2].s_OL_AMOUNT = 15000;
orcls->out_orcls.item[2].s_OL_I_ID = 90488;
orcls->out_orcls.item[2].s_OL_SUPPLY_W_ID = 9;
orcls->out_orcls.item[2].s_OL_QUANTITY = 5;
orcls->out_orcls.item[2].s_OL_DELIVERY_D_time = 1234567890;
orcls->out_orcls.item[3].s_OL_AMOUNT = 25000;
orcls->out_orcls.item[3].s_OL_I_ID = 22741;
orcls->out_orcls.item[3].s_OL_SUPPLY_W_ID = 9;
orcls->out_orcls.item[3].s_OL_QUANTITY = 5;
orcls->out_orcls.item[3].s_OL_DELIVERY_D_time = 1234567890;
orcls->out_orcls.item[4].s_OL_AMOUNT = 20000;
orcls->out_orcls.item[4].s_OL_I_ID = 92952;
orcls->out_orcls.item[4].s_OL_SUPPLY_W_ID = 9;
orcls->out_orcls.item[4].s_OL_QUANTITY = 5;
orcls->out_orcls.item[4].s_OL_DELIVERY_D_time = 1234567890;
orcls->out_orcls.item[5].s_OL_AMOUNT = 2345;
orcls->out_orcls.item[5].s_OL_I_ID = 29956;
orcls->out_orcls.item[5].s_OL_SUPPLY_W_ID = 9;
orcls->out_orcls.item[5].s_OL_QUANTITY = 5;
orcls->out_orcls.item[5].s_OL_DELIVERY_D_time = 1234567890;
dataSet = 1;
}
}
else
{
orcls->out_orcls.s_C_BALANCE = 123000;
orcls->out_orcls.s_C_ID = 856;
orcls->out_orcls.s_O_ID = 418;
orcls->out_orcls.s_O_CARRIER_ID = 10;
orcls->out_orcls.s_ol_cnt = 5;
strcpy(orcls->out_orcls.s_C_FIRST,"Erick");

```

```

strcpy(orcls->out_orcls.s_C_MIDDLE,"J");
strcpy(orcls->out_orcls.s_C_LAST,"Forman");
orcls->out_orcls.s_O_ENTRY_D_time = 1234567890;
orcls->out_orcls.item[0].s_OL_AMOUNT = 12000;
orcls->out_orcls.item[0].s_OL_I_ID = 54602;
orcls->out_orcls.item[0].s_OL_SUPPLY_W_ID = 10;
orcls->out_orcls.item[0].s_OL_QUANTITY = 5;
orcls->out_orcls.item[0].s_OL_DELIVERY_D_time = 1234567890;
orcls->out_orcls.item[1].s_OL_AMOUNT = 2300;
orcls->out_orcls.item[1].s_OL_I_ID = 18860;
orcls->out_orcls.item[1].s_OL_SUPPLY_W_ID = 10;
orcls->out_orcls.item[1].s_OL_QUANTITY = 5;
orcls->out_orcls.item[1].s_OL_DELIVERY_D_time = 1234567890;
orcls->out_orcls.item[2].s_OL_AMOUNT = 56009;
orcls->out_orcls.item[2].s_OL_I_ID = 90488;
orcls->out_orcls.item[2].s_OL_SUPPLY_W_ID = 10;
orcls->out_orcls.item[2].s_OL_QUANTITY = 5;
orcls->out_orcls.item[2].s_OL_DELIVERY_D_time = 1234567890;
orcls->out_orcls.item[3].s_OL_AMOUNT = 98000;
orcls->out_orcls.item[3].s_OL_I_ID = 22741;
orcls->out_orcls.item[3].s_OL_SUPPLY_W_ID = 10;
orcls->out_orcls.item[3].s_OL_QUANTITY = 5;
orcls->out_orcls.item[3].s_OL_DELIVERY_D_time = 1234567890;
orcls->out_orcls.item[4].s_OL_AMOUNT = 25000;
orcls->out_orcls.item[4].s_OL_I_ID = 92952;
orcls->out_orcls.item[4].s_OL_SUPPLY_W_ID = 10;
orcls->out_orcls.item[4].s_OL_QUANTITY = 5;
orcls->out_orcls.item[4].s_OL_DELIVERY_D_time = 1234567890;
dataSet = 0;
}
return OK;
}
extern "C" NULLDB_API int do_dlvly(struct dlvly_wrapper *dlvly,void *ctx)
{
dlvly->out_dlvly.s_transtatus = 0;
if (dataSet == 0)
{
dataSet = 1;
for(int districtIndex=0;districtIndex <
DISTRICTS_PER_WAREHOUSE;districtIndex++)
dlvly->out_dlvly.s_O_ID[districtIndex]= 2055;
}
else
{
for(int districtIndex=0;districtIndex <
DISTRICTS_PER_WAREHOUSE;districtIndex++)
dlvly->out_dlvly.s_O_ID[districtIndex]= 2056;
dataSet = 0;
}
return OK;
}
extern "C" NULLDB_API int do_stok(struct stok_wrapper *stok,void *ctx)
{
stok->out_stok.s_transtatus = 0;
if (dataSet == 0)
{
stok->out_stok.s_low_stock = 100;
dataSet = 1;
}
}
else
{
stok->out_stok.s_low_stock = 40;
dataSet = 0;
}
return OK;
}

```

NULLDB/stdafx.h

```
// stdafx.h : include file for standard system include files,
// or project specific include files that are used frequently, but
// are changed infrequently
//
#pragma once

#define WIN32_LEAN_AND_MEAN // Exclude rarely-used
stuff from Windows headers
// Windows Header Files:
#include <windows.h>
// TODO: reference additional headers your program requires here
```

nullDB/stdafx.cpp

```
// stdafx.cpp : source file that includes just the standard includes
// nullDB.pch will be the pre-compiled header
// stdafx.obj will contain the pre-compiled type information
#include "stdafx.h"
// TODO: reference any additional headers you need in STDAFX.H
// and not in this file
```

tpcc/sapi/htmlPhraser.h

```
////////////////////////////////////
// htmlPhraser.h
////////////////////////////////////
// Class to decode a html query string
////////////////////////////////////
#pragma once
#include <memory.h>
////////////////////////////////////
// Definitions
////////////////////////////////////
#define NULL 0
#define COMMAND_ID 0
#define TERM_ID 1
#define W_ID 2
#define D_ID 3
#define C_ID 4
#define C_NAME 5
#define C_W_ID 6
#define C_D_ID 7
#define AMT_PAID 8
#define STK_THRESHOLD 9
#define CARRIER_NUM 10
#define ITEM_LIST_START 11
#define ITEM_LIST_FINISH 55
#define MAX_QUERY_ID 55
#define MAX_FIELD_LEN 256
#define MAX_FIELD_NUM 56
////////////////////////////////////
// Command Codes
////////////////////////////////////
#define NEW_ORDER_CODE 'n'
#define PAYMENT_CODE 'p'
#define ORDER_STATUS_CODE 'o'
#define DELIVERY_CODE 'd'
#define STOCK_CODE 's'
#define EXIT_CODE 'e'
```

```
#define MENU_CODE 'm'
#define COMMAND_LOGIN 0
#define COMMAND_NEW_ORDER 1
#define COMMAND_PAYMENT 2
#define COMMAND_ORDER_STATUS 3
#define COMMAND_DELIVERY 4
#define COMMAND_STOCK 5
#define COMMAND_EXIT 6
#define COMMAND_LOGIN_RESULTS 7
#define COMMAND_NEW_ORDER_RESULTS 8
#define COMMAND_PAYMENT_RESULTS 9
#define COMMAND_ORDER_STATUS_RESULTS 10
#define COMMAND_DELIVERY_RESULTS 11
#define COMMAND_STOCK_RESULTS 12
////////////////////////////////////
// Class htmlPhraser
////////////////////////////////////
class htmlPhraser
{
public:
    // Constructors / Destructor
    htmlPhraser(char *queryString);
    ~htmlPhraser()
{return;}

    // getters
public:
    int getCommandId();
    int validate(int txnType);

    char * get_TERM_ID()
{return iQueryValues[TERM_ID];}
    char * get_W_ID()
{return iQueryValues[W_ID];}
    char * get_D_ID()
{return iQueryValues[D_ID];}
    char * get_C_ID()
{return iQueryValues[C_ID];}
    char * get_C_NAME()
{return iQueryValues[C_NAME];}
    char * get_C_W_ID()
{return iQueryValues[C_W_ID];}
    char * get_C_D_ID()
{return iQueryValues[C_D_ID];}
    char * get_AMT_PAID()
{return iQueryValues[AMT_PAID];}
    char * get_STK_THRESHOLD()
{return iQueryValues[STK_THRESHOLD];}
    char * get_CARRIER_NUM()
{return iQueryValues[CARRIER_NUM];}
    char * get_ITEM_SUPP_W(int item) {return
iQueryValues[(ITEM_LIST_START + 0) + (item * 3)];}
    char * get_ITEM_ITEM_NUM(int item)
{return iQueryValues[(ITEM_LIST_START + 1) + (item * 3)];}
    char * get_ITEM_QTY(int item)
{return iQueryValues[(ITEM_LIST_START + 2) + (item * 3)];}

    // Class Functions
private:
    char convertQueryToken(char **queryString);
```



```

// Class Attributes
private:
    int      iCustomerIdFlag;
    int      iCarrierNumFlag;
    int      iStockThresholdFlag;
    char
iQueryValues[MAX_FIELD_NUM][MAX_FIELD_LEN];
};
///////////////////////////////////////////////////////////////////

```

tpccsapi/resource.h

```

//{{NO_DEPENDENCIES}}
// Microsoft Visual C++ generated include file.
// Used by tpccsapi.rc
//
#define IDS_PROJNAME 100

// Next default values for new objects
//
#ifdef APSTUDIO_INVOKED
#ifdef APSTUDIO_READONLY_SYMBOLS
#define _APS_NEXT_RESOURCE_VALUE 201
#define _APS_NEXT_COMMAND_VALUE 32768
#define _APS_NEXT_CONTROL_VALUE 201
#define _APS_NEXT_SYMED_VALUE 101
#endif
#endif

```

tpccsapi/StdAfx.h

```

// stdafx.h : include file for standard system include files,
// or project specific include files that are used frequently, but
// are changed infrequently
//
#pragma once
#define WIN32_LEAN_AND_MEAN // Exclude rarely-used
stuff from Windows headers
#define _ATL_CSTRING_EXPLICIT_CONSTRUCTORS // some
CString constructors will be explicit
#define _ATL_ALL_WARNINGS
// critical error descriptions will only be shown to the user
// in debug builds. they will always be logged to the event log
#ifdef _DEBUG
#define ATL_CRITICAL_ISAPI_ERROR_LOGONLY
#endif

#ifdef _WIN32_WINNT
#define _WIN32_WINNT 0x0403
#endif
// TODO: this disables support for registering COM objects
// exported by this project since the project contains no
// COM objects or typelib. If you wish to export COM objects
// from this project, add a typelib and remove this line
#define _ATL_NO_COM_SUPPORT
#include "resource.h"
#include <atlsrvcs.h>
#include <atlisapi.h>
#include <atlstencil.h>
// TODO: reference additional headers your program requires here

```

tpccsapi/tpcc.h

```

// Common defines and structures use internally by client code
// Not to be confused with structures actually passed in transaxtions
//

```

```

// standard includes
#ifndef _COMMON_TPCC
#define _COMMON_TPCC
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <sys/timeb.h>
#include <time.h>
#include <db2tpcc.h>
#include <iostream>
#include <fstream>
#include <process.h>
#include <ios>
///////////////////////////////////////////////////////////////////
// Defines
///////////////////////////////////////////////////////////////////
#define OK
0
#define INVALID_STATUS -1
#define ERR -1
#define INVALID_COM_STATUS -2

#define TXN_MAX_COMMANDS 55
#define MAX_TRANSACTIONS 14
#define MAX_CMD_LENGTH 100
#define INPUT_ITEMS 3
#define MAX_INT_BUFFER 15
#define NORD_ITEMS 15
#define ITEM_START 11
#define ITEM_END 55
#define MAX_ITEMS 15
#define MAX_STRING_LEN 256
#define MAX_HTML_PAGE_LEN 4096
#define MAX_HTML_HEADER_LEN 512
#define DELIVERY_THREADS_NUM 100
#define DISTRICTS_PER_WAREHOUSE 10
// Transaction Codes
///////////////////////////////////////////////////////////////////
#define TXN_LOGIN
0
#define TXN_NEW_ORDER 1
#define TXN_PAYMENT
2
#define TXN_ORDER_STATUS 3
#define TXN_DELIVERY 4
#define TXN_STOCK
5
#define TXN_EXIT
6
#define TXN_LOGIN_RESULTS 7
#define TXN_NEW_ORDER_RESULTS 8
#define TXN_PAYMENT_RESULTS 9
#define TXN_ORDER_STATUS_RESULTS 10
#define TXN_DELIVERY_RESULTS 11
#define TXN_STOCK_RESULTS
12
#define CMD_NORD
"nord"
#define CMD_PYMT
"pymt"
#define CMD_ORDS
"ords"

```

```

#define CMD_DLVY
"dlvy"
#define CMD_STOK
"stok"
#define CMD_EXIT
"exit"
#define CMD_MENU
"menu"
#define APP_NAME
"tpcc.html"
#define HEADER
"Content-Type:text/html\r\nContent-Length: %d\r\nConnection:
Keep-Alive\r\n\r\n"
// URL Commands
// URL Commands
#define CMD_TXN_ID
"00"
#define CMD_TERM_ID
"01"
#define CMD_W_ID
"02"
#define CMD_D_ID
"03"
#define CMD_C_ID
"04"
#define CMD_C_NAME
"05"
#define CMD_C_W_ID
"06"
#define CMD_C_D_ID
"07"
#define CMD_AMT_PAID "08"
#define CMD_STK_THRESHOLD "09"
#define CMD_CARRIER_NUM "10"
#define ITEM01_SUPP_W "11"
#define ITEM01_ITEM_NUM "12"
#define ITEM01_OTY "13"

#define CHAR_FILL ''
#define NUMERIC_FILL ''
#define NEGITIVE_SYMBOL '-'
#define MONEY_SYMBOL '$'
#define DECIMAL_SYMBOL '.'
#define ZERO_SYMBOL '0'
#define ZIP_DELIMITER ','
#define PHONE_DELIMITER '.'
#define DATE_DELIMITER '-'
#define TIME_DELIMITER ':'
#define DEFAULT_MONEY64_LEN 15
#define DEFAULT_MONEY32_LEN 9
#define DEFAULT_MONEY16_LEN 9
#define DEFAULT_NUMERIC64_LEN 15
#define DEFAULT_NUMERIC32_LEN 9
#define DEFAULT_NUMERIC16_LEN 9
#define DEFAULT_DECIMAL64_LEN 5
#define DEFAULT_DECIMAL32_LEN 5
#define DEFAULT_DECIMAL16_LEN 5
#define DEFAULT_DATETIME_LEN 19
#define DEFAULT_DATE_LEN 11
#define DEFAULT_TIME_LEN 8

#define DEFAULT_STRING_LEN 25
#define DEFAULT_ZIP_LEN 17
#define DEFAULT_PHONE_LEN 18
// String Field Lengths
// String Field Lengths
#define NAME_LEN 24
#define LAST_NAME_LEN 16
#define FIRST_NAME_LEN 16
#define INITIALS_LEN 2
#define CREDIT_LEN 2
#define STREET_LEN 20
#define CITY_LEN 20
#define STATE_LEN 2
#define ZIP_LEN 9
#define PHONE_LEN 16
#define DATA_LEN 200
#define ITEM_LIST 15
#define ORDER_LIST 10
// Type definitions
// Type definitions
typedef __int8 INT8b;
typedef __int16 INT16b;
typedef __int32 INT32b;
typedef __int64 INT64b;
typedef unsigned __int8 UINT8b;
typedef unsigned __int16 UINT16b;
typedef unsigned __int32 UINT32b;
typedef unsigned __int64 UINT64b;
typedef INT16b sqlint16;
typedef INT32b sqlint32;
typedef INT64b sqlint64;
typedef INT16b int16_t;
typedef INT32b int32_t;
typedef INT64b int64_t;
typedef char BYTE8b;
typedef double DOUBLE;
typedef unsigned long NATURAL;
// Date and time values
// Date and time values
#define SECONDS_IN_DAY 86400
#define SECONDS_IN_HOUR 3600
#define SECONDS_IN_MINUTE 60
#define GMT_OFFSET 5
#define DAYS_IN_YEAR 365
#define YEARS_IN_LEAP 4
#define START_YEAR 1970
#define MONTHS_IN_YEAR 12
// Error codes
// Error codes
#define ERR_INVALID_TXN_TYPE -1
#define ERR_MISSING_W_ID -2
#define ERR_NON_NUMERIC_W_ID -3
#define ERR_MISSING_D_ID -4
#define ERR_NON_NUMERIC_D_ID -5
#define ERR_MISSING_C_ID -6
#define ERR_NON_NUMERIC_C_ID -7
#define ERR_MISSING_SUPP_W -8
#define ERR_NON_NUMERIC_SUPP_W -9
#define ERR_MISSING_ITEM_NUM -10
#define ERR_NON_NUMERIC_ITEM_NUM -11
#define ERR_MISSING_ITEM_OTY -12
#define ERR_NON_NUMERIC_ITEM_QTY -13

```

#define ERR_MISSING_CLAST_NAME	-14	////////////////////////////////////	
#define ERR_NON_NUMERIC_CUST_W_ID	-15	// Registry Definitions	
#define ERR_NON_NUMERIC_CUST_D_ID	-16	////////////////////////////////////	
#define ERR_MISSING_AMOUNT_PAID	-17	#define	REGISTRY_SUB_KEY "SOFTWARE\TPCC"
#define ERR_NON_NUMERIC_AMOUNT_PAID	-18	#define	DELIVERY_THREADS
#define ERR_INVALID_D_ID	"ERROR:	"dlvyThreads"	
Invalid District ID. Try Again."		#define	DELIVERY_QUEUE_LEN
#define ERR_INVALID_W_ID	"ERROR:	"dlvyQueueLen"	
Invalid Warehouse ID. Try Again."		#define	DELIVERY_LOG_PATH
#define ERR_INVALID_C_ID	"ERROR: Invalid	"dlvyLogPath"	
Customer ID. Try Again."		#define	ERROR_LOG_FILE
#define ERR_INVALID_SUPPLY_W_ID	"ERROR: Invalid Item	"errorLogFile"	
Supply Warehouse. Try Again."		#define	HTML_TRACE_LOG_FILE
#define ERR_INVALID_ITEM_NUM	"ERROR: Invalid Item	"htmlTraceLogFile"	
Number. Try Again."		#define	DB_NAME
#define ERR_INVALID_ITEM_OTY	"ERROR: Invalid Item	"dbName"	
Qty. Try Again."		#define	NULL_DB
#define ERR_MISSING_C_ID_OR_CLAST	"ERROR: Must Enter	"nullDB"	
Customer Id or Customer Last Name. Try Again."		#define	COM_NULL_DB
#define ERR_INVALID_PAYMENT_AMOUNT	"ERROR: Invalid	"commullDB"	
Payment Amount. Try Again."		#define	CLIENT_NULL_DB
#define ERR_INVALID_CARRIER	"ERROR:	"clientNullDB"	
Invalid Carrier Number. Try Again."		#define	NUM_USERS
#define ERR_INVALID_THRESHOLD	"ERROR: Invalid	"numUsers"	
Threshold. Try Again."		#define	DB_TYPE
#define ERR_INVALID_C_D_ID	"ERROR:	"dbType"	
Invalid Customer District Id. Try Again."		#define	TXN_MONITOR
#define ERR_INVALID_C_W_ID	"ERROR:	"txn_server"	
Invalid Customer Warehouse Id. Try Again."		#define	COMM_POOL
#define ERR_TERMINAL_FULL	"ERROR:	"comm_pool"	
Terminal can not support user. Terminal full."		#define	HTML_TRACE
#define ERR_C_ID_OR_CLAST_ONLY	"ERROR: Either	"htmlTrace"	
customer id or customer last name can be specified."		#define	ISAPI_TRACE
#define ERR_UNABLE_TO_OPEN_REG		"isapi_trace"	
-50		#define	DEFAULT_DLVY_THREADS 1
#define ERR_DLVY_THREAD_FAILED		#define	DEFAULT_DLVY_QUEUE_LEN
-51		10	
#define ERR_DLVY_SEMAPHORE_INIT_FAILED	-52	#define	DEFAULT_DLVY_LOG_PATH
#define ERR_DLVY_EVENT_INIT_FAILED	-53	"c:\inetpub\wwwroot\tpcc\dlvy"	
#define ERR_DLVY_QUEUE_EATING_TAIL	-54	#define	DEFAULT_ERROR_LOG_FILE
#define ERR_INVALID_USERNAME	-70	"c:\inetpub\wwwroot\tpcc\errorLog.txt"	
#define ERR_INVALID_PASSWORD	-71	#define	DEFAULT_HTML_TRACE_LOG_FILE
#define ERR_INVALID_DB_NAME	-72	"c:\inetpub\wwwroot\tpcc\htmlTrace.txt"	
#define ERR_INVALID_REGISTRY_KEY	-73	#define	DEFAULT_NUM_USERS
#define ERR_DB2_DLL_NOT_LOADED	-74	10000	
#define ERR_ORACLE_DLL_NOT_LOADED	-75	#define	DEFAULT_DB_NAME
#define ERR_CONNECT_ADDRESS_NOT_FOUND -76		"tpcc"	
#define ERR_NORD_ADDRESS_NOT_FOUND	-77	////////////////////////////////////	
#define ERR_PYMT_ADDRESS_NOT_FOUND	-78	// Structure defines	
#define ERR_ORDS_ADDRESS_NOT_FOUND	-79	////////////////////////////////////	
#define ERR_DLVY_ADDRESS_NOT_FOUND	-80	struct nord_wrapper {	
#define ERR_STOK_ADDRESS_NOT_FOUND	-81	struct in_neword_struct in_nord;	
#define ERR_NULL_DLL_NOT_LOADED	-82	struct out_neword_struct out_nord;	
#define ERR_UNKNOWN_DB	-83	};	
#define ERR_DISCONNECT_ADDRESS_NOT_FOUND -84		struct paym_wrapper {	
#define ERR_SAVING_CONTEXT	-90	struct in_payment_struct in_paym;	
#define ERR_DETACHING_CONTEXT	-91	struct out_payment_struct out_paym;	
#define ERR_ATTACHING_CONTEXT	-92	};	
#define ERR_HANDLE_IN_USE	-93	struct ords_wrapper {	
#define ERR_CONNECT_TO_TM_FAILED	-99	struct in_ordstat_struct in_ords;	
#define ERR_DLVY_LOG_OPEN_FAILED	-100	struct out_ordstat_struct out_ords;	
#define ERR_DLVY_QUEUE_FULL	-101	};	
		struct dlvy_wrapper {	
		struct in_delivery_struct in_dlvly;	
		struct out_delivery_struct out_dlvly;	
		};	
		struct stok_wrapper {	
		struct in_stocklev_struct in_stok;	

```

        struct out_stocklev_struct out_stok;
};
typedef struct
{
    int    year;
    int    month;
    int    day;
    int    hour;
    int    minute;
    int    second;
} datetime;
struct NEWORDERDATA
{
    struct in_items_struct {
        int s_OL_I_ID;
        int s_OL_SUPPLY_W_ID;
        short s_OL_QUANTITY;
    } in_item[15];
    long long in_s_O_ENTRY_D_time; /* init by SUT */
    int in_s_C_ID;
    int in_s_W_ID;
    short in_s_D_ID;
    short in_s_O_OL_CNT; /* init by SUT */
    short in_s_all_local;
    short in_duplicate_items;
    struct out_items_struct {
        double s_I_PRICE;
        double s_OL_AMOUNT;
        short s_S_QUANTITY;
        char s_I_NAME[25];
        char s_brand_generic;
    } out_item[15];
    long long out_s_O_ENTRY_D_time;
    double out_s_W_TAX;
    double out_s_D_TAX;
    double out_s_C_DISCOUNT;
    double out_s_total_amount;
    int out_s_O_ID;
    short out_s_O_OL_CNT;
    short out_s_transtatus;
    short out_deadlocks;
    char out_s_C_LAST[17];
    char out_s_C_CREDIT[3];
};
struct PAYMENTDATA
{
    long long in_s_H_DATE_time;
    double in_s_H_AMOUNT;
    int in_s_W_ID;
    int in_s_C_W_ID;
    int in_s_C_ID;
    short in_s_C_D_ID;
    short in_s_D_ID;
    char in_s_C_LAST[17];
    long long out_s_H_DATE_time;
    long long out_s_C_SINCE_time;
    double out_s_C_CREDIT_LIM;
    double out_s_C_BALANCE;
    double out_s_C_DISCOUNT;
    int out_s_C_ID;
    short out_s_transtatus;
    short out_deadlocks;
    char out_s_W_STREET_1[21];
    char out_s_W_STREET_2[21];
    char out_s_W_CITY[21];
    char out_s_W_STATE[3];
    char out_s_W_ZIP[10];
    char out_s_D_STREET_1[21];

```

```

    char out_s_D_STREET_2[21];
    char out_s_D_CITY[21];
    char out_s_D_STATE[3];
    char out_s_D_ZIP[10];
    char out_s_C_FIRST[17];
    char out_s_C_MIDDLE[3];
    char out_s_C_LAST[17];
    char out_s_C_STREET_1[21];
    char out_s_C_STREET_2[21];
    char out_s_C_CITY[21];
    char out_s_C_STATE[3];
    char out_s_C_ZIP[10];
    char out_s_C_PHONE[17];
    char out_s_C_CREDIT[3];
    char out_s_C_DATA[201];
};
struct ORDERSTATUSDATA
{
    int in_s_C_ID;
    int in_s_W_ID;
    short in_s_D_ID;
    char in_s_C_LAST[17];

    double out_s_C_BALANCE;
    long long out_s_O_ENTRY_D_time;
    int out_s_C_ID;
    int out_s_O_ID;
    short out_s_O_CARRIER_ID;
    short out_s_ol_cnt;
    struct out_oitems_struct {
        long long s_OL_DELIVERY_D_time;
        double s_OL_AMOUNT;
        int s_OL_I_ID;
        int s_OL_SUPPLY_W_ID;
        short s_OL_QUANTITY;
    } out_item[15];
    short out_s_transtatus;
    short out_deadlocks;
    char out_s_C_FIRST[17];
    char out_s_C_MIDDLE[3];
    char out_s_C_LAST[17];
};
struct DELIVERYDATA
{
    long long in_s_O_DELIVERY_D_time;
    int in_s_W_ID;
    short in_s_O_CARRIER_ID;
    int out_s_O_ID[10];
    short out_s_transtatus;
    short outdeadlocks;
};
struct STOCKLEVELDATA
{
    int in_s_threshold;
    int in_s_W_ID;
    short in_s_D_ID;
    int out_s_low_stock;
    short out_s_transtatus;
    short out_deadlocks;
};

// MISCELLANEOUS HELPER FUNCTIONS
inline void appendText(char **string,char *text);
inline void appendText(char **string,char *text,int length,int justify);
inline void appendChar(char **string,char byte);
inline void DEBUGMSG(FILE * debugFile, char * message);
inline void appendSpaces(char **string,int spaces);

```

```

inline void calcOutDateTme(const INT64b value,datetime *timestamp);
inline int copyOutPhone(char *buffer,char *value,int len);
inline bool copyInMoney64(const char * value,INT64 *number);
inline int copyInMoney(const char *value);
inline void copyOutMoney64(char *buffer,INT64b value,unsigned int len);
inline int copyOutDate(char *buffer,INT64b value);
inline int copyOutDate(char *buffer,INT64b value);
inline int copyOutTime(char *buffer,INT64b value);
inline int copyOutDecimal64(char *buffer,INT64b value,unsigned int len);
inline UINT16b changeOrder16(UINT16b value);
inline UINT32b changeOrder32(UINT32b value);
inline UINT64b changeOrder64(UINT64b value);
inline INT16b changeOrder16(INT16b value);
inline INT32b changeOrder32(INT32b value);
inline INT64b changeOrder64(INT64b value);
//
// Name      : appendText
// Description :
//      Append text to string
// Parameters :
//      char ** - string point to append to
//      char * - text to append
// Returns    :
//      None
// Comments   :
//
inline void appendText(char **string,char *text)
{
    while(*text)
    {
        *(*string)++ = *text++;
    }
    **string='\0';
    return;
}
//
// Name      : appendText
// Description :
//      Append text to string
// Parameters :
//      char ** - string point to append to
//      char * - text to append
//
//      int - total field length including
blank spaces
//      int - justify flag
// Returns    :
//      None
// Comments   :
//      right justify
//      left justify

inline void appendText(char **string,char *text,int length,int justify)
{
    int byteCount = 0;

    if(justify)
    {
        while(*text)
        {
            *(*string)++ = *text++;
            byteCount++;
        }

        //append blank spaces if text is less than length at end
        for(byteCount;byteCount < length;byteCount++)
            *(*string)++ = ' ';
    }
    else

```

```

{
    long long textLen = strlen(text);
    for(textLen;textLen < length;textLen++)
        *(*string)++ = ' ';
    while(*text)
        *(*string)++ = *text++;
}
**string='\0';
}
// Name      : appendChar
// Description :
//      Append text to string
// Parameters :
//      char ** - string point to append to
//      char * - text to append
// Returns    :
//      None
// Comments   :
//
inline void appendChar(char **string,char byte)
{
    *(*string)++ = byte;
    **string='\0';
    return;
}
//
// Name      : appendSpaces
// Description :
//      appends buffer spaces to result
page
// Parameters :
//      **htmlPage
// Returns    :
//      amount of characters
the function appened
//      to the html page
// Comments   :
//
inline void appendSpaces(char **string,int spaces)
{
    for(int index=0;index<spaces;index++)
    {
        *(*string)++ = ' ';
    }
    **string='\0';
}
//
// Name      : appendCustData
// Description :
//      appends cust data buffer to result
page
// Parameters :
//      **htmlPage
// Returns    :
//      Adds a newline
character every 50 characters displayed.
// Comments   :
//
inline void appendCustData(char **string,char *text)
{
    short byteCount = 0;
    while(*text)

```

```

    {
        *(*string)++ = *text++;
        byteCount++;
        if(byteCount % 50 == 0)
        {
            *(*string)++ = '\n';
            *(*string)++ = ' ';
            *(*string)++ = ' ';
            *(*string)++ = ' ';
            *(*string)++ = ' ';
            *(*string)++ = ' ';
            *(*string)++ = ' ';
            *(*string)++ = ' ';
            *(*string)++ = ' ';
            *(*string)++ = ' ';
        }
        **string='\0';
    }
}

```

```

//
// calcOutDateTime
//
// Title          : Calculate date & time data out of class array
// Parameters     : INT64b - date & time expressed in seconds
//               : datetime * - timestamp
// Return Value   : None
// Comments      :
//
inline void calcOutDateTime(const INT64b value,datetime *timestamp)
{

```

```

    // fixed days in each month (FEB 29 is special case)
    static int daysInMonth[12] =
{31,28,31,30,31,30,31,31,30,31,30,31};
    // mask out EPOCH seconds
    int dateValue = ((int) (value & 0xffffffff)) +
(SECONDS_IN_DAY -
(GMT_OFFSET * SECONDS_IN_HOUR));
    int offset = (int) (value >> 32);
    // break out the seconds
    int hms = dateValue % SECONDS_IN_DAY;
    int days = dateValue / SECONDS_IN_DAY;
    int years = (days - 1) / DAYS_IN_YEAR;
    int leaps = years / YEARS_IN_LEAP;
    int daysUsed = (years * DAYS_IN_YEAR) + leaps;
    // adjust the number of days to account for calculated years
    days = days - daysUsed;
    // set the starting year, month, and day
    timestamp->day = 1;
    timestamp->month = 1;
    timestamp->year = START_YEAR + years;
    // is the current year a leap year
    int leap = !(timestamp->year % YEARS_IN_LEAP);
    // apply remaining days based on days in months
    int daysInCurrentMonth;
    while(days)
    {
        // get days in current month
        daysInCurrentMonth =
daysInMonth[timestamp->month - 1];
        if(timestamp->month == 2 && leap)
            daysInCurrentMonth =
daysInCurrentMonth + 1;

        // days > days in current month
        if(days > daysInCurrentMonth)
        {
            // increment month
            timestamp->month += 1;
            days = days -
daysInCurrentMonth;

```

```

// month exceeds months in year
if(timestamp->month >
MONTHS_IN_YEAR)
{
    // increment year and
    timestamp->year += 1;
    timestamp->month = 1;
    // are we now on a leap
    leap =
!(timestamp->year % YEARS_IN_LEAP);
}
else
{
    // set day of month to remaining
    timestamp->day = days; days = 0;
}
}
// set time values to remaining seconds
timestamp->hour = hms / SECONDS_IN_HOUR;
hms = hms % SECONDS_IN_HOUR;
timestamp->minute = hms / SECONDS_IN_MINUTE;
timestamp->second = hms % SECONDS_IN_MINUTE;
return;
}

```

```

//
// copyOutZip
//
// Title          : Copy zip data out of class array
// Parameters     : char * - buffer to copy zip string into
//
// Return Value   : int - Length of copy
// Comments      :
//
inline int copyOutZip(char *buffer,char *value,int len = DEFAULT_ZIP_LEN)
{

```

```

    int index = 0;
    int bufferPos = 0;
    // add each digit of zip number to buffer inserting delimiter at 5
    while(value[index] && bufferPos < len)
    {
        if(index == 5)
            buffer[bufferPos++] = ZIP_DELIMITER;
            buffer[bufferPos++] = value[index++];
    }
    // space fill to the required length
    while(bufferPos < len)
        buffer[bufferPos++] = CHAR_FILL;
    buffer[bufferPos] = NULL;
    return len;
}

```

```

//
// copyOutPhone
//
// Title          : Copy phone data out of class array
// Parameters     : char * - buffer to copy phone string into
//
// Return Value   : int - Length of copy
// Comments      :
//
inline int copyOutPhone(char *buffer,char *value,int len =
DEFAULT_PHONE_LEN)
{
    int index = 0;
    int bufferPos = 0;

```

```

// add each digit of phone number to buffer inserting delimiter before
6, 9, and 12
while(value[index] && index < len)
{
    switch(index)
    {
        case 6:
        case 9:
        case 12:
            // insert delimiter
            buffer[bufferPos++] = PHONE_DELIMITER;
        default:
            // add phone digit to buffer
            buffer[bufferPos++] = value[index++];
    }
}

// space fill to the required length
while(bufferPos < len)
    buffer[bufferPos++] = CHAR_FILL;
buffer[bufferPos] = '\0';
return len;
}
//
// copyInMoney64
//
// Title           : Copy money data into class array
// Parameters      : const char * - value string
// Return Value    : INT64b integer value
// Comments       :
//
inline bool copyInMoney64(const char * value,INT64b *number)
{
    //INT64b  number           = 0;
    int      index           = 0;
    int      decimal         = 0;
    int      decimals        = 0;
    int      digitsAfterDec  = 0;
    bool     negativeFlag    = false;
    // convert each digit to a numeric portion
    while(value[index])
    {
        // handle $ . - All the rest assumed numeric
        switch(value[index])
        {
            case MONEY_SYMBOL:
                // ignore $ sign
                break;
            case NEGATIVE_SYMBOL:
                // set negative flag
                negativeFlag = true;
                break;
            case DECIMAL_SYMBOL:
                // set decimal
                decimal=1;
                decimals++;
                if(decimals > 1)
                    //more than 1 decimal point found
                    return false;
                break;
            default:
                // adjust decimal places
                decimal = decimal * 10;
                // add digit to running total
                if(value[index] >= '0' && value[index] <= '9')
                {
                    if(decimal)

```

```

                if(++digitsAfterDec >
2)
                    return false;
                *number = (*number * 10) +
                    (value[index] - '0');
            }
            else
            {
                //non-numeric field inserted
                return false;
            }
        }
        index++;
    }
}

// apply decimal where decimal not found
if(decimal < 100)
{
    if(decimal)
    {
        *number *= (100 / decimal);
    }
    else
    {
        *number *= 100;
    }
}
// make negative
if(negativeFlag)
    *number = *number * (-1);
return true;
}
//
// copyInMoney
//
// Title           : Convert char string money field to double
// Parameters      : const char * - value string
// Return Value    : double integer value
// Comments       :
//
inline int copyInMoney(const char *value)
{
    char buf[20];
    int i,j,decimalFound,digitsAfterDecimal=0;
    int decimal=0;
    //walk past $ if present in char string
    if(*value == '$')
        *value++;
    int len=(int)strlen(value);
    for (i=0;i<len;i++)
    {
        if(value[i] == '.')
        {
            decimalFound++;
            if(decimalFound > 1)
                return -1;
        }
        if(value[i] == '-')
        {
            if (value[i] != '-')
            {
                if(decimal)
                {
                    if(digitsAfterDecimal<2)
                        digitsAfterDecimal++;
                    else
                        return -1;
                }
            }

```

```

        buf[j++] = value[i];
    }
    }
    int amount = atoi(buf);
    return amount;
}

//
// copyOutMoney64
//
// Title      : Copy money data out of class array
// Parameters  : char * - buffer to copy string 64 bit money into
//              INT64b - value
//              unsigned len - max number of
//              bytes to copy
// Return Value : int - Length of copy
// Comments   :
//
inline void copyOutMoney64(char *buffer,INT64b value,unsigned int len =
DEFAULT_MONEY64_LEN)
{
    unsigned int    index          = len;
    int             places         = 0;
    bool            negativeFlag   = false;
    bool            moneyFlag      = true;
    // NULL terminate string
    buffer[index] = NULL;
    // check length > 0
    // if(!index) return len;
    // handle negative value
    if(value < 0)
    {
        negativeFlag = true;
        value = value * (-1);
    }
    // break off each digit from value, fill if needed
    do
    {
        if(value)
        {
            // get next digit and add to buffer
            buffer[--index] = (char) (value % 10 + '0');
            value /= 10; places++;
            if(places == 2 && index)
            {
                places++;
                buffer[--index] =
DECIMAL_SYMBOL;
            }
            else
            {
                // add zeros to first place before decimal point
                on (i.e. 0.00)
                if(places < 2 || places == 3)
                {
                    buffer[--index] =
ZERO_SYMBOL;
                }
                else
                {
                    // add the decimal point
                    if(places == 2)
                    {
                        buffer[--index] =
DECIMAL_SYMBOL;
                    }
                }
            }
        }
    }
}

```

```

    else
    {
        // add the negative
        indicator
        if(negativeFlag)
        {
            negativeFlag
        }
        = false;
        buffer[--index] = NEGATIVE_SYMBOL;
    }
    else
    {
        // add the
        money indicator
        if(moneyFlag)
        {
            moneyFlag = false;
            buffer[--index] = MONEY_SYMBOL;
        }
        else
        {
            buffer[--index] = NUMERIC_FILL;
        }
    }
    // need to trace place for decimal point and
    zero fill
    places++;
    } while(index);
    //return len;
}
//
// copyOutDateTime
//
// Title      : Copy date & time data out of class array
// Parameters  : char * - buffer to copy date & time string into
//              INT64b - value
// Return Value : int - Length of copy
// Comments   : Fixed length
//
inline int copyOutDateTime(char *buffer,INT64b value)
{
    datetime timestamp;
    // break value into time/date components
    calcOutDateTime(value,&timestamp);
    // put month into buffer
    *buffer++ = (char) ((timestamp.month / 10) + '0');
    *buffer++ = (char) ((timestamp.month % 10) + '0');
    *buffer++ = DATE_DELIMITER;
    // put day into buffer
    *buffer++ = (char) ((timestamp.day / 10) + '0');
    *buffer++ = (char) ((timestamp.day % 10) + '0');
    *buffer++ = DATE_DELIMITER;
    // put year into buffer
    int year = timestamp.year;
    *buffer++ = (char) ((year / 1000) + '0');
    year = year% 1000;
    *buffer++ = (char) ((year / 100) + '0'); year = year
% 100;
    *buffer++ = (char) ((year / 10) + '0');
    *buffer++ = (char) ((year % 10) + '0');
    *buffer++ = CHAR_FILL;
    // put hour into buffer
}

```



```

    *buffer++ = (char)((timestamp.hour / 10) +
'0');
    *buffer++ = (char)((timestamp.hour % 10) +
'0');
    *buffer++ = TIME_DELIMITER;
// put minute into buffer
    *buffer++ = (char)((timestamp.minute / 10) +
'0');
    *buffer++ = (char)((timestamp.minute % 10) +
'0');
    *buffer++ = TIME_DELIMITER;
// put second into buffer
    *buffer++ = (char)((timestamp.second / 10) +
'0');
    *buffer++ = (char)((timestamp.second % 10) +
'0');
    *buffer = NULL; return DEFAULT_DATETIME_LEN;
}
//
// copyOutTime
//
// Title : Copy date data out of class array
// Parameters : char * - buffer to copy date string into
// INT64b - value
// Return Value : int - Length of copy
// Comments : Fixed length
//
inline int copyOutDate(char *buffer,INT64b value)
{
    datetime timestamp;
// break value into time/date components
    calcOutDateTime(value,&timestamp);
// put month into buffer
    *buffer++ = (char)((timestamp.month / 10) + '0');
    *buffer++ = (char)((timestamp.month % 10) + '0');
    *buffer++ = DATE_DELIMITER;
// put day into buffer
    *buffer++ = (char)((timestamp.day / 10) + '0');
    *buffer++ = (char)((timestamp.day % 10) + '0');
    *buffer++ = DATE_DELIMITER;
// put year into buffer
    int year = timestamp.year;
    *buffer++ = (char)((year / 1000) + '0'); year = year % 1000;
    *buffer++ = (char)((year / 100) + '0'); year = year % 100;
    *buffer++ = (char)((year / 10) + '0');
    *buffer++ = (char)((year % 10) + '0');
    *buffer++ = CHAR_FILL;
    *buffer = NULL;

    return DEFAULT_DATE_LEN;
}
//
// copyOutTime
//
// Title : Copy time data out of class array
// Parameters : char * - buffer to copy time string into
// INT64b - value
// Return Value : int - Length of copy
// Comments : Fixed length TBD
//
inline int copyOutTime(char *buffer,INT64b value)
{
    datetime timestamp;
// break value into time/date components
    calcOutDateTime(value,&timestamp);
// put hour into buffer
    *buffer++ = (char)((timestamp.hour / 10) + '0');
    *buffer++ = (char)((timestamp.hour % 10) + '0');
    *buffer++ = TIME_DELIMITER;
// put minute into buffer
    *buffer++ = (char)((timestamp.minute / 10) + '0');
    *buffer++ = (char)((timestamp.minute % 10) + '0');
    *buffer++ = TIME_DELIMITER;
// put second into buffer
    *buffer++ = (char)((timestamp.second / 10) + '0');
    *buffer++ = (char)((timestamp.second % 10) + '0');
    *buffer = NULL; return DEFAULT_TIME_LEN;
}
//
// copyOutDecimal64
//
// Title : Copy decimal data out of class array
// Parameters : char * - buffer to copy string 64 bit money into
// INT64b - value
// unsigned len - max number of
// bytes to copy
// Return Value : int - Length of copy
// Comments :
//
inline int copyOutDecimal64(char *buffer,INT64b value,unsigned int len =
DEFAULT_DECIMAL64_LEN)
{
    unsigned int index = len;
    int places = 0;
    bool negativeFlag = false;
// NULL terminate string
    buffer[index] = NULL;
// check length > 0
    if(!index) return len;
// handle negative value
    if(value < 0)
    {
        negativeFlag = true;
        value = value * (-1);
    }
// break off each digit from value, fill if needed
    do
    {
        if(value)
        {
            // get next digit and add to buffer
            buffer[--index] = (char)(value % 10 + '0');
            value /= 10; places++;
            if(places == 2 && index)
            {
                places++;
                buffer[--index] =
DECIMAL_SYMBOL;
            }
        }
        else
        {
            // add zeros to first place before decimal point
            on (i.e. 0.00)
            if(places < 2 || places == 3)
            {
                buffer[--index] =
ZERO_SYMBOL;
            }
            else
            {
                // add the decimal point
                if(places == 2)
                {

```

```

DECIMAL_SYMBOL;
    }
    else
    {
        // add the negative
        indicator
        if(negativeFlag)
        {
            negativeFlag
            = false;
            buffer[--index] = NEGITIVE_SYMBOL;
        }
        else buffer[--index] =
        NUMERIC_FILL;
    }
    // need to trace place for decimal point and
    zero fill
    places++;
    } while(index);
    return len;
}
// Macros
using namespace std;
#ifdef _DEBUG
    int debugFlag = 1;
#else
    int debugFlag = 0;
#endif
inline BYTE8b *debugFileName(BYTE8b *filePath)
{
    BYTE8b *fileName = filePath + strlen(filePath);
    while(fileName != filePath)
    {
        if(*fileName == '/' || *fileName == '\\' && *(fileName +
1))
            return (fileName + 1);
        fileName--;
    }
    return filePath;
}
#define DEBUGADDRESS(POINTER)    hex << (void *) POINTER << dec
#define ERRORMSG(TEXT)
\
EnterCriticalSection(&errorMutex);
\
<< debugFileName(__FILE__)
    errorStream
\
    << "|" <<
    __TIMESTAMP__ << "|" << __LINE__ << "|"
    \
    << _getpid()
    << "|" << GetCurrentThreadId() << "|"
    \
    << TEXT;
\
errorStream.flush();
\
LeaveCriticalSection(&errorMutex);
#ifdef _DEBUG
    #define DEBUGMSG(TEXT)
\
EnterCriticalSection(&debugMutex);
\
debugStream << debugFileName(__FILE__)
    << "|" <<
    __TIMESTAMP__ << "|" << __LINE__ << "|"
    \
    << _getpid()
    << "|" << GetCurrentThreadId() << "|"
    \
    << TEXT;
\
debugStream.flush();
\
LeaveCriticalSection(&debugMutex);
    #define DEBUGSTRING(TEXT,LENGTH)
\
    debugVarString(TEXT,LENGTH)
\
#else
    #define DEBUGMSG(TEXT);
    #define DEBUGSTRING(TEXT,LENGTH);
\
#endif
#endif /* _COMMON_TPCC */

tpccIsapi/tpccIsapi.def
; tpccIsapi.def : declares the module parameters for the DLL.
LIBRARY "tpccIsapi"
EXPORTS
    HttpExtensionProc
    GetExtensionVersion
    TerminateExtension

tpccIsapi/tpccIsapi.hpp
/*
*****
** Project      : AIX
** Component    : Performance/TPC-W Benchmark
** Name         : tpccIsapi.hpp
** Title        : ISAPI interface for tpcc
*****
** Copyright (c) 2001,2002 IBM Corporation
** All rights reserved
*****
** History      :
**              : Developed at IBM Austin by the AIX RS/6000
**              : performance group.

```

```

**
** Comments :
**
*****
*/
#ifdef __tpccSAPI_hpp__
#define __tpccSAPI_hpp__
#include <windows.h>
#include <httpext.h>
#include <tpcc.h>
#include <htmlPhraser.h>
#include <iomanip>
#include <db2tpcc.h>
#include <comsvcs.h>
// Terminal struct
struct TERM_ENTRY
{
    int terminalID;
    bool terminalInUse;
    int w_id;
    short d_id;
};
// COM interface
struct COM_HANDLE
{
    Itpcc_com *comHandle;
    char *txnBuffer;
    int size;
};
// TXN handle
struct TXN_HANDLE
{
    char htmlPage[MAX_HTML_PAGE_LEN];
    char htmlHeader[MAX_HTML_HEADER_LEN];
    char *urlString;
    //user data
    int w_id;
    int d_id;
    int sync_id;
    int term_id;
    int conn_id;
    COM_HANDLE comInterface;
};
struct DLVYQUEUEDATA
{
    int warehouse;
    short in_s_0_CARRIER_ID;
    struct _timeb enqueueTime;
};
// Definitions
#define INVALID_ITEM 100
#define HEADER
"Content-Type:text/html\r\nContent-Length: %d\r\nConnection:
Keep-Alive\r\n\r\n"
#define TLS_NULL
0xFFFFFFFF
#define ACCESS_TIMEOUT 360000
seconds //One hour in milli

```

```

#define DELIVERY_LOG_SUCCESS_STR "--Tran %d
Queue %d.%03d Start %d.%03d\nW_ID: %d CARRIER_ID: %d
%\s\nend-time: %d.%03d\n"
// Function Prototypes
int initDlvy();
int initTxnHandle(TXN_HANDLE **txnHandle);
int closeTxnHandle(TXN_HANDLE *txnHandle);
int readRegistryValues();
int getTerminal(int terminal, TXN_HANDLE *txnHandle);
int assignTerminal(TXN_HANDLE *txnHandle);
int getDBInstance();
void doHtml(TXN_HANDLE *txnHandle);
int doLoginForm(htmlPhraser *commandBlock, TXN_HANDLE *txnHandle);
int doLoginResults(htmlPhraser *commandBlock, TXN_HANDLE *txnHandle);
int doNewOrderForm(htmlPhraser *commandBlock, TXN_HANDLE *txnHandle);
int doNewOrderResults(htmlPhraser *commandBlock, TXN_HANDLE *txnHandle);
int doPaymentForm(htmlPhraser *commandBlock, TXN_HANDLE *txnHandle);
int doPaymentResults(htmlPhraser *commandBlock, TXN_HANDLE *txnHandle);
int doOrderStatusForm(htmlPhraser *commandBlock, TXN_HANDLE *txnHandle);
int doOrderStatusResults(htmlPhraser *commandBlock, TXN_HANDLE *txnHandle);
int doDeliveryForm(htmlPhraser *commandBlock, TXN_HANDLE *txnHandle);
int doDeliveryResults(htmlPhraser *commandBlock, TXN_HANDLE *txnHandle);
int doStockForm(htmlPhraser *commandBlock, TXN_HANDLE *txnHandle);
int doStockResults(htmlPhraser *commandBlock, TXN_HANDLE *txnHandle);
int doExit(htmlPhraser *commandBlock, TXN_HANDLE *txnHandle);
int doLoginErrorPage(char *htmlPage, char *message);
int doNewOrderErrorPage(char *htmlPage, char *message, htmlPhraser *commandBlock, TXN_HANDLE *txnHandle);
int doPaymentErrorPage(char *htmlPage, char *message, htmlPhraser *commandBlock, TXN_HANDLE *txnHandle);
int doOrderStatusErrorPage(char *htmlPage, char *message, htmlPhraser *commandBlock, TXN_HANDLE *txnHandle);
int doDeliveryErrorPage(char *htmlPage, char *message, htmlPhraser *commandBlock, TXN_HANDLE *txnHandle);
int doStockErrorPage(char *htmlPage, char *message, htmlPhraser *commandBlock, TXN_HANDLE *txnHandle);
void dlvyThreadEntry(void *);
int queueDlvyTxn(int warehouse, short carrier_id);
int appendButtons(char *htmlPage);
int appendItems(char *htmlPage, short itemCount, short cmdIDStart);
int appendHiddenFields(char *htmlPage, TXN_HANDLE *txnHandle);
int displayStatus(char *htmlPage, int rc);
#endif

```

tpccsapi/htmlPhraser.cpp

```

// htmlPhraser.cpp
// Class implementation of htmlPhraser.
// This class will take a query string and break it into a series
// of consuitant parts
#include "htmlPhraser.h"
// htmlPhraser:htmlPhraser

```

```

// Title      : Constructor
// Parameters   : char * query string
// Return Value : None
// Comments    :
///////////////////////////////////////////////////////////////////
htmlPhraser::htmlPhraser(char *queryString)
{
    // initialize query values
    iCustomerIdFlag = iCarrierNumFlag = iStockThresholdFlag = false;
    // this initializes the query list to NULL's. This means that
    // characters being added are overwriting null characters and
    // therefore the string will be null terminated implicitly.

    memset(iQueryValues, NULL, (MAX_FIELD_NUM *
MAX_FIELD_LEN));
    // controls
    char          queryChar          = NULL;
    int           queryIndex          = -1;
    int           valueIndex          = -1;
    // process each character of query string
    while(*queryString)
    {
        // check for special case characters
        if(queryChar)
        {
            // a percentage sign would indicate a token
            if(*queryString != '%')
            {
                // a plus sign represents a space
                if(*queryString == '+')
                {
                    queryChar = ' ';
                    *queryString++;
                }
                else queryChar = *queryString++;
            }
            else queryChar =
convertQueryToken(&queryString);
        }
        else queryChar = '&';
        // handle query reference (&)
        if(queryChar == '&')
        {
            // reset value index
            valueIndex = -1;
            // do we have a numeric query reference
            if(*queryString >= '0' && *queryString <=
'9')
            {
                // numeric query id
                queryIndex =
                    ((*queryString - '0') *
10) + (*(queryString + 1) - '0');

                // walk past the two command
                characters

                queryString += 2;

                // validate query value
                if(queryIndex >
MAX_QUERY_ID)
                    queryIndex = -1;
            }
            else queryIndex = -1;
            // finished processing for query reference
            continue;
        }
    }
}

```

```

'='
// we have a query reference but need to wait until we see
// before accepting value
if(valueIndex == -1)
{
    // we are waiting for '='
    if(queryChar == '=')
    {
        valueIndex = 0;
        // set query string flags
        switch(queryIndex)
        {
            case C_ID:
                iCustomerIdFlag = true;

            case CARRIER_NUM:
                iCarrierNumFlag =

            case STK_THRESHOLD:
                iStockThresholdFlag =

            default: break;
        }
    }
    // finishes looking for '='
    continue;
}
// add each character to the query value
if(queryIndex > -1 && valueIndex > -1)
{
    // we are processing a query value
    if(valueIndex < MAX_FIELD_LEN)
    {
        // we have not exceeded max line len
        iQueryValues[queryIndex][valueIndex++] = queryChar;
    }
    continue;
}
}
return;
}
/////////////////////////////////////////////////////////////////
// htmlPhraser::getCommandId
/////////////////////////////////////////////////////////////////
// Title      : Returns the page command
// Parameters   : None
// Return Value : int - page command
// Comments    :
///////////////////////////////////////////////////////////////////
int htmlPhraser::getCommandId()
{
    // return command numeric code
    switch(*iQueryValues[COMMAND_ID])
    {
        case NEW_ORDER_CODE:
            if(iCustomerIdFlag)
                return
COMMAND_NEW_ORDER_RESULTS;
            else return COMMAND_NEW_ORDER;
        case PAYMENT_CODE:
            if(iCustomerIdFlag)
                return COMMAND_PAYMENT_RESULTS;
            else return COMMAND_PAYMENT;
        case ORDER_STATUS_CODE:
            if(iCustomerIdFlag)

```

```

        return
COMMAND_ORDER_STATUS_RESULTS;
    else return COMMAND_ORDER_STATUS;
case DELIVERY_CODE:
    if(iCarrierNumFlag)
        return COMMAND_DELIVERY_RESULTS;
    else return COMMAND_DELIVERY;
case STOCK_CODE:
    if(iStockThresholdFlag)
        return COMMAND_STOCK_RESULTS;
    else return COMMAND_STOCK;
case MENU_CODE:
    return COMMAND_LOGIN_RESULTS;
case EXIT_CODE:
    return COMMAND_EXIT;
default:
    return COMMAND_LOGIN;

};

// should not get here
return COMMAND_LOGIN;
}
// =====
// htmlPhraser::validate
// =====
// Title    : validate url parameter list for all txn types
// Parameters : int - txn type
// Return Value : int - error code
// Comments :
// =====

int validate(int txnType)
{
    return 0;
}
// =====
// htmlPhraser::convertQueryToken
// =====
// Title    : Returns the page command
// Parameters : None
// Return Value : int - page command
// Comments :
// =====
char htmlPhraser::convertQueryToken(char **queryString)
{
    char queryChar = NULL;
    // skip over %
    (*queryString)++;
    // look at first character
    switch(**queryString)
    {
        case '2':
            {
                // what follows?
                (*queryString)++;
                switch(**queryString)
                {
                    case '1':
                        queryChar = '!';
                        break;

                    case '3':
                        queryChar = '#';
                        break;

                    case '4':
                        queryChar = '$';
                        break;

                    case '5':

```

```

                        queryChar = '%';
                        break;

                    case '6':
                        queryChar = '&';
                        break;

                    case '8':
                        queryChar = '(';
                        break;

                    case '9':
                        queryChar = ')';
                        break;

                    case 'B':
                        queryChar = '+';
                        break;

                    case 'C':
                        queryChar = ',';
                        break;

                    case 'F':
                        queryChar = '/';
                        break;

                    case ':':
                        queryChar = ':';
                        break;

                }
            }
        case '3':
            {
                // what follows?
                (*queryString)++;
                switch(**queryString)
                {
                    case 'A':
                        queryChar = '!';
                        break;

                    case 'B':
                        queryChar = '+';
                        break;

                    case 'D':
                        queryChar = '=';
                        break;

                    case 'F':
                        queryChar = '?';
                        break;

                    case ':':
                        queryChar = ':';
                        break;

                }
            }
        case '4':
            {
                // what follows?
                (*queryString)++;
                switch(**queryString)
                {
                    case '0':
                        queryChar = '@';
                        break;

                    case ':':
                        queryChar = ':';
                        break;

                }
            }
        case '5':
            {
                // what follows?

```

```

        (*queryString)++;
        switch(**queryString)
        {
        case 'B':
            queryChar = '[';
            break;

        case 'D':
            queryChar = ']';
            break;

        case 'E':
            queryChar = '^';
            break;

        case ' ':
            queryChar = ' ';
            break;

        }
        break;
    case '7':
    {
        // what follows?
        (*queryString)++;
        switch(**queryString)
        {
        case 'B':
            queryChar = '{';
            break;

        case 'C':
            queryChar = '|';
            break;

        case 'D':
            queryChar = '}';
            break;

        case 'E':
            queryChar = '~';
            break;

        case ' ':
            queryChar = ' ';
            break;

        }
        break;
    case '+':
        queryChar = '+';
        break;
    }
    // advance pointer and return
    (*queryString)++; return queryChar;
}
///////////////////////////////////////////////////////////////////

```

tpccsapi/StdAfx.cpp

```

// stdafx.cpp : source file that includes just the standard includes
//          tpccsapi.pch will be the pre-compiled header
//          stdafx.obj will contain the pre-compiled type information
#include "stdafx.h"
// TODO: reference any additional headers you need in STDAFX.H
// and not in this file

```

tpccsapi/tpccsapi.cpp

```

/*
*****
** Project      : AIX
** Component    : Performance/TPC-C Benchmark

```

```

** Name          : tpccsapi.cpp
** Title         : TPCc html processing
*****
** Copyright (c) 2003 IBM Corporation
** All rights reserved
*****
** History      :
**              : Developed at IBM Austin by the AIX RS/6000
**              : performance group.
**
** Comments    :
**
*****
*/
#include "stdafx.h"
#include "..\tpccCom\tpccCom.h"
#include "..\tpccCom\tpccCom_i.c"
#include <tpccsapi.hpp>
// For custom assert and trace handling with WebDbg.exe
[ module(name="tpccsapi", type="dll") ];
[ emitidl(restricted) ];
#define _WIN32_DCOM
///////////////////////////////////////////////////////////////////
// Globals
///////////////////////////////////////////////////////////////////
int          maxDataSize;
//max struct size of all txn(s)
int          numUsers;
//number of users that client will service.
int          dlvyQueueLen;
//static length of dlvy queue
int          dlvyThreads;
//number of dlvy threads to create
int          dlvyBufferFreeSlots;           //length of dlvy txn
queue
int          dlvyBufferSlotIndex;          //index into next
available slot in dlvy txn queue
int          dlvyBufferThreadIndex;        //thread
index into dlvy txn queue
int          nullDB;
//null db on client(bypass com call).
int          trace;
static DWORD          threadLSIndex;
//isapi thread local storage index
CRITICAL_SECTION     isapiLock;
//isapi lock
CRITICAL_SECTION     errorLock;
//error log file lock.
CRITICAL_SECTION     termLock;
//terminal array lock.
CRITICAL_SECTION     dlvyQueueLock;
//dlvy queue critical section lock
HANDLE               dlvyThreadDone =
INVALID_HANDLE_VALUE; //dlvy thread exit event
HANDLE               dlvyThreadSemaphore
= INVALID_HANDLE_VALUE; //dlvy thread wrk to do semaphore
int
dlvyThreadID = 0;
struct DLVYQUEUEDATA *dlvyQueue;
//dlvy queue
HANDLE               *dlvyThreadHandles;
//ptr to array of thread handles
TERM_ENTRY           *termArray;
//array of terminal entries to store each users info.
int
termNextFree;
//next available slot in terminal array
FILE                 *htmlDebug = NULL;
//html debug file

```

```

FILE *errorLog = NULL;
//error file
FILE *htmlTrace = NULL;

ofstream debugStream;
ofstream errorStream;
CRITICAL_SECTION debugMutex;
CRITICAL_SECTION errorMutex;
char dlvyLogPath[128] = {NULL};
char errorLogFile[128] = {NULL};
char htmlTraceLogFile[128] = {NULL};
char dbName[64] = {NULL};
char dbType[16] = {NULL};
typedef INT (*CONNECT_PTR)(char *dbName,void **connectHandle);
typedef INT (*DISCONNECT_PTR)(void *connectHandle);
typedef INT (*DLVY_FUNC_PTR)(dlvy_wrapper *dlvy,void
*connectHandle);
typedef INT (*NORD_FUNC_PTR)(nord_wrapper *nord,void
*connectHandle);
typedef INT (*PYMT_FUNC_PTR)(paym_wrapper *pymt,void
*connectHandle);
typedef INT (*ORDS_FUNC_PTR)(ords_wrapper *ords,void *connectHandle);
typedef INT (*STOK_FUNC_PTR)(stok_wrapper *stok,void *connectHandle);
HINSTANCE dbInstance;
CONNECT_PTR db_connect;
DISCONNECT_PTR db_disconnect;
DLVY_FUNC_PTR dlvyCall;

// Page functions arrays
typedef int (*pageFuncPtr) (htmlPhraser *commandBlock, TXN_HANDLE
*txnHandle);
pageFuncPtr htmlPageFunctions[MAX_TRANSACTIONS] =
{
    {doLoginForm},
    {doNewOrderForm},
    {doPaymentForm},
    {doOrderStatusForm},
    {doDeliveryForm},
    {doStockForm},
    {doExit},
    {doLoginResults},
    {doNewOrderResults},
    {doPaymentResults},
    {doOrderStatusResults},
    {doDeliveryResults},
    {doStockResults}
};

extern "C" DWORD WINAPI
HttpExtensionProc(LPEXTENSION_CONTROL_BLOCK lpECB)
{
    struct TXN_HANDLE *txnHandle = NULL;
    txnHandle = (TXN_HANDLE *) TlsGetValue(threadLSIndex);
    if(txnHandle == NULL)
    {
        int rc = initTxnHandle(&txnHandle);
        if (rc != OK)
        {
            function failed.\n");
            size_t htmlPageLen = strlen(response);
            //add content length and keep alive header
            sprintf(htmlHeader,HEADER,htmlPageLen);

            lpECB->ServerSupportFunction(lpECB->ConnID,HSE_REQ_SEND_RESPO
NSE_HEADER,"200 OK",NULL,(DWORD*)htmlHeader);
        }
    }
}

```

```

lpECB->WriteClient(lpECB->ConnID,response,(LPDWORD)&htmlPageLen,0
);
return
HSE_STATUS_SUCCESS_AND_KEEP_CONN;
}
txnHandle = (TXN_HANDLE *)
TlsGetValue(threadLSIndex);
if (txnHandle == NULL)
{
    char response[256]; char htmlHeader[256];
    sprintf(response,"ERROR: Unable to retrieve
txnHandle from TLS.\n");
    size_t htmlPageLen = strlen(response);
    //add content length and keep alive header
    sprintf(htmlHeader,HEADER,htmlPageLen);

    lpECB->ServerSupportFunction(lpECB->ConnID,HSE_REQ_SEND_RESPO
NSE_HEADER,"200 OK",NULL,(DWORD*)htmlHeader);

    lpECB->WriteClient(lpECB->ConnID,response,(LPDWORD)&htmlPageLen,0
);
return
HSE_STATUS_SUCCESS_AND_KEEP_CONN;
}
try
{
    txnHandle->urlString =
(char*)lpECB->lpszQueryString;

    DEBUGMSG("calling doHtml() w/ query string:" <<
txnHandle->urlString << endl);
    doHtml(txnHandle);

    size_t htmlPageLen;
    htmlPageLen = strlen(txnHandle->htmlPage);
    if(htmlPageLen >= 4096)
    {
        ERRORMSG("WARNING: HTML PAGE IS
>= 4096!, page size:"<<htmlPageLen<<endl);
    }
    //add content length and keep alive header
    sprintf(txnHandle->htmlHeader,HEADER,htmlPageLen);
    size_t headerLen = strlen(txnHandle->htmlHeader);
    if(headerLen >= 256)
    {
        ERRORMSG("WARNING: HTML
HEADER IS >= 256!, header size:"<<headerLen<<endl);
    }
    //write response to user

    lpECB->ServerSupportFunction(lpECB->ConnID,HSE_REQ_SEND_RESPO
NSE_HEADER,"200 OK",NULL,(DWORD*)txnHandle->htmlHeader);

    lpECB->WriteClient(lpECB->ConnID,txnHandle->htmlPage,(LPDWORD)&ht
mlPageLen,0);
    DEBUGMSG("HTML
PAGE-->"<<endl<<txnHandle->htmlHeader<<txnHandle->htmlPage<<endl);
}
catch (...)
{
    char response[256];
    ZeroMemory(response,256);
    char *ptr = response;
    appendText(&ptr,"<HTML><BODY> Error : Unhandled
Exception </BODY></HTML>");
    DWORD cbResponse = sizeof(response)-1 ;
}
}

```

```

//write response to user

lpECB->ServerSupportFunction(lpECB->ConnID,HSE_REQ_SEND_RESPON
NSE_HEADER,"200 OK",NULL,(DWORD*)response);

lpECB->WriteClient(lpECB->ConnID,response,&cbResponse,0);
}
return HSE_STATUS_SUCCESS_AND_KEEP_CONN;
}
extern "C" BOOL WINAPI GetExtensionVersion(HSE_VERSION_INFO*
pVer)
{
// Create the extension version string, and copy string to
HSE_VERSION_INFO structure.
pVer->dwExtensionVersion =
MAKELONG(HSE_VERSION_MINOR, HSE_VERSION_MAJOR);

// Copy description string into HSE_VERSION_INFO structure.
strcpy(pVer->lpszExtensionDesc, "TPCC ISAPI Extension");
// Initialize isapi critical section
InitializeCriticalSection(&isapiLock);
// Initialize error log critical section
InitializeCriticalSection(&errorLock);
// Initialize terminal critical section
InitializeCriticalSection(&termLock);
// Initialize debug/error critical sections
if(debugFlag)
InitializeCriticalSection(&debugMutex);
InitializeCriticalSection(&errorMutex);
// Read registry values
if(readRegistryValues() != OK)
return(FALSE);
// Initialize terminal array
termArray = (TERM_ENTRY*)
calloc(numUsers,sizeof(TERM_ENTRY));
termNextFree = 1;
//open up error/debug streams
errorStream.rdbuf( )->open(errorLogFile,ios::out);
if(debugFlag)
debugStream.rdbuf( )->open(htmlTraceLogFile,ios::out);
ERRORMSG("Error log file open."<<endl);
DEBUGMSG("Loading library for dlvy txn."<<endl);
int rc = getDBInstance();
if (rc != OK)
{
ERRORMSG("Error, unable to load database dll,
rc:<<rc);
DEBUGMSG("Error, unable to load database dll,
rc:<<rc);
return FALSE;
}
DEBUGMSG("Library loaded for dlvy txn."<<endl);
DEBUGMSG("Calling initDlvy." <<endl);
if(initDlvy() != OK)
return (FALSE);

DEBUGMSG("Initializing TLS." << endl);

// Initialize thread local storage index
threadLSIndex = TlsAlloc();
if (threadLSIndex == TLS_NULL)
{
ERRORMSG("Isapi error: unable to initialize thread
local storage(TLS), rc:" << GetLastError()<<endl);
return(FALSE);
}

```

```

DEBUGMSG("sizeof out_neword_struct: "<<sizeof(struct
out_neword_struct)<<endl);
DEBUGMSG("sizeof in_neword_struct: "<<sizeof(struct
in_neword_struct)<<endl);
DEBUGMSG("sizeof out_payment_struct: "<<sizeof(struct
out_payment_struct)<<endl);
DEBUGMSG("sizeof in_payment_struct: "<<sizeof(struct
in_payment_struct)<<endl);
DEBUGMSG("sizeof out_ordstat_struct: "<<sizeof(struct
out_ordstat_struct)<<endl);
DEBUGMSG("sizeof in_ordstat_struct: "<<sizeof(struct
in_ordstat_struct)<<endl);
DEBUGMSG("sizeof out_delivery_struct: "<<sizeof(struct
out_delivery_struct)<<endl);
DEBUGMSG("sizeof in_delivery_struct: "<<sizeof(struct
in_delivery_struct)<<endl);
DEBUGMSG("sizeof out_stocklev_struct: "<<sizeof(struct
out_stocklev_struct)<<endl);
DEBUGMSG("sizeof in_stocklev_struct: "<<sizeof(struct
in_stocklev_struct)<<endl);
//compute the max struct size for com data construct
maxDataSize = max(maxDataSize,sizeof(nord_wrapper));
maxDataSize = max(maxDataSize,sizeof(paym_wrapper));
maxDataSize = max(maxDataSize,sizeof(ords_wrapper));
maxDataSize = max(maxDataSize,sizeof(dlvy_wrapper));
maxDataSize = max(maxDataSize,sizeof(stok_wrapper));
maxDataSize += 10;
DEBUGMSG("max data struct size:"<<maxDataSize <<endl);

return true;
}
extern "C" BOOL WINAPI TerminateExtension(DWORD dwFlags)
{
return true;
}
/*
*****
** Name : initTxnHandle
** Description : Isapi thread initializes
its own com interface
** Parameters : structure.
** Returns : TXN_HANDLE**
** Comments : int - return code
*****
*/
int initTxnHandle(TXN_HANDLE **txnHandle)
{
DEBUGMSG("Inside init txn handle, getting isapiLock." << endl);
EnterCriticalSection(&isapiLock);

HRESULT hres = NULL;
try
{
DEBUGMSG("Got ispaiLock, initializing txnHandle:
"<<DEBUGADDRESS(*txnHandle)<< endl);
*txnHandle = (TXN_HANDLE *)
calloc(1,sizeof(TXN_HANDLE));
if (*txnHandle == NULL)
{
ERRORMSG("Unable to allocated
TXN_HANDLE, rc:"<<GetLastError()<<endl);

```



```

        return ERR;
    };
    (*txnHandle)->comInterface.comHandle = NULL;
    DEBUGMSG("Initializing txnHandle com data buffer to
" <<maxDataSize<<"bytes" <<endl);
    (*txnHandle)->comInterface.txnBuffer = (char *)
CoTaskMemAlloc(maxDataSize);
    if (!((*txnHandle)->comInterface.txnBuffer))
    {
        ERRORMSG("CoTaskMemAlloc() failed of
size " <<maxDataSize<< ", rc: " <<hres<<endl);
        return(ERR);
    };
    DEBUGMSG("txnHandle com data buffer initialized to "
<< maxDataSize << "bytes" <<endl);
    DEBUGMSG("Calling CoInitialize with txnHandle:
" <<DEBUGADDRESS(*txnHandle)<<endl);
    hres =
CoInitializeEx(NULL, COINIT_MULTITHREADED);
    if (FAILED(hres))
    {
        ERRORMSG("CoInitializeEx() failed, rc :
" <<hres<<endl);
        return(ERR);
    };

    struct _timeb
startTime;
    struct _timeb
endTime;

    DEBUGMSG("Calling CoCreateInstance with
txnHandle:" <<DEBUGADDRESS(*txnHandle)<< endl);
    _ftime(&startTime);
    hres =
CoCreateInstance(CLSID_tpcc_com, NULL, CLSCTX_SERVER, IID_Itpcc_co
m, (void **)&(*txnHandle)->comInterface.comHandle);
    if (FAILED(hres))
    {
        _ftime(&endTime);
        //store error code in txnHandle
        ERRORMSG("CoCreateInstance() failed,
code:" <<HRESULT_CODE(hres)<<"
facility:" <<HRESULT_FACILITY(hres)<<
" hres:" <<hres<< " time
waiting:" <<
(((endTime.time -
startTime.time)*1000)+
(endTime.millitm -
startTime.millitm))/1000.0)<<endl);

        DEBUGMSG("CoCreateInstance() failed,
code:" <<HRESULT_CODE(hres)<<"
facility:" <<HRESULT_FACILITY(hres)<<
" hres:" <<hres<< " time
waiting:" <<
(((endTime.time -
startTime.time)*1000)+
(endTime.millitm -
startTime.millitm))/1000.0)<<endl);

        return(ERR);
    };
    _ftime(&endTime);
    DEBUGMSG("CoCreateInstance successful.txnHande
com initialized, time waiting for object to be activated:" <<
(((endTime.time - startTime.time)*1000)+

```

```

        (endTime.millitm -
startTime.millitm))/1000.0)<<endl);
        //call set complete to return object to pool.
        (*txnHandle)->comInterface.comHandle->doSetComplete();
        //set the com buffers size
        DEBUGMSG("Setting txnHandle: " <<
DEBUGADDRESS(*txnHandle) << "com buffer size to " << maxDataSize<<
endl)
        (*txnHandle)->comInterface.size = maxDataSize;
        DEBUGMSG("txnHandle:
" <<DEBUGADDRESS(*txnHandle) <<"set to " << maxDataSize << endl);
        TlsSetValue(threadLSIndex, *txnHandle);
        DEBUGMSG("txnHandle:
" <<DEBUGADDRESS(*txnHandle) << "stored in TLS" << endl);

ZeroMemory((*txnHandle)->htmlPage, MAX_HTML_PAGE_LEN);

ZeroMemory((*txnHandle)->htmlHeader, MAX_HTML_HEADER_LEN);

        LeaveCriticalSection(&isapiLock);
        return(OK);
    }
    catch(...)
    {
        DEBUGMSG("Unhandled exeception in initTxnHandle,
unlocking isapi lock" <<endl);
        ERRORMSG("Unhandled exeception in initTxnHandle,
unlocking isapi lock" <<endl);
        LeaveCriticalSection(&isapiLock);
    };
    return ERR;
}
/*
*****
** Name          :          getDBInstance
** Description    :
**               :          load db specific lib
based on dbType registry
**               :          value.
** Parameters    :
**               :
** Returns       :
**               :          int - return code
** Comments      :
**               :          This function only
exists for the dlvy threads
**               :          Dlvy threads hold
direct connections to the database
**               :          and therefore need to
know what db interface to talk to.
*****
*/
int getDBInstance()
{
    if(nullDB)
    {
        dbInstance =
LoadLibrary("c:\\inetpub\\wwwroot\\tpcc\\nullDB.dll");
        if(dbInstance == NULL)
        {
            return ERR_NULL_DLL_NOT_LOADED;
        }
    }
    else if( (strcmp(dbType, "DB2") == 0) )

```

```

    {
        dbInstance =
LoadLibrary("c:\\inetpub\\wwwroot\\tpcc\\tpccDB2glue.dll");
        if(dbInstance == NULL)
            {
                return ERR_DB2_DLL_NOT_LOADED;
            }
        else if( (strcmp(dbType,"ORACLE") == 0) )
            {
                return ERR_ORACLE_DLL_NOT_LOADED;
            }
        else
            {
                return ERR_UNKNOWN_DB;
            }
        db_connect =
(CONNECT_PTR)GetProcAddress(dbInstance,"connect_db");
        if(db_connect == NULL)
            {
                return ERR_CONNECT_ADDRESS_NOT_FOUND;
            }
        dlvyCall =
(DLVY_FUNC_PTR)GetProcAddress(dbInstance,"do_dlvy");
        if(dlvyCall == NULL)
            {
                return ERR_DLVY_ADDRESS_NOT_FOUND;
            }

        return OK;
    }
/*
*****
** Name          :          initDlvy
** Description   :
**              initialize dlvy
threads/dlvy queueu
** Parameters   :
**
** Returns      :
**              int - return code
** Comments     :
**
*****
*/
int initDlvy()
{
    // Initialize critical section
    InitializeCriticalSection(&dlvyQueueLock);
    //create dlvy queue
    dlvyQueue = (DLVYQUEUEDATA *)
calloc(dlvyQueueLen,sizeof(DLVYQUEUEDATA));
    dlvyThreadDone = CreateEvent(NULL,

TRUE,          //manual reset

FALSE, //initially not signalled.

NULL);
    if(dlvyThreadDone == NULL)
        {
            DEBUGMSG("Error: dlvyThreadDone handled init
failed, GetLastError:"<<GetLastError()<<endl);
            ERRORMSG("Error : dlvyThreadDone handled init
failed, GetLastError:"<<GetLastError()<<endl);

            return ERR_DLVY_EVENT_INIT_FAILED;
        }
}

```

```

//create dlvy semaphore
dlvyThreadSemaphore =
CreateSemaphore(NULL,0,dlvyQueueLen,NULL);
if(dlvyThreadSemaphore == NULL)
    {
        DEBUGMSG("Error: dlvyThreadSemaphore semaphore
init failed, GetLastError:"<<GetLastError()<<endl);
        ERRORMSG("Error: dlvyThreadSemaphore semaphore
init failed, GetLastError:"<<GetLastError()<<endl);
        return ERR_DLVY_SEMAPHORE_INIT_FAILED;
    }
//set number of free slots available in queue
dlvyBufferFreeSlots = dlvyQueueLen;

//index into next available slot in dlvy txn queue
dlvyBufferSlotIndex = 0;

//thread index into dlvy txn queue
dlvyBufferThreadIndex = 0;
dlvyThreadHandles = new HANDLE[dlvyThreads];
//create threads
for(int threadCount = 0;threadCount < dlvyThreads;threadCount++)
    {
        dlvyThreadHandles[threadCount] =
(HANDLE)_beginthread(dlvyThreadEntry,0,NULL);
        if(dlvyThreadHandles[threadCount] ==
INVALID_HANDLE_VALUE)
            return ERR_DLVY_THREAD_FAILED;
    }
    return OK;
}
/*
*****
** Name          :          readRegistryValues
** Description   :
**              initialize isapi global
variables from registry
** Parameters   :
**
** Returns      :
**              int - return code
** Comments     :
**
*****
*/
int readRegistryValues()
{
    HKEY    registryKey;
    char    value[MAX_STRING_LEN];
    DWORD   regType;
    DWORD   regValue;
    DWORD   regValueSize = MAX_STRING_LEN;

    //open up registry key

if(RegOpenKeyEx(HKEY_LOCAL_MACHINE,REGISTRY_SUB_KEY,0,K
EY_READ,&registryKey) != ERROR_SUCCESS)
    return ERR_UNABLE_TO_OPEN_REG;

    //get null db flag
    regValueSize = sizeof(regValue);
    if(RegQueryValueEx(registryKey,NULL_DB,0,&regType,(BYTE
*)&regValue,&regValueSize) == ERROR_SUCCESS)
        nullIDB = regValue;
    else
        nullIDB = 0;

    //get num dlvy threads

```

```

        regValueSize = sizeof(regValue);

if(RegQueryValueEx(registryKey,DELIVERY_THREADS,0,&regType,(BYTE
E *)&regValue,&regValueSize) == ERROR_SUCCESS)
        dlvyThreads = regValue;
    else
        dlvyThreads =
DEFAULT_DLVI_THREADS;
    //get dlvy queue len
    regValueSize = sizeof(regValue);

if(RegQueryValueEx(registryKey,DELIVERY_QUEUE_LEN,0,&regType,(B
YTE *)&regValue,&regValueSize) == ERROR_SUCCESS)
        dlvyQueueLen = regValue;
    else
        dlvyQueueLen =
DEFAULT_DLVI_QUEUE_LEN;
    //get the htmlTrace flag
    regValueSize = sizeof(regValue);

if(RegQueryValueEx(registryKey,HTML_TRACE,0,&regType,(BYTE
*)&regValue,&regValueSize) == ERROR_SUCCESS)
        trace = regValue;
    else
        trace = 0;
    //get the client null db flag
    regValueSize = sizeof(regValue);
if(RegQueryValueEx(registryKey,NULL_DB,0,&regType,(BYTE
*)&regValue,&regValueSize) == ERROR_SUCCESS)
        nullDB = regValue;
    else
        nullDB = 0;
    //get the num of users
    regValueSize = sizeof(regValue);

if(RegQueryValueEx(registryKey,NUM_USERS,0,&regType,(BYTE
*)&regValue,&regValueSize) == ERROR_SUCCESS)
        numUsers = regValue;
    else
        numUsers = DEFAULT_NUM_USERS;
    //get dlvy log file path
    regValueSize = sizeof(value);
    if
(RegQueryValueEx(registryKey,DELIVERY_LOG_PATH,0,&regType,(BYT
E *)&value,&regValueSize) == ERROR_SUCCESS )
        strcpy(dlvyLogPath,value);
    else
        strcpy(dlvyLogPath,DEFAULT_DLVI_LOG_PATH);
    //get global error log file path/name
    regValueSize = sizeof(value);
    if
(RegQueryValueEx(registryKey,ERROR_LOG_FILE,0,&regType,(BYTE *)
&value,&regValueSize) == ERROR_SUCCESS )
        strcpy(errorLogFile,value);
    else
        strcpy(errorLogFile,DEFAULT_ERROR_LOG_FILE);
    //get global error log file path/name
    regValueSize = sizeof(value);
    if
(RegQueryValueEx(registryKey,HTML_TRACE_LOG_FILE,0,&regType,(B
YTE *)&value,&regValueSize) == ERROR_SUCCESS )
        strcpy(htmlTraceLogFile,value);
    else

strcpy(htmlTraceLogFile,DEFAULT_HTML_TRACE_LOG_FILE);
    //get db name
    regValueSize = sizeof(value);

```

```

        if(RegQueryValueEx(registryKey,DB_NAME,0,&regType,(BYTE
*)&value,&regValueSize) == ERROR_SUCCESS )
            strcpy(dbName,value);
        else
            strcpy(dbName,DEFAULT_DB_NAME);
    //get db type
    regValueSize = sizeof(value);
if(RegQueryValueEx(registryKey,DB_TYPE,0,&regType,(BYTE
*)&value,&regValueSize) == ERROR_SUCCESS )
        strcpy(dbType,value);
    RegCloseKey(registryKey);

    return OK;
}
/*
*****
** Name                : doLoginForm
** Description         :
**                    : HTML Login page entry point
** Parameters         :
**                    : htmlPhraser*      command
block
**                    : TXN_HANDLE*      txn handle
struct
** Returns            :
**                    : int - return code
** Comments          :
**
*****
*/
int doLoginForm(htmlPhraser *commandBlock, TXN_HANDLE *txnHandle)
{
    DEBUGMSG("Entering doLoginForm()."<<endl);
    char *html=txnHandle->htmlPage;
    DEBUGMSG("Creating html login page"<<endl);
    //begin html page
    appendText(&html,"<HTML><HEAD><TITLE>TPC-C Client
Home Page</TITLE></HEAD>"
                "
                "<FORM
                "
                "APP_NAME
                "
                "METHOD=\"GET\">"
                "
                "<H2>Please
                "
                "Login.</H2>"
                "
                "<INPUT
                "
                "TYPE=\"hidden\" NAME=\""
                "
                "CMD_TXN_ID
                "
                "\" VALUE=\""
                "
                "CMD_MENU
                "
                "\">"
                "
                "<H3>Warehouse
                "
                "
                "<INPUT NAME=\""
                "
                "CMD_W_ID
                "
                "\" SIZE=6>"
                "
                "District <INPUT
                "
                "NAME=\""
                "
                "CMD_D_ID
                "
                "\" SIZE=2></H3>"
                "
                "<INPUT
                "
                "TYPE=\"submit\" VALUE=\"Submit\">"
                "
                "</FORM>");

html+=sprintf(html,"dlvy Queue Length:%d <BR> num dlvy threads:%d <BR>
dlvy queue free slots:%d <BR> isapi queue index:%d <BR> thread queue
index:%d <BR> </BODY></HTML>\n",
                dlvyQueueLen,
                dlvyThreads,

```

```

                dlvyBufferFreeSlots,
                dlvyBufferSlotIndex,
                dlvyBufferThreadIndex);
DEBUGMSG("Html login page done"<<endl);
return OK;
}
/*
*****
** Name          : doLoginResults
** Description   :
**              : HTML Login results page entry
point
** Parameters    :
**              : htmlPhraser*      command
block
**              : TXN_HANDLE*      txn handle
struct
** Returns      :
**              : int - return code
** Comments     :
**
*****
*/
int doLoginResults(htmlPhraser *commandBlock, TXN_HANDLE *txnHandle)
{
    char *html=txnHandle->htmlPage;

    //validate parameters
    if( (txnHandle->w_id = atoi(commandBlock->get_W_ID())) == 0 )
    {
        doLoginErrorPage(html,ERR_INVALID_W_ID);
        return OK;
    }
    if( (txnHandle->d_id = atoi(commandBlock->get_D_ID())) == 0 )
    {
        doLoginErrorPage(html,ERR_INVALID_D_ID);
        return OK;
    }
    //store user into terminal array,
    //function will ERR if the terminal array is full
    if( assignTerminal(txnHandle) != OK)
    {
        doLoginErrorPage(html,ERR_TERMINAL_FULL);
        return OK;
    };
    appendText(&html,"<HTML><HEAD><TITLE>TPC-C Main
Menu</TITLE></HEAD>\r\n"
                " <BODY><FORM
ACTION=\\"
                APP_NAME
                "\\"
METHOD=\\"GET\>\r\n"
                " <H3>Please Select
Transaction.</H3>\r\n");
    html+=appendButtons(html);
    html+=appendHiddenFields(html,txnHandle);
    appendText(&html,"</FORM></BODY></HTML>");
    return OK;
}
/*
*****
** Name          : doLoginErrorPage
** Description   :
**              : HTML Login page entry point
** Parameters    :
**              : char *          html page
buffer

```

```

**              : char *          error
message
** Returns      :
**              : int - return code
** Comments     :
**
*****
*/
int doLoginErrorPage(char *htmlPage,char *errorMessage)
{
    char *html=htmlPage;
    //begin html page
    appendText(&html,"<HTML><HEAD><TITLE>TPC-C Client
Home Page</TITLE></HEAD>"
                " <FORM
ACTION=\\"
                APP_NAME
                "\\"
METHOD=\\"GET\>");
    appendText(&html,"<H2>Please Login.</H2>"
                " <INPUT
TYPE=\\"hidden\> NAME=\\"
                CMD_TXN_ID
                "\\" VALUE=\\"
                CMD_MENU
                "\\">"
                " <H3>Warehouse
<INPUT NAME=\\"
                CMD_W_ID
                "\\" SIZE=6>"
                " District <INPUT
TYPE=\\"submit\> VALUE=\\"Submit\>"
                " </FORM>");
    appendText(&html,errorMessage);
    appendText(&html,"<BODY></HTML>");
    return OK;
}
/*
*****
** Name          : doNewOrderForm
** Description   :
**              : HTML neworder page entry point
** Parameters    :
**              : htmlPhraser*      command
block
**              : TXN_HANDLE*      txn handle
struct
** Returns      :
**              : int - return code
** Comments     :
**
*****
*/
int doNewOrderForm(htmlPhraser *commandBlock, TXN_HANDLE
*txnHandle)
{
    char *html=txnHandle->htmlPage;
    appendText(&html,"<HTML><HEAD><TITLE>TPC-C New
Order</TITLE></HEAD>\r\n"
                " <BODY><FORM
ACTION=\\"
                APP_NAME

```



```

//missing previous value of item supp
warehouse, flag error
else
{
doNewOrderErrorPage(html,ERR_INVALID_SUPPLY_W_ID,commandBlock
,txnHandle);
return OK;
}
}
else if( (itemComplete==1) )//nothing in the command
block, check to see if the previous item value is present
{
doNewOrderErrorPage(html,ERR_INVALID_ITEM_NUM,commandBlock,txn
Handle);
return OK;
}
//item qty
if*(commandBlock->get_ITEM_QTY(itemIndex))
{
if(itemComplete==2)
{
if(
(nord->in_nord.in_item[nord->in_nord.s_O_OL_CNT].s_OL_QUANTITY =
atoi(commandBlock->get_ITEM_QTY(itemIndex))) == 0)
{
doNewOrderErrorPage(html,ERR_INVALID_ITEM_OTY,commandBlock,txn
Handle);
return OK;
}
else
itemComplete++;
}
//missing previous value of item number
else if (itemComplete ==1)
{
doNewOrderErrorPage(html,ERR_INVALID_ITEM_NUM,commandBlock,txn
Handle);
return OK;
}
//missing 1st value of supp warehouse
else
{
doNewOrderErrorPage(html,ERR_INVALID_SUPPLY_W_ID,commandBlock
,txnHandle);
return OK;
}
}
else if(itemComplete==2) //nothing in
the command block, check to see if the previous item values are present
{
doNewOrderErrorPage(html,ERR_INVALID_ITEM_NUM,commandBlock,txn
Handle);
return OK;
}
return OK;
}
DEBUGMSG("nord item:" <<
nord->in_nord.s_O_OL_CNT << "SUPPLY_W_ID:" <<
nord->in_nord.in_item[nord->in_nord.s_O_OL_CNT].s_OL_SUPPLY_W_ID
<<
" OL_I_ID:" <<
nord->in_nord.in_item[nord->in_nord.s_O_OL_CNT].s_OL_I_ID << "

```

```

OL_QUANTITY:" <<
nord->in_nord.in_item[nord->in_nord.s_O_OL_CNT].s_OL_QUANTITY
<<endl);
if(itemComplete == 3)
nord->in_nord.s_O_OL_CNT++;
itemComplete=0;
}
DEBUGMSG("complete nord
items:"<<nord->in_nord.s_O_OL_CNT<<" initializing remainig unused items "
<< "NORD_ITEMS - nord->in_nord.s_O_OL_CNT << " to 0" <<endl);
for(int
itemIndex=nord->in_nord.s_O_OL_CNT;itemIndex<NORD_ITEMS;itemIndex
++)
{
nord->in_nord.in_item[itemIndex].s_OL_SUPPLY_W_ID=0;
nord->in_nord.in_item[itemIndex].s_OL_I_ID = 0;
nord->in_nord.in_item[itemIndex].s_OL_QUANTITY
=0;
}
DEBUGMSG("nord creating new order results html title page"
<<endl);
appendText(&html,"<HTML><HEAD><TITLE>TPC-C New Order
Results</TITLE></HEAD>\r\n"
" <BODY><FORM
ACTION=\"\"
APP_NAME
\"\"
METHOD=\"GET\">\r\n");
//append menu buttons
html+=appendButtons(html);
html+=appendHiddenFields(html,txnHandle);
appendText(&html,"</FORM><CENTER><H3>New Order</H3>
<BR></CENTER>"
" <PRE>"
" 1 2 3
4 5 6 7 8 9\r\n"
"123456789012345678901234567890123456789012345678901234567890123
456789012345678901234567890\r\n
\"");
//assume failure
nord->out_nord.s_transtatus = -1;
DEBUGMSG("nord executing COM interface function" << endl);
HRESULThres;
try
{
hres =
txnHandle->comInterface.comHandle->doNewOrder(&txnHandle->comInterfa
ce.size,(UCHAR**) &txnHandle->comInterface.txnBuffer);
}
catch(...)
{
html+=sprintf(html,"ERROR: nord com call caused
exeception to occur.</PRE></BODY></HTML>");
ERRORMSG("ERROR : nord com call cause exeception
to occur,"<<endl);
return OK;
}
if(FAILED(hres))
{
ERRORMSG("ERROR : nord com call failed, rc:" <<
hex << hres);
DEBUGMSG("ERROR : nord com call failed, rc:" <<
hex << hres);
return OK;
}
}

```

```

//com call successful, return object back to pool.
hres = txnHandle->comInterface.comHandle->doSetComplete();
if(FAILED(hres))
{
    ERRORMSG("ERROR : nord setcomplete call failed,
rc:" << hex << hres);
    DEBUGMSG("ERROR : nord setcomplete call failed,
rc:" << hex << hres);
}
nord = (nord_wrapper *)txnHandle->comInterface.txnBuffer;
if(FAILED(hres))
{
    html+=sprintf(html,"ERROR: nord com doSetComplete
failed, rc:%ld</PRE></BODY></HTML>",hres);
    ERRORMSG("ERROR : nord com doSetComplete
failed, rc:"<<DEBUGADDRESS(hres)<<endl);
    return OK;
}
DEBUGMSG("nord COM interface function successful,
s_transtatus:" << nord->out_nord.s_transtatus << endl);
int rc = nord->out_nord.s_transtatus;
char buffer[10];
appendText(&html,"Warehouse: ");
appendText(&html,ittoa(nord->in_nord.s_W_ID,buffer,10),6,1);
appendText(&html,"District: ");
appendText(&html,ittoa(nord->in_nord.s_D_ID,buffer,10),26,1);
appendText(&html,"Date: ");
if(rc == OK)
{
    char dateTimeBuffer[50];

copyOutDateTime(dateTimeBuffer,nord->out_nord.s_O_ENTRY_D_time);
    appendText(&html,dateTimeBuffer);
}
appendText(&html," <BR>"
"Customer: ");
appendText(&html,ittoa(nord->in_nord.s_C_ID,buffer,10),8,1);
appendText(&html,"Name: ");

appendText(&html,nord->out_nord.s_C_LAST,LAST_NAME_LEN+3,1);
appendText(&html,"Credit: ");
appendText(&html,nord->out_nord.s_C_CREDIT,5,1);

appendText(&html,"%Disc.: ");
if(rc == OK)
{
html+=sprintf(html,"%2.2lf",nord->out_nord.s_C_DISCOUNT/100.0);
}
appendText(&html," <BR>"
"Order Number: ");
if(rc != INVALID_STATUS)

appendText(&html,ittoa(nord->out_nord.s_O_ID,buffer,10),10,1);

appendText(&html,"Number of Lines: ");

if(rc != INVALID_STATUS)

appendText(&html,ittoa(nord->out_nord.s_O_OL_CNT,buffer,10),10,1);
appendText(&html,"W_Tax: ");
if(rc == OK)
{

html+=sprintf(html,"%5.2lf",nord->out_nord.s_W_TAX/100.0);
}
appendText(&html," D_Tax: ");

```

```

if(rc == OK)
{
html+=sprintf(html,"%5.2lf",nord->out_nord.s_D_TAX/100.0);
}
appendText(&html," <BR> <BR>"
" 1 2 3 4 5 6
// 7 8 9\r\n"
//
"123456789012345678901234567890123456789012345678901234567890123
456789012345678901234567890\r\n"
" Supp_W Item_Id Item_Name
Qty Stock B/G Price Amount <BR>");

//display items
if (rc == OK)
{
//display valid items
for(int itemCount=0;itemCount <
nord->out_nord.s_O_OL_CNT;itemCount++)
{
appendText(&html,ittoa(nord->in_nord.in_item[itemCount].s_OL_SUPPLY_W
_ID,buffer,10),8,1);

appendText(&html,ittoa(nord->in_nord.in_item[itemCount].s_OL_I_ID,buffer,1
0),10,1);

appendText(&html,nord->out_nord.item[itemCount].s_I_NAME,DEFAULT_S
TRING_LEN+1,1);

appendText(&html,ittoa(nord->in_nord.in_item[itemCount].s_OL_QUANTITY,
buffer,10),5,1);

appendText(&html,ittoa(nord->out_nord.item[itemCount].s_S_QUANTITY,buf
fer,10),7,1);
html+=sprintf(html,"%c $%-7.2lf $%-7.2lf
<BR> ",nord->out_nord.item[itemCount].s_brand_generic,
nord->out_nord.item[itemCount].s_I_PRICE/100.0,
nord->out_nord.item[itemCount].s_OL_AMOUNT/100.0);
}
//display blank line for remaining empty items in the
order
for(int lineBreaks=0;lineBreaks <
(NORD_ITEMS-nord->out_nord.s_O_OL_CNT);lineBreaks++)
appendText(&html," <BR>");
}
else
appendText(&html," <BR> <BR> <BR> <BR> <BR>
<BR> <BR> <BR> <BR> <BR> <BR> <BR> <BR> <BR>");
appendText(&html,"\r\n <BR> ");
html+=displayStatus(html,rc);
if(rc == OK)
html+=sprintf(html," Total:
$%.2lf",nord->out_nord.s_total_amount/100.0);
else
appendText(&html," Total: <BR>");

appendText(&html,"</PRE></BODY> </HTML>");
DEBUGMSG("nord html page complete. returning to calling
function" << endl);
return OK;
}
/*
*****
** Name : doNewOrderErrorPage
** Description :
** HTML neworder page entry point

```

```

** Parameters      :
**                char *          html result
page
**                char *          error
message
**                htmlPhraser*    command block
**                TXN_HANDLE*    txn handle
struct
** Returns        :
**                int - return code
** Comments       :
**
*****
*/

int doNewOrderErrorPage(char *htmlPage,char *message,htmlPhraser
*commandBlock, TXN_HANDLE *txnHandle)
{
    char *html=htmlPage;
    appendText(&html,"<HTML><HEAD><TITLE>TPC-C New
Order</TITLE></HEAD>\r\n"
                "<BODY><FORM
ACTION=\\\"
                APP_NAME
                \\\"
METHOD=\\\"GET\\\">\r\n"
                "<CENTER><H3>Please Fill In New Order Form.</H3></CENTER>\r\n"
                "Submit Transaction
<INPUT TYPE=\\\"submit\\\" NAME=\\\"
                CMD_TXN_ID
                \\\" VALUE=\\\"
                CMD_NORD
                \\\">");
    //append the hidden warehouse and district fields
    html+=appendHiddenFields(html,txnHandle);
    //int buffer for warehouse
    char buffer[15];
    /*appendText(&html,"<PRE>
6 7 8 9\r\n"
                "123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890\r\n"
                "Warehouse: ");*/
    appendText(&html,"<PRE>Warehouse: ");
    appendText(&html,ittoa(txnHandle->w_id,buffer,10),7,1);
    appendText(&html,"District: <INPUT NAME=\\\"
                CMD_D_ID
                \\\" SIZE=1>
                Date:<BR>
                \"Customer <INPUT NAME=\\\"
                CMD_C_ID
                \\\" SIZE=6> Name:
                Credit: %Disc.:<BR>
                \"Order Number:
                Number of Lines: W_tax: D_tax:<BR> <BR>
                //\" 1 2 3
                4 5 6 7 8 9\r\n"
                //"12345678901234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890\r\n"
                " Supp_W Item_Num
                Item_Name Qty Stock B/G Price Amount<BR> ");
    //append the 15 items commands
    html+=appendItems(html,NORD_ITEMS,ITEM_START);
    appendText(&html,message);

```

```

//seal up html page
appendText(&html,"</PRE></BODY></HTML>");
return OK;
}
/*
*****
** Name          : doPaymentForm
** Description   :
**              HTML payment page entry point
** Parameters    :
**              htmlPhraser*    command
**              TXN_HANDLE*    txn handle
struct
** Returns      :
**              int - return code
** Comments     :
**
*****
*/
int doPaymentForm(htmlPhraser *commandBlock, TXN_HANDLE
*txnHandle)
{
    char *html=txnHandle->htmlPage;
    appendText(&html,"<HTML><HEAD><TITLE>TPC-C
Payment</TITLE></HEAD>\r\n"
                "<BODY><FORM
ACTION=\\\"
                APP_NAME
                \\\"
METHOD=\\\"GET\\\">\r\n"
                "<CENTER><H3>Please Fill In Payment Form.</H3></CENTER> <BR>\r\n"
                "Submit Transaction
                CMD_TXN_ID
                \\\" VALUE=\\\"
                CMD_PYMT
                \\\">");
    html+=appendHiddenFields(html,txnHandle);
    appendText(&html,"<BR><PRE>\r\n"
                "Date:<BR>
                \"Warehouse: ");
    char buffer[15];
    appendText(&html,ittoa(txnHandle->w_id,buffer,10));
    appendSpaces(&html,10);
    appendText(&html,"District: <INPUT NAME=\\\"
                CMD_D_ID
                \\\" SIZE=1>\r\n<BR>
                \"<BR> <BR> <BR>
                \"Customer: \"
                \"<INPUT NAME=\\\"
                CMD_C_ID
                \\\" SIZE=5>
                \"
                \"Cust-Warehouse: \"
                \"<INPUT NAME=\\\"
                CMD_C_W_ID
                \\\" SIZE=5>
                \"
                \"Cust-District: \"
                \"<INPUT NAME=\\\"
                CMD_C_D_ID
                \\\" SIZE=1><BR>

```



```

Name: <INPUT
NAME=""
appendText(&html,"
Credit: <BR>"
%Disc: <BR>"
Cust-Balance:<BR>"
<BR>Cust-Data:<BR> <BR> <BR> <BR> </PRE>");
return OK;
}
/*
*****
** Name          : doPaymentResults
** Description   :
**               HTML neworder page entry point
** Parameters    :
**               htmlPhraser*   command
block
**               TXN_HANDLE*   txn handle
struct
** Returns      :
**               int - return code
** Comments     :
**
*****
*/
int doPaymentResults(htmlPhraser *commandBlock, TXN_HANDLE
*txnHandle)
{
char *html=txnHandle->htmlPage;
char buffer[50];
struct paym_wrapper *pymt = NULL;
pymt = (paym_wrapper*)txnHandle->comInterface.txnBuffer;
ZeroMemory(pymt,maxDataSize);

//set login warehouse id from command block
pymt->in_paym.s_W_ID = txnHandle->w_id;
//set district from command block
if( pymt->in_paym.s_D_ID = atoi(commandBlock->get_D_ID()))
== 0)
{
doPaymentErrorPage(html,ERR_INVALID_D_ID,commandBlock,txnHandle);
return OK;
}

//set customer id from command block
if( pymt->in_paym.s_C_ID = atoi(commandBlock->get_C_ID()))
== 0)
{
if(*(commandBlock->get_C_NAME()) == NULL)
{
//no customer id nor customer last name
specified.
}
}
}

doPaymentErrorPage(html,ERR_MISSING_C_ID_OR_CLAST,commandBlock,txnHandle);
return OK;
}
else
strcpy(pymt->in_paym.s_C_LAST,commandBlock->get_C_NAME());
}
else
{
//make sure that the user only inserted just c_id
if(*(commandBlock->get_C_NAME()) != NULL)
{
doPaymentErrorPage(html,ERR_C_ID_OR_CLAST_ONLY,commandBlock,txnHandle);
return OK;
}
}
//get customer warehouse id field
if( pymt->in_paym.s_C_W_ID =
atoi(commandBlock->get_C_W_ID())) == 0)
{
doPaymentErrorPage(html,ERR_INVALID_C_W_ID,commandBlock,txnHandle);
return OK;
}
//get customer district id field
if( ( pymt->in_paym.s_C_D_ID =
atoi(commandBlock->get_C_D_ID())) == 0)
{
doPaymentErrorPage(html,ERR_INVALID_C_D_ID,commandBlock,txnHandle);
return OK;
}
}
if(!copyInMoney64(commandBlock->get_AMT_PAID(),&pymt->in_paym.s_H
_AMOUNT))
{
doPaymentErrorPage(html,ERR_INVALID_PAYMENT_AMOUNT,commandBlock,txnHandle);
return OK;
}
appendText(&html,"<HTML><HEAD><TITLE>TPC-C Payment
Results</TITLE></HEAD>\r\n"
" <BODY><FORM
ACTION=""
APP_NAME
""
METHOD=""GET"">\r\n");
html+=appendButtons(html);
html+=appendHiddenFields(html,txnHandle);

appendText(&html,"</FORM><CENTER><H3>Payment</H3></CENTER>");
;

DEBUGMSG("Calling com entry api payment,
w_id:"<<pymt->in_paym.s_W_ID<<"
d_id:"<<pymt->in_paym.s_D_ID<<endl);
//assume failure
pymt->out_paym.s_transtatus = -1;
HRESULThres;
try
{

```

```

    hres =
txnHandle->comInterface.comHandle->doPayment(&txnHandle->comInterface
.size,(UCHAR**)&txnHandle->comInterface.txnBuffer);
    }
    catch(...)
    {
        html+=sprintf(html,"ERROR: Com Payment call caused
exception to occur.</PRE></BODY></HTML>");
        ERRORMSG("ERROR : Com Payment call caused
exception to occur."<<endl);
        return OK;
    }
    if(FAILED(hres))
    {
        html+=sprintf(html,"ERROR: pymt com call failed,
rc:%x</PRE></BODY></HTML>",hres);
        ERRORMSG("ERROR : pymt com call failed,
rc:"<<hres<<endl);
        return OK;
    }
    hres = txnHandle->comInterface.comHandle->doSetComplete();
    if(FAILED(hres))
    {
        html+=sprintf(html,"ERROR: pymt com doSetComplete
failed, rc:%ld</PRE></BODY></HTML>",hres);
        ERRORMSG("ERROR : pymt com doSetComplete
failed, rc:"<<DEBUGADDRESS(hres)<<endl);
        return OK;
    }
    pymt = (pymt_wrapper *)txnHandle->comInterface.txnBuffer;
    //get return code
    int rc = pymt->out_paym.s_transtatus;
    if( rc != OK)
    {
        html+=displayStatus(html,rc);
        appendText(&html,"</PRE></BODY></HTML>");
        ERRORMSG("Payment TXN ERROR"<<endl
<<"pymt->in_paym.s_C_D_ID:"<<pymt->in_paym.s_C_D_ID<<endl
<<"pymt->in_paym.s_C_ID:"<<pymt->in_paym.s_C_ID<<endl
<<"pymt->in_paym.s_C_LAST:"<<pymt->in_paym.s_C_LAST<<endl
<<"pymt->in_paym.s_C_W_ID:"<<pymt->in_paym.s_C_W_ID<<endl
<<"pymt->in_paym.s_D_ID:"<<pymt->in_paym.s_D_ID<<endl
<<"pymt->in_paym.s_H_AMOUNT:"<<pymt->in_paym.s_H_AMOUNT<<en
dl
<<"pymt->in_paym.s_H_DATE_time:"<<pymt->in_paym.s_H_DATE_time<<
endl
<<"pymt->in_paym.s_W_ID:"<<pymt->in_paym.s_W_ID<<endl
<<"pymt->out_paym.deadlocks:"<<pymt->out_paym.deadlocks<<endl
<<"pymt->out_paym.s_C_BALANCE:"<<pymt->out_paym.s_C_BALANCE<
endl
<<"pymt->out_paym.s_C_CITY:"<<pymt->out_paym.s_C_CITY<<endl
<<"pymt->out_paym.s_C_CREDIT:"<<pymt->out_paym.s_C_CREDIT<<endl
<<"pymt->out_paym.s_C_CREDIT_LIM:"<<pymt->out_paym.s_C_CREDIT
LIM<<endl
<<"pymt->out_paym.s_C_DATA:"<<pymt->out_paym.s_C_DATA<<endl
<<"pymt->out_paym.s_C_DISCOUNT:"<<pymt->out_paym.s_C_DISCOUNT
<<endl
<<"pymt->out_paym.s_C_FIRST:"<<pymt->out_paym.s_C_FIRST<<endl
<<"pymt->out_paym.s_C_ID:"<<pymt->out_paym.s_C_ID<<endl
<<"pymt->out_paym.s_C_LAST:"<<pymt->out_paym.s_C_LAST<<endl
<<"pymt->out_paym.s_C_MIDDLE:"<<pymt->out_paym.s_C_MIDDLE<<end
l
<<"pymt->out_paym.s_C_PHONE:"<<pymt->out_paym.s_C_PHONE<<endl
<<"pymt->out_paym.s_C_SINCE_time:"<<pymt->out_paym.s_C_SINCE_time
<<endl
<<"pymt->out_paym.s_C_STATE:"<<pymt->out_paym.s_C_STATE<<endl
<<"pymt->out_paym.s_C_STREET_1:"<<pymt->out_paym.s_C_STREET_1<
endl
<<"pymt->out_paym.s_C_STREET_2:"<<pymt->out_paym.s_C_STREET_2<
endl
<<"pymt->out_paym.s_C_ZIP:"<<pymt->out_paym.s_C_ZIP<<endl
<<"pymt->out_paym.s_D_CITY:"<<pymt->out_paym.s_D_CITY<<endl
<<"pymt->out_paym.s_D_STATE:"<<pymt->out_paym.s_D_STATE<<endl
<<"pymt->out_paym.s_D_STREET_1:"<<pymt->out_paym.s_D_STREET_1<
endl
<<"pymt->out_paym.s_D_STREET_2:"<<pymt->out_paym.s_D_STREET_2<
endl
<<"pymt->out_paym.s_D_ZIP:"<<pymt->out_paym.s_D_ZIP<<endl
<<"pymt->out_paym.s_H_DATE_time:"<<pymt->out_paym.s_H_DATE_time
<<endl
<<"pymt->out_paym.s_transtatus:"<<pymt->out_paym.s_transtatus<<endl
<<"pymt->out_paym.s_W_CITY:"<<pymt->out_paym.s_W_CITY<<endl
<<"pymt->out_paym.s_W_STATE:"<<pymt->out_paym.s_W_STATE<<endl
<<"pymt->out_paym.s_W_STREET_1:"<<pymt->out_paym.s_W_STREET_1
<<endl
<<"pymt->out_paym.s_W_STREET_2:"<<pymt->out_paym.s_W_STREET_2
<<endl
<<"pymt->out_paym.s_W_ZIP:"<<pymt->out_paym.s_W_ZIP<<endl);
        return OK;
    }
    //      appendText(&html, "<BR><PRE>\r\n");
    //      appendText(&html, " 1 2 3 4 5 6 7
8<BR>");
    //
    appendText(&html,"12345678901234567890123456789012345678901234567
890123456789012345678901234567890<BR>");
    //start creating result body
    appendText(&html, "<BR><PRE>\r\n"

```

```

                                "Date: ");
copyOutDateTime(buffer,pymt->out_paym.s_H_DATE_time);
appendText(&html,buffer);
appendText(&html,"<BR>"
"Warehouse: ");

appendText(&html,itoa(pymt->in_paym.s_W_ID,buffer,10),6+24,1);
appendText(&html,"District: ");
appendText(&html,itoa(pymt->in_paym.s_D_ID,buffer,10),2,1);
appendText(&html,"<BR>");
//print out warehouse and district information

appendText(&html,pymt->out_paym.s_W_STREET_1,STREET_LEN+21,1);

appendText(&html,pymt->out_paym.s_D_STREET_1,STREET_LEN,1);
appendText(&html,"<BR>");

appendText(&html,pymt->out_paym.s_W_STREET_2,STREET_LEN+21,1);

appendText(&html,pymt->out_paym.s_D_STREET_2,STREET_LEN,1);
appendText(&html,"<BR>");
appendText(&html,pymt->out_paym.s_W_CITY,CITY_LEN+1,1);

appendText(&html,pymt->out_paym.s_W_STATE,STATE_LEN+1,1);
copyOutZip(buffer,pymt->out_paym.s_W_ZIP);
appendText(&html,buffer);
appendText(&html,pymt->out_paym.s_D_CITY,CITY_LEN+1,1);

appendText(&html,pymt->out_paym.s_D_STATE,STATE_LEN+1,1);
copyOutZip(buffer,pymt->out_paym.s_D_ZIP);
appendText(&html,buffer);
//print out customer information
appendText(&html,"<BR><BR>Customer: ");
appendText(&html,itoa(pymt->out_paym.s_C_ID,buffer,10),5+1,1);
appendText(&html,"Cust-Warehouse: ");

appendText(&html,itoa(pymt->in_paym.s_C_W_ID,buffer,10),6+1,1);
appendText(&html,"Cust-District: ");
appendText(&html,itoa(pymt->in_paym.s_C_D_ID,buffer,10));
//add customer information
appendText(&html,"<BR>Name: ");

appendText(&html,pymt->out_paym.s_C_FIRST,FIRST_NAME_LEN+1,1);

appendText(&html,pymt->out_paym.s_C_MIDDLE,INITIALS_LEN+1,1);
DEBUGMSG("Last name:"<<pymt->out_paym.s_C_LAST<<endl);

appendText(&html,pymt->out_paym.s_C_LAST,LAST_NAME_LEN+5,1);
appendText(&html,"Since: ");
copyOutDateTime(buffer,pymt->out_paym.s_C_SINCE_time);
appendText(&html,buffer);
appendText(&html,"<BR>");
appendSpaces(&html,8);

appendText(&html,pymt->out_paym.s_C_STREET_1,STREET_LEN+20,1);
appendText(&html," Credit: ");
appendText(&html,pymt->out_paym.s_C_CREDIT);
appendText(&html,"<BR>");
appendSpaces(&html,8);

appendText(&html,pymt->out_paym.s_C_STREET_2,STREET_LEN+21,1);
appendText(&html,"%Disc: ");

html+=sprintf(html,"%02.2lf",pymt->out_paym.s_C_DISCOUNT/100.0);

appendText(&html,"<BR>");
appendSpaces(&html,8);

```

```

appendText(&html,pymt->out_paym.s_C_CITY,CITY_LEN+1,1);

appendText(&html,pymt->out_paym.s_C_STATE,STATE_LEN+1,1);
copyOutZip(buffer,pymt->out_paym.s_C_ZIP);
appendText(&html,buffer,15,1);

appendText(&html,"Phone: ");
copyOutPhone(buffer,pymt->out_paym.s_C_PHONE);
appendText(&html,buffer);
appendText(&html,"<BR><BR>Amount Paid: $");

html+=sprintf(html,"%-9.2lf",pymt->in_paym.s_H_AMOUNT/100.0);

appendText(&html," New Cust-Balance: $");

html+=sprintf(html,"%-9.2lf",pymt->out_paym.s_C_BALANCE/100.0);

appendText(&html,"<BR>Credit Limit: $");

html+=sprintf(html,"%-9.2lf",pymt->out_paym.s_C_CREDIT_LIM/100.0);

appendText(&html,"<BR><BR>Cust-Data: ");
if(pymt->out_paym.s_C_CREDIT[0] == 'B' &&
pymt->out_paym.s_C_CREDIT[1] == 'C')
{
    appendCustData(&html,pymt->out_paym.s_C_DATA);
    appendText(&html,"<BR>");
}
else
    appendText(&html,"<BR><BR><BR>");
html+=displayStatus(html,rc);
appendText(&html,"</PRE></BODY></HTML>");
return OK;
}
/*
*****
** Name          : doPaymentErrorPage
** Description   :
**               : append payment error body
** Parameters    :
**               : char *          html page
result
**               : char *          error
message
**               : htmlPhraser * command block
**               : TXN_HANDLE*   txn handle
struct
** Returns      :
**               : int - return code
** Comments     :
*****
*/
int doPaymentErrorPage(char *htmlPage,char *message,htmlPhraser
*commandBlock,TXN_HANDLE *txnHandle)
{
    char *html=htmlPage;
    appendText(&html,"<HTML><HEAD><TITLE>TPC-C
Payment</TITLE></HEAD>\r\n"
                                "<BODY><FORM
ACTION=\"\"
                                APP_NAME
                                \"\"
METHOD=\"GET\">\r\n"

"<CENTER><H3>Please Fill In Payment Form.</H3></CENTER><BR>\r\n"

```

<pre> <INPUT TYPE="submit" NAME="" </pre>	<pre> Submit Transaction CMD_TXN_ID "\ VALUE="" CMD_PYMT ">"; html+=appendHiddenFields(html,txnHandle); appendText(&html,"
<PRE>\r\n" >Date:
" "Warehouse: "); char buffer[15]; appendText(&html,itoa(txnHandle->w_id,buffer,10)); appendSpaces(&html,10); appendText(&html,"District: <INPUT NAME="" CMD_D_ID "\ SIZE=1>\r\n
" "

" "Customer: " "<INPUT NAME="" CMD_C_ID "\ SIZE=5>" " " "Cust-Warehouse: " "<INPUT NAME="" CMD_C_W_ID "\ SIZE=6>" " " "Cust-District: " "<INPUT NAME="" CMD_C_D_ID "\ SIZE=1>
" "Name: <INPUT NAME="" CMD_C_NAME "\ SIZE=20>"); appendText(&html," Since:
" " " " " " "Amount Paid: " "<INPUT NAME="" CMD_AMT_PAID "\ SIZE=10>" " " "New "Credit Limit:
"

 Cust-Data:

 "); appendText(&html,message); appendText(&html,"</PRE>"); return OK; } /* ***** ** Name : doOrderStatusForm ** Description : HTML orderStatus page entry ** Parameters : htmlPhraser* command ** block char * html result </pre>	<pre> TXN_HANDLE* txn handle struct ** Returns : int - return code ** Comments : ** ***** int doOrderStatusForm(htmlPhraser *commandBlock, TXN_HANDLE *txnHandle) { char *html=txnHandle->htmlPage; appendText(&html,"<HTML><HEAD><TITLE>TPC-C Order Status</TITLE></HEAD>\r\n" "<BODY><FORM ACTION="" APP_NAME " METHOD="GET">\r\n" "<CENTER><H3>Please Fill In Order Status Form.</H3></CENTER>
\r\n" "<INPUT TYPE="submit" NAME="" CMD_TXN_ID "\ VALUE="" CMD_ORDS ">" "
 "); html+=appendHiddenFields(html,txnHandle); appendText(&html,"<PRE>\r\n" "Warehouse: "); char buffer[15]; appendText(&html,itoa(txnHandle->w_id,buffer,10)); appendText(&html," District: <INPUT NAME="" CMD_D_ID "\ SIZE=1>\r\n
" "Customer: " "<INPUT NAME="" CMD_C_ID "\ SIZE=5>" " " "Cust-Warehouse: " "<INPUT NAME="" CMD_C_W_ID "\ SIZE=6>" " " "Cust-District: " "<INPUT NAME="" CMD_C_D_ID "\ SIZE=1>
" "Name: " CMD_C_NAME "\ SIZE=20>"); appendText(&html," Since:
" " " " " " "Amount Paid: " "<INPUT NAME="" CMD_AMT_PAID "\ SIZE=10>" " " "New "Credit Limit:
" "

 Cust-Data:

 "); appendText(&html,message); appendText(&html,"</PRE>"); return OK; } /* ***** ** Name : doOrderStatusResults ** Description : HTML orderStatus page entry ** Parameters : htmlPhraser* command ** block char * html result </pre>
--	---	--

```

**                                     int - return code
** Comments                             :
**
*****
*/
int doOrderStatusResults(htmlPhraser *commandBlock, TXN_HANDLE
*txnHandle)
{
    char *html=txnHandle->htmlPage;
    struct ords_wrapper *ords = NULL;
    ords = (ords_wrapper *) txnHandle->comInterface.txnBuffer;
    ZeroMemory(ords,maxDataSize);
    //set warehouse login id from command blk
    ords->in_ords.s_W_ID = txnHandle->w_id;
    //set district login id from command blk
    if( ords->in_ords.s_D_ID = atoi(commandBlock->get_D_ID())) ==
0)
    {
doOrderStatusErrorPage(html,ERR_INVALID_D_ID,commandBlock,txnHandl
e);
        return OK;
    }
    if( ords->in_ords.s_C_ID = atoi(commandBlock->get_C_ID())) ==
0)
    {
        if(*(commandBlock->get_C_NAME()) == NULL)
        {
            //no customer id nor customer last name
            specified.
doOrderStatusErrorPage(html,ERR_MISSING_C_ID_OR_CLAST,commandBl
ock,txnHandle);
                return OK;
            }
            else
            {
                strcpy(ords->in_ords.s_C_LAST,commandBlock->get_C_NAME());
            }
            else
            {
                //make sure that the user only inserted just c_id
                if(*(commandBlock->get_C_NAME()) != NULL)
                {
doOrderStatusErrorPage(html,ERR_C_ID_OR_CLAST_ONLY,commandBloc
k,txnHandle);
                    return OK;
                }
            }
            appendText(&html,"<HTML><HEAD><TITLE>TPC-C Order
Status Results</TITLE></HEAD>\r\n"
            "ACTION=\\"
            "APP_NAME
            "\\"
            METHOD=\\"GET\\">\r\n");
            html+=appendButtons(html);
            html+=appendHiddenFields(html,txnHandle);
            appendText(&html,"<FORM>");
            ords->out_ords.s_transtatus = -1;
            HRESULThres;
            try
            {
                hres =
                txnHandle->comInterface.comHandle->doOrderStatus(&txnHandle->comInterf
ace.size,(UCHAR**) &txnHandle->comInterface.txnBuffer);
            }
            catch(...)
            {
                html+=sprintf(html,"ERROR: ords com call caused
exeception.</PRE></BODY></HTML>");
                return OK;
            }
            if(FAILED(hres))
            {
                html+=sprintf(html,"ERROR: ords com call failed,
rc:%x</PRE></BODY></HTML>",hres);
                ERRORMSG("ERROR : ords com call failed,
rc:"<<DEBUGADDRESS(hres));
                return OK;
            }
            hres = txnHandle->comInterface.comHandle->doSetComplete();
            if(FAILED(hres))
            {
                html+=sprintf(html,"ERROR: ords com doSetComplete
failed, rc:%ld</PRE></BODY></HTML>",hres);
                ERRORMSG("ERROR : ords com doSetComplete failed,
rc:"<<DEBUGADDRESS(hres)<<endl);
                return OK;
            }
            ords = (ords_wrapper *)txnHandle->comInterface.txnBuffer;
            int rc = ords->out_ords.s_transtatus;
            if( rc != OK)
            {
                html+=displayStatus(html,rc);
                appendText(&html,"</PRE></BODY></HTML>");
                ERRORMSG("ERROR order status"<<endl
                "<<"ords->in_ords.s_C_ID:"<<"ords->in_ords.s_C_ID<<endl
                "<<"ords->in_ords.s_C_LAST:"<<"ords->in_ords.s_C_LAST<<endl
                "<<"ords->in_ords.s_D_ID:"<<"ords->in_ords.s_D_ID<<endl
                "<<"ords->in_ords.s_W_ID:"<<"ords->in_ords.s_W_ID<<endl
                "<<"ords->out_ords.deadlocks:"<<"ords->out_ords.deadlocks<<endl
                "<<"ords->out_ords.s_C_BALANCE:"<<"ords->out_ords.s_C_BALANCE<<en
d
                "<<"ords->out_ords.s_C_FIRST:"<<"ords->out_ords.s_C_FIRST<<endl
                "<<"ords->out_ords.s_C_ID:"<<"ords->out_ords.s_C_ID<<endl
                "<<"ords->out_ords.s_C_ID:"<<"ords->out_ords.s_C_ID<<endl
                "<<"ords->out_ords.s_C_MIDDLE:"<<"ords->out_ords.s_C_MIDDLE<<endl
                "<<"ords->out_ords.s_O_CARRIER_ID:"<<"ords->out_ords.s_O_CARRIER_ID
                <<endl
                "<<"ords->out_ords.s_O_ENTRY_D_time:"<<"ords->out_ords.s_O_ENTRY_D_
                time<<endl
                "<<"ords->out_ords.s_O_ID:"<<"ords->out_ords.s_O_ID<<endl
                "<<"ords->out_ords.s_ol_cnt:"<<"ords->out_ords.s_ol_cnt<<endl);
                return OK;
            }
            //start creating result body
            appendText(&html,"</FORM><CENTER><H3>Order-Status</H3></CENTE
R>");

```

```

appendText(&html, "<BR><PRE>\r\nWarehouse: ");
char buffer[50];

appendText(&html, itoa(ords->in_ords.s_W_ID, buffer, 10), 6+1, 1);
appendText(&html, "District: ");
appendText(&html, itoa(ords->in_ords.s_D_ID, buffer, 10));
appendText(&html, "<BR>"
                "Customer: ");

//get customer id
appendText(&html, itoa(ords->in_ords.s_C_ID, buffer, 10), 6+1, 1);
appendText(&html, "Name: ");
//get first, middle, and last from wrapper

appendText(&html, ords->out_ords.s_C_FIRST, FIRST_NAME_LEN+1, 1);

appendText(&html, ords->out_ords.s_C_MIDDLE, INITIALS_LEN+1, 1);

appendText(&html, ords->out_ords.s_C_LAST, LAST_NAME_LEN+5, 1);
//get customer balance from wrapper
appendText(&html, "\r\nCust-Balance: $");
html+=sprintf(html, "%0.2lf", ords->out_ords.s_C_BALANCE/100.0);
//display order number, entry date, and carrier number
appendText(&html, "<BR> <BR>"
                "Order-Number ");

appendText(&html, itoa(ords->out_ords.s_O_ID, buffer, 10), 12, 1);
appendText(&html, "Entry-Date: ");
copyOutDateTime(buffer, ords->out_ords.s_O_ENTRY_D_time);
appendText(&html, buffer, 22, 1);
appendText(&html, "Carrier-Number: ");

appendText(&html, itoa(ords->out_ords.s_O_CARRIER_ID, buffer, 10));
//add item title columns
appendText(&html, "<BR>"
                "Supply-W  "
                "Item-Id  "
                "Qty  "
                "Amount  "
                "Delivery-Date<BR>"
                "
                ");

//display items
for (int
itemCount=0; itemCount<ords->out_ords.s_ol_cnt; itemCount++)
{
//appendSpaces(&html, 2);

//get supp w

appendText(&html, itoa(ords->out_ords.item[itemCount].s_OL_SUPPLY_W_I
D, buffer, 10), 11, 1);

//get item num

appendText(&html, itoa(ords->out_ords.item[itemCount].s_OL_I_ID, buffer, 10),
11, 1);

//get item qty

appendText(&html, itoa(ords->out_ords.item[itemCount].s_OL_QUANTITY, bu
ffer, 10), 6, 1);

//get item dollar amount

html+=sprintf(html, "%0.2lf", ords->out_ords.item[itemCount].s_OL_AMOU
NT/100.0);

//get delivery date

copyOutDate(buffer, ords->out_ords.item[itemCount].s_OL_DELIVERY_D_ti
me);

appendText(&html, buffer);
appendText(&html, "<BR> ");
}

```

```

//append line breaks if item count is less than 15
for (int itemCount=0; itemCount <
(15-ords->out_ords.s_ol_cnt); itemCount++)
appendText(&html, "<BR> ");

html+=displayStatus(html, rc);

appendText(&html, "</PRE></BODY></HTML>");
return OK;
}
/*
*****
** Name           : doOrderStatusErrorPage
** Description    :
**               : HTML orderStatus error page
** Parameters    :
**               : char *          html page
result
**               : char *          error
message
**               : htmlPhraser*    command
block
**               : TXN_HANDLE*     txn handle
** Returns      :
**               : int - return code
** Comments     :
**
*****
*/
int doOrderStatusErrorPage(char *htmlPage, char *message, htmlPhraser
*commandBlock, TXN_HANDLE *txnHandle)
{
char *html=htmlPage;
appendText(&html, "<HTML><HEAD><TITLE>TPC-C Order
Status</TITLE></HEAD>\r\n"
                "<BODY><FORM
ACTION=\\"
                APP_NAME
                "\\"
METHOD=\\"GET\\"">
                "\r\n"
                "<CENTER><H3>Please Fill In Order Status Form.</H3></CENTER>
<BR>\r\n"
                "Submit Transaction
                CMD_TXN_ID
                "\\" VALUE=\\"
                CMD_ORDS
                "\\">"
                "<BR> ");

html+=appendHiddenFields(html, txnHandle);

appendText(&html, "<PRE>\r\n"
                "Warehouse: ");

char buffer[15];
appendText(&html, itoa(txnHandle->w_id, buffer, 10));
appendText(&html, "District: <INPUT NAME=\\"
                CMD_D_ID
                "\\" SIZE=1>\r\n<BR>"
                "Customer: "
                "<INPUT NAME=\\"
                CMD_C_ID
                "\\" SIZE=5>"
                " "
                "Name: "
                "<INPUT NAME=\\"
                CMD_C_NAME
                "\\" SIZE=20><BR>"

```

```

                                "Cust-Balance: <BR>"
                                "Order-Number:
Entry-Date:          Carrier-Number<BR>"
                                "Supply-W
Item-Num  Qty    Amount    Delivery <BR>");
                                appendText(&html,message);
                                appendText(&html,"<PRE></BODY></HTML>");
                                return OK;
}
/*
*****
** Name          : doDeliveryForm
** Description   :
**              : HTML payment page entry point
** Parameters    :
**              : htmlPhraser*    command
block
**              : TXN_HANDLE*    txn handle
struct
** Returns      :
**              : int - return code

** Comments     :
**
*****
*/
int doDeliveryForm(htmlPhraser *commandBlock, TXN_HANDLE
*txnHandle)
{
    char *html=txnHandle->htmlPage;
    appendText(&html,"<HTML><HEAD><TITLE>TPC-C
Delivery</TITLE></HEAD>\r\n"
                                "
                                "<BODY><FORM
ACTION=\\\"
                                APP_NAME
                                \"\"
                                METHOD=\\\"GET\\\">\r\n"
                                "<CENTER><H3>Delivery.</H3></CENTER>\r\n"
                                "Submit Transaction
                                CMD_TXN_ID
                                \"\" VALUE=\\\"
                                CMD_DLVY
                                \"\">");
    html+=appendHiddenFields(html,txnHandle);
    appendText(&html,"<BR> <PRE>"
                                "Warehouse: ");
    char buffer[10];
    appendText(&html,ittoa(txnHandle->w_id,buffer,10));
    appendText(&html," <BR> <BR>"
                                "Carrier Number: "
                                "<INPUT NAME=\\\"
                                CMD_CARRIER_NUM
                                \"\" SIZE=1>"
                                "</FORM></PRE>");
    appendText(&html,"</BODY></HTML>");
    return OK;
}
/*
*****
** Name          : doDeliveryResults
** Description   :
**              : HTML payment page entry point
                                ** Parameters    :
                                **              : htmlPhraser*    command
                                block
                                **              : TXN_HANDLE*    txn handle
                                ** Returns      :
                                **              : int - return code
                                ** Comments     :
                                **
                                *****
                                */
                                int doDeliveryResults(htmlPhraser *commandBlock, TXN_HANDLE
                                *txnHandle)
                                {
                                    char *html = txnHandle->htmlPage;
                                    //declare delivery structure
                                    struct dlvy_wrapper dlvy;
                                    //set warehouse login id from command blk
                                    dlvy.in_dlvvy.s_W_ID = txnHandle->w_id;
                                    //set the carrier id from command blk
                                    if( dlvy.in_dlvvy.s_O_CARRIER_ID =
                                    atoi(commandBlock->get_CARRIER_NUM()) == 0)
                                    {
                                        doDeliveryErrorPage(html,ERR_INVALID_CARRIER,commandBlock,txnHan
                                        dle);
                                        return OK;
                                    }
                                    //print title, add hidden fields , txn buttons
                                    appendText(&html,"<HTML><HEAD><TITLE>TPC-C Delivery
                                    Results</TITLE></HEAD>\r\n<BODY><FORM ACTION=\\\"
                                    APP_NAME
                                    \"\"
                                    METHOD=\\\"GET\\\">\r\n");
                                    html+=appendButtons(html);
                                    html+=appendHiddenFields(html,txnHandle);
                                    appendText(&html,
                                    "<FORM><CENTER><H3>Delivery</H3></CENTER>");
                                    int rc =
                                    queueDlvyTxn(dlvy.in_dlvvy.s_W_ID,dlvy.in_dlvvy.s_O_CARRIER_ID);
                                    if( rc != OK)
                                    {
                                        html+=displayStatus(html,rc);
                                        appendText(&html,"<PRE></BODY></HTML>\r\n");
                                        ERRORMSG("ERROR: Unable to queue dlvy txn,
                                        rc:<<rc<<endl);
                                        return OK;
                                    }
                                    //start creating result body
                                    appendText(&html,"Warehouse: ");
                                    //get w_id from wrapper
                                    char buffer[15];
                                    appendText(&html,ittoa(dlvy.in_dlvvy.s_W_ID,buffer,10));
                                    appendText(&html,"<BR> <BR>Carrier Number: ");
                                    //get carrier_id from wrapper
                                    appendText(&html,ittoa(dlvy.in_dlvvy.s_O_CARRIER_ID,buffer,10));
                                    appendText(&html,"<BR> <BR>Execution Status: Delivery has
                                    been queued <PRE></BODY></HTML>");
                                    return OK;
                                }
                                /*
                                *****
                                ** Name          : doDeliveryErrorPage
                                ** Description   :

```

```

**                                     HTML payment error page entry
point
** Parameters       :
**                                     char *           html result
page
**                                     char *           error
message
**                                     htmlPhraser     command
block
**                                     TXN_HANDLE*    txn handle
**
** Returns          :
**                                     int - return code
** Comments         :
**
*****
*/
int doDeliveryErrorPage(char *htmlPage,char *message,htmlPhraser
*commandBlock,TXN_HANDLE *txnHandle)
{
    char *html=htmlPage;
    appendText(&html,"<HTML><HEAD><TITLE>TPC-C
Delivery</TITLE></HEAD>\r\n"
    "ACTION=\\"
    APP_NAME
    "\\"
    METHOD=\\"GET\\"\r\n"
    "<CENTER><H3>Delivery.</H3></CENTER>\r\n"
    "<INPUT TYPE=\\"submit\\" NAME=\\"
    CMD_TXN_ID
    "\\" VALUE=\\"
    CMD_STOK
    "\\">");
    html+=appendHiddenFields(html,txnHandle);
    appendText(&html,"<BR> <PRE>"
    "Warehouse: ");
    char buffer[15];
    appendText(&html,ittoa(txnHandle->w_id,buffer,10));
    appendText(&html,"District: ");
    appendText(&html,ittoa(txnHandle->d_id,buffer,10));
    appendText(&html," <BR> <BR>"
    "Stock Level
    Threshold: "
    "<INPUT NAME=\\"
    CMD_STK_THRESHOLD
    "\\" SIZE=1> <BR>"
    "<BR>"
    "Low Stock: <BR>"
    "</PRE>");
    appendText(&html,"</FORM></BODY></HTML>");
    return OK;
}
/*
*****
** Name           : doStockForm
** Description    :
**               HTML stock page entry point
** Parameters     :
**               htmlPhraser     command
block
**               TXN_HANDLE*    txn handle
struct
** Returns       :
**               int - return code
** Comments      :
**
*****
*/
int doStockForm(htmlPhraser *commandBlock,TXN_HANDLE *txnHandle)
{
    char *html=txnHandle->htmlPage;
    struct stok_wrapper *stok;
    stok = (stok_wrapper*)txnHandle->comInterface.txnBuffer;
    ZeroMemory(stok,maxDataSize);
    //set warehouse login id from command blk
    stok->in_stok.s_W_ID = txnHandle->w_id;
    //set district login id from command blk
    stok->in_stok.s_D_ID = txnHandle->d_id;
    //set stock level threshold id from command blk
}
/*
*****
** Name           : doStockResults
** Description    :
**               HTML stock page entry point
** Parameters     :
**               htmlPhraser*    command
block
**               TXN_HANDLE*    txn handle
struct
** Returns       :
**               int - return code
** Comments      :
**
*****
*/
int doStockResults(htmlPhraser *commandBlock,TXN_HANDLE *txnHandle)
{
    char *html = txnHandle->htmlPage;
    struct stok_wrapper *stok;
    stok = (stok_wrapper*)txnHandle->comInterface.txnBuffer;
    ZeroMemory(stok,maxDataSize);
    //set warehouse login id from command blk
    stok->in_stok.s_W_ID = txnHandle->w_id;
    //set district login id from command blk
    stok->in_stok.s_D_ID = txnHandle->d_id;
    //set stock level threshold id from command blk
}

```



```

        if( (stok->in_stok.s_threshold =
atoi(commandBlock->get_STK_THRESHOLD()) == 0)
        {
doStockErrorPage(html,ERR_INVALID_THRESHOLD,commandBlock,txnHandle);
        return OK;
        }
//assume failure, set s_transtatus to err
stok->out_stok.s_transtatus = INVALID_STATUS;
//print title, add hidden fields , txn buttons
appendText(&html,"<HTML><HEAD><TITLE>TPC-C Stock
Level Results</TITLE></HEAD>\r\n"
        " <BODY><FORM
ACTION=\\"
        APP_NAME
        "\\"
METHOD=\\"GET\>\r\n");
html+=appendButtons(html);
html+=appendHiddenFields(html,txnHandle);
appendText(&html,"</FORM>");
stok->out_stok.s_transtatus = -1;

        DEBUGMSG("Calling com entry api for stock call,
w_id:"<<stok->in_stok.s_W_ID<<" d_id:"<<stok->in_stok.s_D_ID<<
        " threshold:"<<stok->in_stok.s_threshold<<endl);
        HRESULT hres;
        try
        {
                hres =
txnHandle->comInterface.comHandle->doStockLevel(&txnHandle->comInterface.size,(UCHAR*)&txnHandle->comInterface.txnBuffer);
        }
        catch(...)
        {
                html+=sprintf(html,"ERROR: Com Stock call caused
exeception to occur.</PRE></BODY></HTML>");
                ERRORMSG("ERROR : Com Stock call caused
exeception to occur."<<endl);
                return OK;
        }
        if(FAILED(hres))
        {
                html+=sprintf(html,"ERROR: stok com call failed,
rc:%ld</PRE></BODY></HTML>",hres);
                ERRORMSG("ERROR : stok com call failed,
rc:"<<DEBUGADDRESS(hres)<<endl);
                return OK;
        }

        hres = txnHandle->comInterface.comHandle->doSetComplete();
        if(FAILED(hres))
        {
                html+=sprintf(html,"ERROR: stok com doSetComplete
failed, rc:%ld</PRE></BODY></HTML>",hres);
                ERRORMSG("ERROR : stok com doSetComplete failed,
rc:"<<DEBUGADDRESS(hres)<<endl);
                return OK;
        }
        stok = (stok_wrapper *)txnHandle->comInterface.txnBuffer;
        int rc = stok->out_stok.s_transtatus;
        if(rc != OK)
        {
                html+=displayStatus(html,rc);
                appendText(&html,"</PRE></BODY></HTML>");
                ERRORMSG("ERROR stok txn failed"<<endl
<<"stok->in_stok.s_D_ID:"<<stok->in_stok.s_D_ID<<endl

```

```

<<"stok->in_stok.s_threshold:"<<stok->in_stok.s_threshold<<endl
<<"stok->in_stok.s_W_ID:"<<stok->in_stok.s_W_ID<<endl
<<"stok->out_stok.deadlocks:"<<stok->out_stok.deadlocks<<endl
<<"stok->out_stok.s_low_stock:"<<stok->out_stok.s_low_stock<<endl
<<"stok->out_stok.s_transtatus:"<<stok->out_stok.s_transtatus<<endl);
        return OK;
        }
//start creating result body
appendText(&html,"<FORM><CENTER><H3>Stock-Level</H3></CENTER>");
        appendText(&html,"<BR><PRE>\r\n"
                "Warehouse: ");
//get w_id from wrapper
char buffer[10];
appendText(&html,itoa(stok->in_stok.s_W_ID,buffer,10),6+1,1);
appendText(&html,"District: ");
appendText(&html,itoa(stok->in_stok.s_D_ID,buffer,10));
        appendText(&html,"<BR><BR>"
                "Stock Level
Threshold: ");
        appendText(&html,itoa(stok->in_stok.s_threshold,buffer,10));
        appendText(&html,"<BR><BR>"
                "Low Stock: ");
        appendText(&html,itoa(stok->out_stok.s_low_stock,buffer,10));
        appendText(&html,"<BR><BR>");
html+=displayStatus(html,rc);
        appendText(&html,"</PRE></BODY></HTML>");
        return OK;
}
/*
*****
** Name          : doStockErrorPage
** Description   :
**               : HTML stock page entry point
** Parameters    :
**               : char *          html result
page
**               : char *          query string
**               : htmlPhraser     command
block
**               : TXN_HANDLE *   handle for
this transaction
** Returns      :
**               : int - return code
** Comments     :
**
*****
*/
int doStockErrorPage(char *htmlPage,char *message,htmlPhraser
*commandBlock,TXN_HANDLE *txnHandle)
{
        char *html=htmlPage;
        appendText(&html,"<HTML><HEAD><TITLE>TPC-C Stock
Level</TITLE></HEAD>\r\n"
                " <BODY><FORM
ACTION=\\"
                APP_NAME
                "\\"
METHOD=\\"GET\>\r\n"

```

```

"<CENTER><H3>Please Fill In Stock Form.</H3></CENTER> <BR>\r\n"
"Submit Transaction
<INPUT TYPE=\"submit\" NAME=\"\"
        CMD_TXN_ID
        \"\" VALUE=\"\"
        CMD_STOK
        \"\">");
html+=appendHiddenFields(html,txnHandle);
appendText(&html,"<PRE>"
        "Warehouse: ");
char buffer[15];
appendText(&html,ittoa(txnHandle->w_id,buffer,10));
appendSpaces(&html,2);
appendText(&html,"District: ");
appendText(&html,commandBlock->get_D_ID());
appendText(&html,"<BR> <BR>"
        "Stock Level
Threshold: "
        "<INPUT NAME=\"\"
        \"\" SIZE=1> <BR>
        \"Low Stock: <BR>");
appendText(&html,message);
appendText(&html,"</PRE></FORM></BODY></HTML>");
return OK;
}
*/
*****
** Name          : doExit
** Description    :
**               HTML exit page entry point
** Parameters    :
**               htmlPhraser*   command
block
**               TXN_HANDLE*   txn handle
struct
** Returns      :
**               int - return code
** Comments     :
*****
*/
int doExit(htmlPhraser *commandBlock, TXN_HANDLE *txnHandle)
{
    return (doLoginForm(commandBlock,txnHandle));
}
/*
*****
** Name          : displayStatus
** Description    :
**               appends status string to the html
page
** Parameters    :
**               char*          html page
**               int            rc
** Returns      :
**               amount of characters the function
appened
**               to the html page
** Comments     :
*****
*/

```

```

*/
int displayStatus(char *htmlPage,int rc)
{
    char *html = htmlPage;
    appendText(&html,"");
    switch (rc)
    {
        case OK:
            appendText(&html,"Execution Status: Transaction
Committed",50,1);
            break;
        case INVALID_ITEM:
            appendText(&html,"Execution Status: Item number is not
valid",50,1);
            break;
        case INVALID_STATUS:
            appendText(&html,"Execution Status: ERROR: Rollback
INVALID_STATUS",50,1);
            break;
        case INVALID_COM_STATUS:
            appendText(&html,"Execution Status: ERROR: Rollback
COM FAILURE",50,1);
            break;
        case ERR_DLVY_QUEUE_FULL:
            appendText(&html,"Execution Status: ERROR: Rollback
DLVY QUEUE FULL",50,1);
            break;
        default:
            appendText(&html,"Execution Status: ERROR:
Rollback",50,1);
    }
    appendText(&html," ");
    return (int)(html - htmlPage);
}
/*
*****
** Name          : appendButtons
** Description    :
**               append hidden field to recognize
user after login
** Parameters    :
**               *htmlPage
html result page
**               *TXN_HANDLE
txn handle
** Returns      :
**               int
amount of characters the function appened
**               to the html page
** Comments     :
*****
*/
int appendHiddenFields(char *htmlPage, TXN_HANDLE *txnHandle)
{
    char *html = htmlPage;
    char buffer[15];
    appendText(&html,"<INPUT TYPE=\"hidden\" NAME=\"\"
        CMD_TERM_ID
        \"\" VALUE=\"\"");
    appendText(&html,ittoa(txnHandle->term_id,buffer,10));
    appendText(&html,"\">\r\n");
    return (int)(html-htmlPage);
}

```

```

/*
*****
** Name          : appendButtons
** Description    :
**               : appends buttons transaction
buttons to result page
** Parameters    :
**               : *htmlPage
**
** Returns       :
**               : amount of characters the function
appened
**               : to the html page
** Comments      :
**
*****
*/
int appendButtons(char *htmlPage)
{
    char *html = htmlPage;
    appendText(&html,"<INPUT TYPE='submit' NAME='\"
        CMD_TXN_ID
        \" VALUE='\"
        CMD_NORD
        \">\r\n\"
        \"<INPUT
TYPE='submit' NAME='\"
        CMD_TXN_ID
        \" VALUE='\"
        CMD_PYMT
        \">\r\n\"
        \"<INPUT
TYPE='submit' NAME='\"
        CMD_TXN_ID
        \" VALUE='\"
        CMD_ORDS
        \">\r\n\"
        \"<INPUT
TYPE='submit' NAME='\"
        CMD_TXN_ID
        \" VALUE='\"
        CMD_DLVE
        \">\r\n\"
        \"<INPUT
TYPE='submit' NAME='\"
        CMD_TXN_ID
        \" VALUE='\"
        CMD_STOK
        \">\r\n\"
        \"<INPUT
TYPE='submit' NAME='\"
        CMD_TXN_ID
        \" VALUE='\"
        CMD_EXIT
        \">\r\n <BR>\";
    return (int)(html - htmlPage);
}
/*
*****
** Name          : appendItems
** Description    :
**               : appends items to new order and
order status page
** Parameters    :
**               : *htmlPage
**               : html result page
**               : short
**               : items to append
**
** Returns       :
**               : short
**               : item CMD id start
** Returns      :
**               : amount of characters the function
appened
** Comments     :
**               : to the html page
*****
*/
int appendItems(char *htmlPage,short itemCount,short cmdIDStart)
{
    char *html = htmlPage;
    char numBuffer[MAX_INT_BUFFER];
    for(int item=0;item < itemCount;item++)
    {
        appendText(&html,\"<BR> <INPUT NAME='\"");
        appendText(&html,ittoa(cmdIDStart++,numBuffer,10));
        appendText(&html,\" SIZE=6> <INPUT NAME='\"");
        appendText(&html,ittoa(cmdIDStart++,numBuffer,10));
        appendText(&html,\" SIZE=6>
<INPUT NAME='\"");
        appendText(&html,ittoa(cmdIDStart++,numBuffer,10));
        appendText(&html,\" SIZE=2>\r\n\"");
    }
    return (int)(html - htmlPage);
}
/*
*****
** Name          : dlvyThreadEntry
** Description    :
**               : dlvy thread worker entry point
** Parameters    :
** Returns       :
** Comments      :
**               : All dlvy threads created by
initDlvy enter at
**               : this point. They must first make a
connection
**               : to the database, then go to sleep.
**               : Main isapi threads control dlvy
worker semaphore
**               : and signal when a dlvy txn is
queued.
**
*****
*/
void dlvyThreadEntry(void *)
{
    int rc = 0;
    DEBUGMSG("dlvyThread " << GetCurrentThreadId() << " entered
dlvyThreadEntry, calling db_connect to db:" << dbName << endl);
    void *connectHandle;
    //connect to database.
    DEBUGMSG("ptr created. calling db_connect to db:" << dbName
<< endl);
    rc = db_connect(dbName,&connectHandle);
    if(rc != OK)
    {
        ERRORMSG("dlvyThread " << GetCurrentThreadId()
<<" unable to connect to database, rc:" << rc << endl);
    }
}

```

```

        DEBUGMSG("dlvyThread " << GetCurrentThreadId()
<<" unable to connect to database, rc:" << rc << endl);
        return;
    }

    DEBUGMSG("dlvyThread " << GetCurrentThreadId() << " connect
to db:" << dbName << " successful" << endl);

    FILE *dlvyLog = NULL;
    char logFileName[MAX_STRING_LEN] = {NULL};

    EnterCriticalSection(&isapiLock);
    //open dlvy log file for this thread
    sprintf(logFileName,"%s\\del_%d.txt",dlvyLogPath,dlvyThreadID);
    dlvyLog = fopen(logFileName,"w");
    if(!dlvyLog)
    {
        ERRORMSG("dlvyThread " << GetCurrentThreadId()
<< " unable to open dlvy log "
                << dlvyLogPath << "\\del_" <<
dlvyThreadID << endl);
        DEBUGMSG("dlvyThread " << GetCurrentThreadId()
<< " unable to open dlvy log "
                << dlvyLogPath << "\\del_" <<
dlvyThreadID << endl);
        return;
    }
    //increment the global dlvy thread id
    dlvyThreadID++;
    LeaveCriticalSection(&isapiLock);

    DEBUGMSG("dlvyThread " << GetCurrentThreadId() <<" dlvy log
file name: " << logFileName << " open." << endl);
    HANDLE workerHandles[2];
    //handle array to store event to wait on
    struct DLVYQUEUEDATA dlvyQueueData;
    //dlvy queue struct to store queued txn
    struct dlvy_wrapper dlvyTxn;
    //dlvy wrapper of db2 structs
    struct _timeb
endQueueTime; //time stamp to queue removal time
    struct _timeb
endProcessTime; //time stamp for end process time
    char orderIDs[MAX_STRING_LEN] = {NULL};
    //string to store oids for each district
    int bytesWritten = 0;
    int dlvyCount = 0;
    DEBUGMSG("dlvyThread entering work loop" << endl);
    //successful, while true
    while(true)
    {
        try
        {
            DEBUGMSG("dlvyThread initializing wait
handles" << endl);

            //wait for both program exit AND if there is
work to do
            workerHandles[0] = dlvyThreadDone;
            workerHandles[1] = dlvyThreadSemaphore;
            DEBUGMSG("dlvyThread going to sleep
waiting for wrk" << endl);

            rc =
WaitForMultipleObjects(2,&workerHandles[0],FALSE,INFINITE);

            DEBUGMSG("dlvyThread awake, checking
wake condition" << endl);

            if(rc == WAIT_OBJECT_0)
                break;
            else if(rc == (WAIT_OBJECT_0+1) )
            {
                DEBUGMSG("dlvyThread awake,
wake condition of dlvyThreadSemaphore" << endl);
            }
            DEBUGMSG("dlvyThread trying to enter
critical section" << endl);

            EnterCriticalSection(&dlvyQueueLock);

            DEBUGMSG("dlvyThread entered critical
section" << endl);

            //remove queued dlvy txn
            dlvyQueueData.enqueueTime.time
            = dlvyQueue[dlvyBufferThreadIndex].enqueueTime.time;
            dlvyQueueData.enqueueTime.millitm
            = dlvyQueue[dlvyBufferThreadIndex].enqueueTime.millitm;
            dlvyQueueData.in_s_0_CARRIER_ID
            = dlvyQueue[dlvyBufferThreadIndex].in_s_0_CARRIER_ID;
            dlvyQueueData.warehouse
            = dlvyQueue[dlvyBufferThreadIndex].warehouse;

            DEBUGMSG("dlvyThread removed dlvy:"
<< dlvyCount << ",w_id:" << dlvyQueueData.warehouse
<< " carrier_id:" << dlvyQueueData.in_s_0_CARRIER_ID << endl);
            DEBUGMSG("dlvyThread removed dlvy in
queue index: " <<dlvyBufferThreadIndex<< " w_id: " <<
dlvyQueueData.warehouse
<< " carrier_id: " << dlvyQueueData.in_s_0_CARRIER_ID << endl);
            //increment the number of free slots
            dlvyBufferFreeSlots++;
            //increment the thread index to next slot in
dlvy queue
            dlvyBufferThreadIndex++;

            DEBUGMSG("dlvyThread incremented
amount of free slots:" << dlvyBufferFreeSlots << " and thread index:" <<
dlvyBufferThreadIndex << endl);
            //check if we reached the end of dlvy queue, if
so, reset back index back to 0
            if(dlvyBufferThreadIndex == dlvyQueueLen)
            {
                DEBUGMSG("dlvyThread reset
dlvyBufferThreadIndex to 0, current dlvyBufferThreadIndex:" <<
dlvyBufferThreadIndex
                << " free
slots:"<<dlvyBufferFreeSlots<<endl);
                dlvyBufferThreadIndex=0;
            }
            DEBUGMSG("dlvyThread releasing critical
section" << endl);

            LeaveCriticalSection(&dlvyQueueLock);
            //take enqueue time
            _ftime(&endQueueTime);

            DEBUGMSG("dlvyThread executing txn
w_id:" << dlvyQueueData.warehouse
                << " carrier_id:" <<
dlvyQueueData.in_s_0_CARRIER_ID << endl);

            //prepare to call database

```

```

        dlvyTxn.in_dlvy.s_O_CARRIER_ID =
dlvyQueueData.in_s_0_CARRIER_ID;
        dlvyTxn.in_dlvy.s_W_ID
=      dlvyQueueData.warehouse;
        dlvyTxn.out_dlvy.s_transtatus = -1;

        //increment dlvy count
        dlvyCount++;
        DEBUGMSG("dlvyThread %d calling dlvy
txn" << rc << endl);

        //call dlvy txn
        rc = dlvyCall(&dlvyTxn,connectHandle);
        _ftime(&endProcessTime);
        rc = dlvyTxn.out_dlvy.s_transtatus;

        DEBUGMSG("dlvy txn response time:"<<
                (((endProcessTime.time -
endQueueTime.time)*1000)+
(endProcessTime.millitm - endQueueTime.millitm))/1000.0)<<
                "
w_id:"<<dlvyTxn.in_dlvy.s_W_ID<<" carrier:"
<<dlvyTxn.in_dlvy.s_O_CARRIER_ID<<
                "
deadLocks:"<<dlvyTxn.out_dlvy.deadlocks<<" rc: "<< rc <<endl);
        DEBUGMSG("dlvyThread dlvy s_transtatus:"
<< rc << endl);

        if(rc == OK)
        {
                bytesWritten=0;
                char *buffer = orderIDs;
                for(int
districtIndex=0;districtIndex <
DISTRICTS_PER_WAREHOUSE;districtIndex++)
                {
                        if(dlvyTxn.out_dlvy.s_O_ID[districtIndex] == 0)
                                bytesWritten
= sprintf(buffer,"%nD_ID %d had no new orders",districtIndex);
                                else
                                        bytesWritten
= sprintf(buffer,"%d ",dlvyTxn.out_dlvy.s_O_ID[districtIndex]);
                                buffer+=bytesWritten;
                }
                else
                        sprintf(orderIDs,"%nDelivery
transaction failed");

        fprintf(dlvyLog,DELIVERY_LOG_SUCCESS_STR,
dlvyCount,
dlvyQueueData.enqueueTime.time,
dlvyQueueData.enqueueTime.millitm,
endQueueTime.time,
endQueueTime.millitm,
dlvyQueueData.warehouse,
dlvyQueueData.in_s_0_CARRIER_ID,
orderIDs,
endProcessTime.time,
endProcessTime.millitm);

                fflush(dlvyLog);
        }
        catch(...)
        {
                ERRORMSG("ERROR: Unhandled
exeception in dlvy thread. Thread exiting"<<endl);
                fprintf(dlvyLog,"ERROR: Unhandled
exeception in dlvy thread %ld. Thread exiting.\n",GetCurrentThreadId());
                fflush(dlvyLog);
                LeaveCriticalSection(&dlvyQueueLock);
        }
        } //end while true
}
/*
*****
** Name : queueDlvyTxn
** Description :
**          function queues dlvy txn in dlvy
queue
** Parameters :
**          int warehouse
**          short carrier
** Returns :
**          int error code
** Comments :
**          Function will queue
dlvy txn if 2 points are true
**          1) We have room in our
dlvy buffer
**          2) We writing over the
end of the queue
**
*****
*/
int queueDlvyTxn(int warehouse, short carrier_id)
{
        DEBUGMSG("Taking lock to queue dlvy txn.");
        EnterCriticalSection(&dlvyQueueLock);
        DEBUGMSG("Lock aquired to queue dlvy txn");
        if(dlvyBufferFreeSlots)
        {
                DEBUGMSG("Checking if we are inserting at tail of
dlvy queue."<<endl);
                if( dlvyBufferSlotIndex == (dlvyBufferThreadIndex-1))
                {
                        ERRORMSG("Error dlvy queue inserting
over unserviced queued dlvy txn."<<endl);
                        DEBUGMSG("Error dlvy queue inserting
over unserviced queued dlvy txn."<<endl);
                        LeaveCriticalSection(&dlvyQueueLock);
                        return
ERR_DLVY_QUEUE_EATING_TAIL;
                }
                DEBUGMSG("free slots dlvy
queue:"<<dlvyBufferFreeSlots<<" inserting txn in slot: "
<<dlvyBufferSlotIndex<<
                "w_id: "<<warehouse<<" carrier:
"<<carrier_id<<endl);
                dlvyQueue[dlvyBufferSlotIndex].warehouse =
warehouse;
                dlvyQueue[dlvyBufferSlotIndex].in_s_0_CARRIER_ID
= carrier_id;

```

```

    _ftime(&dlvyQueue[dlvyBufferSlotIndex].enqueueTime);
    //decrement the number of free slots in the buffer
    dlvyBufferFreeSlots--;

    //increment the index to the next dlvy queue slot.
    dlvyBufferSlotIndex++;

    DEBUGMSG("dlvy txn queued, slots available in
queue:"<<dlvyBufferFreeSlots<<" queue slot index:"<<dlvyBufferSlotIndex
carrier:"<<carrier_id<<endl);
    DEBUGMSG("dlvy txn queued, slots available in queue:
"<<dlvyBufferFreeSlots<<" queue slot index: "<<dlvyBufferSlotIndex
<<" w_id:"<<warehouse<<"
carrier: "<<carrier_id<<endl);
    if(dlvyBufferSlotIndex == dlvyQueueLen)
    {
        DEBUGMSG("queue slot index hit end of
queue, reset to 0, current index:"<<dlvyBufferSlotIndex<<" free
slots:"<<dlvyBufferFreeSlots<<endl);
        dlvyBufferSlotIndex=0;
    }
    else
    {
        //no slots available in dlvy buffer, release critical section
and return an nord->in_nord.in_item
        LeaveCriticalSection(&dlvyQueueLock);
        ERRORMSG("dlvy queue buffer full, increase the dlvy
queue length."<<endl);
        return ERR_DLVE_QUEUE_FULL;
    }
    LeaveCriticalSection(&dlvyQueueLock);
    //release semaphore to wake thread that there is work
    ReleaseSemaphore(dlvyThreadSemaphore,1,NULL);
    return OK;
}

/*
*****
** Name          : doHtml
** Description   :
**               HTML processing page entry
point
** Parameters   :
**               txn handle
** Returns      :
**               int - return code
** Comments     :
**
*****
*/
void doHtml(TXN_HANDLE *txnHandle)
{
    DEBUGMSG("Entered doHtml(), parsing query string:"<<
txnHandle->urlString <<" into command block"<< endl);
    htmlPhraser      commandBlock(txnHandle->urlString);
    DEBUGMSG("Query string parsed.  command:"<<
commandBlock.getCommandId() <<" user's terminal id:" <<
commandBlock.get_TERM_ID() <<endl);

    int terminalID = atoi(commandBlock.get_TERM_ID());
    int commandID = commandBlock.getCommandId();
    DEBUGMSG("User sent in a terminal id:"<<terminalID<<" , checking
to see if user has logged in before"<<endl);
    if(terminalID > 0)

```

```

    {
        DEBUGMSG("Terminal id > 0, user has logged in
already, terminalID:"<<terminalID<<" retrieving warehouse district
pair"<<endl);
        if(getTerminal(terminalID,txnHandle) != OK)
            return;
        DEBUGMSG("User had valid terminal id, user's login
warehouse:"<<txnHandle->w_id<<" district:"<<txnHandle->d_id<<endl);
    }
    else
    {
        DEBUGMSG("User did not submit a terminal id or valid
terminal id, ensure that the user is trying to log in."<<endl);
        if( (commandID != TXN_LOGIN) && (commandID !=
TXN_LOGIN_RESULTS) )
        {
            DEBUGMSG("ERROR : User has not logged
in."<<endl);
            ERRORMSG("ERROR : User has not logged
in."<<endl);
            sprintf(txnHandle->htmlPage,"ERROR: User
has not logged in or did not submit a valid terminal.");
            return;
        }
        DEBUGMSG("User is in process of logging in,
commandID:"<<commandID<<endl);
    }
    DEBUGMSG("Calling html page
function:"<<commandBlock.getCommandId()<<endl);
    int rc =
htmlPageFunctions[commandBlock.getCommandId()](&commandBlock,txnHa
ndle);
    DEBUGMSG("Return from html page
function:"<<commandBlock.getCommandId()<<endl);
    return;
}
/*
*****
** Name          : getTerminal
** Description   :
**               retrieves terminal information
based on terminal id
** Parameters   :
**               int
terminal id
** Returns      :
**               TERM_HANDLE* txn handle
** Comments     :
**               int - return code
**
*****
*/
int getTerminal(int terminal,TXN_HANDLE *txnHandle)
{
    //check to see if terminal id is out of range
    if(terminal >= numUsers)
    {
        //terminal id not valid.
        sprintf(txnHandle->htmlPage,"ERROR: Client does not
support more than %d users, terminal id:%d",numUsers,terminal);
        ERRORMSG("ERROR : Client does not support more
than "<<numUsers<<" users, terminal id:"<<terminal<<endl);
        return ERR;
    }
    //check if terminal id is points to a not in use terminal
    if(!(termArray+terminal)->terminalInUse)
    {

```

```

        sprintf(txnHandle->htmlPage,"ERROR: Terminal id
given points to a not in use terminal.");
        ERRORMSG("ERROR : Terminal id given points to a
not in use terminal."<<endl);
        return ERR;
    }
    DEBUGMSG("Storing terminal warehouse, district , and initial term
id for user:"<<terminal<<endl);
    //assign terminal values to txn_handle
    txnHandle->d_id = termArray[terminal].d_id;
    txnHandle->w_id = termArray[terminal].w_id;
    txnHandle->term_id = termArray[terminal].terminalID;
    DEBUGMSG("Users terminal:"<<terminal<< ", stored
warehouse:"<<txnHandle->w_id<<
        " district:"<<txnHandle->d_id<<" terminalID
stored:"<<txnHandle->term_id<<endl);
    return OK;
}
/*
*****
** Name          : assignTerminal
** Description    :
**               : assigns terminal index to user
** Parameters    :
**               : TERM_HANDLE* txn handle
** Returns       :
**               : int - return code
** Comments      :
**               :
*****
*/
int assignTerminal(TXN_HANDLE *txnHandle)
{
    EnterCriticalSection(&termLock);

    //check if terminal array is full.
    if(termNextFree == numUsers)
    {
        LeaveCriticalSection(&termLock);
        return ERR;
    }

    DEBUGMSG("Storing user warehouse:"<<txnHandle->w_id<<"
district:"<< txnHandle->d_id<<
        " in terminal slot:"<<termNextFree<<endl);
    //store users w_id and d_id
    termArray[termNextFree].d_id = txnHandle->d_id;
    termArray[termNextFree].w_id = txnHandle->w_id;

    //set terminal slot to be in use
    termArray[termNextFree].terminalInUse = true;
    termArray[termNextFree].terminalID = termNextFree;
    //in txn handle, set the terminal id
    txnHandle->term_id = termNextFree;

    //increment to next free terminal.
    termNextFree++;
    DEBUGMSG("User warehouse:"<<txnHandle->w_id<<"
district:"<< txnHandle->d_id <<
        " stored in terminal slot:"<<txnHandle->term_id<<" next
terminal free:"<<termNextFree<<endl);
    LeaveCriticalSection(&termLock);
    return OK;
}

```

A.2 Client Transaction Code

Makefile.config

```

#####
#####
## Licensed Materials - Property of IBM
##
## Governed under the terms of the International
## License Agreement for Non-Warranted Sample Code.
##
## (C) COPYRIGHT International Business Machines Corp. 1996 - 2005
## All Rights Reserved.
##
## US Government Users Restricted Rights - Use, duplication or
## disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
#####
#####
#
# Makefile.config - NT/Win2000 Makefile Configuration
#
# Make Configuration (MSVC)
MAKE=nmake.exe

# Compiler Configuration (MSVC).
# CFLAGS_DEBUG may be set to "-Zi -Od", "-DDEBUGIT" "-Zi -Od
-DDEBUGIT" or left blank
CC=cl.exe
CFLAGS_OS=-DSQLWINT -MT -GS- -DWIN64 -J -Zp8
-DREG_KIT_METHOD
CFLAGS_OUT=/Fo
CFLAGS_DEBUG=

# Linker Configuration (MSVC)
LD_EXEC=link.exe
LD_STORP=link.exe
LD_FLAGS_EXEC=
LD_FLAGS_SHLIB=/DLL
LD_FLAGS_STORP=$(LD_FLAGS_SHLIB) /DEF:rptcpcc.def
LD_FLAGS_LIB=/LIBPATH:$(TPCC_SQLLIB)\lib
/LIBPATH:"C:\MsSDKx64\Lib\AMD64" db2api.lib WinMM.lib
LD_FLAGS_OUT=/OUT:

# Library Configuration
AR=lib.exe
AR_FLAGS=
AR_FLAGS_LIB=
AR_FLAGS_OUT=/OUT:

# OS Commands
ERASE=del /F
ERASEDIR=rmdir /S
MOVE=MOVE
COPY=COPY

# OS File Extensions & Path Separator
OBJEXT=.obj
LIBEXT=.lib
SHLIBEXT=.dll
BINEXT=.exe
SLASH=\\
CMDSEP=&

tpccenv.bat

@REM
*****
*****
@REM Licensed Materials - Property of IBM

```

```

@REM
@REM Governed under the terms of the International
@REM License Agreement for Non-Warranted Sample Code.
@REM
@REM (C) COPYRIGHT International Business Machines Corp. 1996 - 2005
@REM All Rights Reserved.
@REM
@REM US Government Users Restricted Rights - Use, duplication or
@REM disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
@REM
*****
@REM
@REM tpccenv.bat - Windows Environment Setup
@REM

@REM The Kit Version
set TPCC_VERSION=CK041012

@REM The DB2 Instance Name (for DB2)
set DB2INSTANCE=DB2

@REM The OS being used (i.e. "UNIX", "WINDOWS")
set PLATFORM=WINDOWS

@REM The type of make command and slash used by the OS
@REM (i.e. UNIX - "/", WINDOWS - "\")
@REM These are referenced all over the kit.
set SLASH=\
set MAKE=nmake

set TPCC_SPTYPE=SPGENERAL

set DB2VERSION=v8

@REM The schema name is typically the SQL authorization ID (or username).
@REM This is required for runstats and EEE.
set TPCC_SCHEMA=%USERNAME%

@REM DB2 EE/EEE Configuration
set DB2EDITION=EE
set DB2NODE=0
set DB2NODES=1

@REM TPCC General Configuration
set HOME=C
set TPCC_DBNAME=TPCC
set TPCC_ROOT=c:\tpc-c.ibm
set TPCC_SQLLIB=c:\SQLLIB
set TPCC_RUNDATA=c:\tpccdata

@REM TPCC Debug Configuration
set TPCC_DEBUGDIR=c:\temp

@REM Specifies where stored procedures should be placed and if they should
@REM be fenced.
set TPCC_SPDIR=%TPCC_SQLLIB%\function
set TPCC_FENCED=NO

include/db2tpcc.h

/*****
*****
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International

```

```

** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2004
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****
/
* db2tpcc.h - Macros and Miscellany
*/
#ifdef _DB2TPCC_H
#define __DB2TPCC_H
#include <sys/types.h>
#include "lval.h"
/
*****
/* Transaction Return Codes (s_transtatus) */
/
*****
#define INVALID_ITEM 100
#define TRAN_OK 0
#define FATAL_SQLERROR -1
/
*****
/* Definition of Unused and Bad Items */
/
*****
/* Define unused item ID to be 0. This allows the SUT to determine the */
/* number of items in the order as required by 2.4.1.3 and 2.4.2.2 since */
/* the assumption that any item with OL_I_ID = 0 is unused will be true. */
/* This in turn requires that the value used for an invalid item is */
/* equal to ITEMS + 1. */
/
*****
#define INVALID_ITEM_ID (2 * ITEMS) + 1
#define UNUSED_ITEM_ID 0
#define MIN_WAREHOUSE 1
#define MAX_WAREHOUSE WAREHOUSES
/
/* NURand Constants */
/* C_C_LAST_RUN and C_C_LAST_LOAD must adhere to clause 2.1.6. */
/
/* Analysis indicates that a C_LAST delta of 85 is optimal. */
/
*****
#define C_C_LAST_RUN 88
#define C_C_LAST_LOAD 173
#define C_C_ID 319
#define C_OL_I_ID 3849
#define A_C_LAST 255
#define A_C_ID 1023
#define A_OL_I_ID 8191
/
*****
/* Transaction Type Identifiers */
/
*****
#define CLIENT_SQL 0
#define NEWORD_SQL 1
#define PAYMENT_SQL 2

```



```

#define ORDDSTAT_SQL 3
#define DELIVERY_SQL 4
#define STOCKLEV_SQL 5
#define SPGENERAL_PAD 3
#define SPGENERAL_ADJUST sizeof(int16_t)
struct in_neword_struct {
    int16_t len;
    int16_t pad[SPGENERAL_PAD];
    struct in_items_struct {
        int32_t s_OL_I_ID;
        int32_t s_OL_SUPPLY_W_ID;
        int16_t s_OL_QUANTITY;
        int16_t pad1[3];
    } in_item[15];
    int64_t s_O_ENTRY_D_time; /* init by SUT */
    int32_t s_C_ID;
    int32_t s_W_ID;
    int16_t s_D_ID;
    int16_t s_O_OL_CNT; /* init by SUT */
    int16_t s_all_local;
    int16_t duplicate_items;
};
struct out_neword_struct {
    int16_t len;
    int16_t pad[SPGENERAL_PAD];
    struct items_struct {
        int32_t s_I_PRICE;
        int32_t s_OL_AMOUNT;
        int16_t s_S_QUANTITY;
        int16_t pad2;
        char s_I_NAME[25];
        char s_brand_generic;
    } item[15];
    int64_t s_O_ENTRY_D_time;
    int32_t s_W_TAX;
    int32_t s_D_TAX;
    int32_t s_C_DISCOUNT;
    int32_t s_total_amount;
    int32_t s_O_ID;
    int16_t s_O_OL_CNT;
    int16_t s_transtatus;
    int16_t deadlocks;
    char s_C_LAST[17];
    char s_C_CREDIT[3];
};
struct in_payment_struct {
    int16_t len;
    int16_t pad[SPGENERAL_PAD];
    int64_t s_H_DATE_time; /* init by SUT */
    int64_t s_H_AMOUNT;
    int32_t s_W_ID;
    int32_t s_C_W_ID;
    int32_t s_C_ID;
    int16_t s_C_D_ID;
    int16_t s_D_ID;
    char s_C_LAST[17];
};
struct out_payment_struct {
    int16_t len;
    int16_t pad[SPGENERAL_PAD];
    int64_t s_H_DATE_time;
    int64_t s_C_SINCE_time;
    int64_t s_C_CREDIT_LIM;
    int64_t s_C_BALANCE;
    int32_t s_C_DISCOUNT;
    int32_t s_C_ID;
    int16_t s_transtatus;
    int16_t deadlocks;
};
char s_W_STREET_1[21];
char s_W_STREET_2[21];
char s_W_CITY[21];
char s_W_STATE[3];
char s_W_ZIP[10];
char s_D_STREET_1[21];
char s_D_STREET_2[21];
char s_D_CITY[21];
char s_D_STATE[3];
char s_D_ZIP[10];
char s_C_FIRST[17];
char s_C_MIDDLE[3];
char s_C_LAST[17];
char s_C_STREET_1[21];
char s_C_STREET_2[21];
char s_C_CITY[21];
char s_C_STATE[3];
char s_C_ZIP[10];
char s_C_PHONE[17];
char s_C_CREDIT[3];
char s_C_DATA[201];
};
struct in_ordstat_struct {
    int16_t len;
    int16_t pad[SPGENERAL_PAD];
    int32_t s_C_ID;
    int32_t s_W_ID;
    int16_t s_D_ID;
    int16_t pad1[3];
    char s_C_LAST[17];
};
struct out_ordstat_struct {
    int16_t len;
    int16_t pad[SPGENERAL_PAD];
    int64_t s_C_BALANCE;
    int64_t s_O_ENTRY_D_time;
    int32_t s_C_ID;
    int32_t s_O_ID;
    int16_t s_O_CARRIER_ID;
    int16_t s_ol_cnt;
    int16_t pad1[2];
    struct oitems_struct {
        int64_t s_OL_DELIVERY_D_time;
        int32_t s_OL_AMOUNT;
        int32_t s_OL_I_ID;
        int32_t s_OL_SUPPLY_W_ID;
        int16_t s_OL_QUANTITY;
        int16_t pad2;
    } item[15];
    int16_t s_transtatus;
    int16_t deadlocks;
    char s_C_FIRST[17];
    char s_C_MIDDLE[3];
    char s_C_LAST[17];
};
struct in_delivery_struct {
    int16_t len;
    int16_t pad[SPGENERAL_PAD];
    int64_t s_O_DELIVERY_D_time; /* init by SUT */
    int32_t s_W_ID;
    int16_t s_O_CARRIER_ID;
};
struct out_delivery_struct {
    int16_t len;
    int16_t pad[SPGENERAL_PAD];
    int32_t s_O_ID[10];
    int16_t s_transtatus;
};

```

```

int16_t deadlocks;
};
struct in_stocklev_struct {
int16_t len;
int16_t pad[SPGENERAL_PAD];
int32_t s_threshold;
int32_t s_W_ID;
int16_t s_D_ID;
};
struct out_stocklev_struct {
int16_t len;
int16_t pad[SPGENERAL_PAD];
int32_t s_low_stock;
int16_t s_transtatus;
int16_t deadlocks;
};
/*
***** */
/* Transaction Prototypes */
/*
***** */
#ifdef __cplusplus
extern "C" {
#endif
extern int neword_sql(struct in_neword_struct*, struct out_neword_struct*);
extern int payment_sql(struct in_payment_struct*, struct out_payment_struct*);
extern int ordstat_sql(struct in_ordstat_struct*, struct out_ordstat_struct*);
extern int delivery_sql(struct in_delivery_struct*, struct out_delivery_struct*);
extern int stocklev_sql(struct in_stocklev_struct*, struct out_stocklev_struct*);
#ifdef __cplusplus
}
#endif
/*
***** */
/* DB2 Connect/Disconnect & Thread Context Wrappers */
/*
***** */
#ifdef __cplusplus
extern "C" {
#endif
extern int connect_to_TM(char*);
extern int connect_to_TM_auth(char*, char*, char*);
extern int disconnect_from_TM(void);

#ifdef __cplusplus
}
#endif
#endif // __DB2TPCC_H

```

include/lval.h

```

#ifdef __LVAL_H
#define __LVAL_H
#define WAREHOUSES 20000
#define DISTRICTS_PER_WAREHOUSE 10
#define CUSTOMERS_PER_DISTRICT 3000
#define ITEMS 100000
#define STOCK_PER_WAREHOUSE 100000
#define MIN_OL_PER_ORDER 5
#define MAX_OL_PER_ORDER 15
#define NU_ORDERS_PER_DISTRICT 900
#endif // __LVAL_H

```

include/tpccapp.h

```

/*****
*****
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2004
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****
*****/
/*
* tpccapp.h - Application Macros
*/
#ifndef __TPCCAPP_H
#define __TPCCAPP_H
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <string.h>
#include <time.h>
#define daricall
#include "sqlca.h"
#include "sqlcodes.h"
#ifdef SWAP_ENDIAN
#define SWAP_BYTE(Var) SwapEndian((void*)&Var, sizeof(Var))
/*****
*****
FUNCTION: SwapEndian
PURPOSE: Swap the byte order of a structure
EXAMPLE: int I=0x12345678; SWAP_BYTE(I); I=> 0x78563412;
IMPLEMENTATION: Fold Addr in half, swap header & tail by XOR op
e.g.: *a = 0x12 [ Addr + 0];
       *b = 0x78 [ Addr + 4 - 0 - 1 = Addr+3];
       *a ^= *b; // sets *a to 0x6A
       *b ^= *a; // sets *b to 0x12
       *a ^= *b; // sets *a to 0x78
       Now *a => 0x78 && *b => 0x12
*****
*****/
void SwapEndian(void *Addr, int nb)
{
int i;
for (i=0; i<nb/2; i++)
{
char *a = (char*)Addr+i;
char *b = (char*)Addr+(nb-i-1);
*a ^= *b;
*b ^= *a;
*a ^= *b;
}
}
#endif //SWAP_ENDIAN

/*****
*****
** SQLCODE Macros */
/*****
*****/
#define DLCHK(a) \
if (sqlca.sqlcode == SQL_RC_E911) { goto a; }
#define NACOMPCHK(last) \
if (sqlca.sqlcode != SQL_RC_E1339) { last = -1; } \

```

```

else { int a = ((sqlca.sqlerrmc[4] == 0x20) ? 0 : sqlca.sqlerrmc[4]-0x30); \
      int b = ((sqlca.sqlerrmc[5] == 0x20) ? 0 : sqlca.sqlerrmc[5]-0x30); \
      if (b == 0) { last = a; } else { last = a * 10 + b; } \
    }
#endif // __TPCCAPP_H

```

include/tpccdbg.h

```

/*****
*****
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2004
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****
*****/
/*
 * tpccdbg.h - Debugging Macros
 */
#ifndef __TPCCDBG_H
#define __TPCCDBG_H
#ifdef __cplusplus
extern "C" {
#endif
extern void new_debug (struct out_neword_struct *neword_ptr,
                      struct in_neword_struct *in_neword_ptr,
                      char *msg);
extern void pay_debug (struct out_payment_struct *payment_ptr,
                      struct in_payment_struct *in_payment_ptr,
                      char *msg);
extern void ord_debug (struct out_ordstat_struct *ordstat_ptr,
                      struct in_ordstat_struct *in_ordstat_ptr,
                      char *msg);
extern void del_debug (struct out_delivery_struct *delivery_ptr,
                      struct in_delivery_struct *in_delivery_ptr,
                      char *msg);
extern void stk_debug (struct out_stocklev_struct *stocklev_ptr,
                      struct in_stocklev_struct *in_stocklev_ptr,
                      char *msg);
extern void new_print (struct out_neword_struct *neword_ptr,
                      struct in_neword_struct *in_neword_ptr,
                      char *filename,
                      char *msg);
extern void pay_print (struct out_payment_struct *payment_ptr,
                      struct in_payment_struct *in_payment_ptr,
                      char *filename,
                      char *msg);
extern void ord_print (struct out_ordstat_struct *ordstat_ptr,
                      struct in_ordstat_struct *in_ordstat_ptr,
                      char *filename,
                      char *msg);
extern void del_print (struct out_delivery_struct *delivery_ptr,
                      struct in_delivery_struct *in_delivery_ptr,
                      char *filename,
                      char *msg);
extern void stk_print (struct out_stocklev_struct *stocklev_ptr,
                      struct in_stocklev_struct *in_stocklev_ptr,
                      char *filename,
                      char *msg);
#ifdef __cplusplus
}
#endif

```

```

#endif // __TPCCDBG_H

```

Src.Common/Makefile

```

#####
#####
## Licensed Materials - Property of IBM
##
## Governed under the terms of the International
## License Agreement for Non-Warranted Sample Code.
##
## (C) COPYRIGHT International Business Machines Corp. 1996 - 2005
## All Rights Reserved.
##
## US Government Users Restricted Rights - Use, duplication or
## disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
#####
#####
#
# Makefile - Makefile for Src.Common
#

!include $(TPCC_ROOT)/Makefile.config

#
#####
#####
# Preprocessor, Compiler and Linker Flags
#
#####
#####
BND_OPTS =      GRANT PUBLIC \
                MESSAGES $*.bnd.msg
PRP_OPTS =      BINDFILE \
                OPTLEVEL 1 \
                ISOLATION RR \
                MESSAGES $*.prep.msg \
                LEVEL $(TPCC_VERSION) \
                NOLINEMACRO

INCLUDES =      -I$(TPCC_SQLLIB)$(SLASH)include
                -I$(TPCC_ROOT)$(SLASH)include

CFLAGS =        $(CFLAGS_OS) $(CFLAGS_DEBUG) $(INCLUDES) \
                -DSQLA_NOLINES -D$(DB2EDITION)
                -D$(DB2VERSION) \
                -D$(TPCC_SPTYPE)

UTIL_OBJ =      tpccmisc$(OBJEXT) tpccdbg$(OBJEXT)
tpccctx$(OBJEXT)

#
#####
#####
# User Targets
#
#####
#####
all:            connect $(UTIL_OBJ) disconnect

clean:
                - $(ERASE) *$(OBJEXT) *.bnd *.msg tpccctx.c

```

```

#
#####
#####
# Helper Targets
#
#####
#####

connect:
    - db2 connect to $(TPCC_DBNAME)

disconnect:
    - db2 connect reset
    - db2 terminate

rebind:
    db2 bind tpcctx.bnd $(BND_OPTS)

#
#####
#####
# Build Rules
#
#####
#####

.SUFFIXES:
.SUFFIXES: $(OBJEXT) .c .sqc

.sqc.c:
    @echo "Prepping $*.sqc"
    -db2 prep $*.sqc $(PRP_OPTS)
    @echo "Binding $*.bnd"
    db2 bind $*.bnd $(BND_OPTS)

#
#####
#####
# Dependencies
#
#####
#####

# Source
tpccdbg$(OBJEXT): tpcdbg.c
tpcctx$(OBJEXT): tpcctx.c
tpcmisc$(OBJEXT): tpcmisc.c

# Headers
tpccdbg.c: $(TPCC_ROOT)/include/db2tpcc.h

Src.Common/tpcctx.sqc

/*****
*****
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2004
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****
*****/
/*

```

```

* tpcctx.sqc - TPCC context code
*/
#include <stdlib.h>
#include <stdio.h>
#include <sqlutil.h>
#include "db2tpcc.h"

int connect_to_TM(char *in_dbname);
int connect_to_TM_auth(char *in_dbname, char *in_username, char
    *in_password);
int disconnect_from_TM(void);
int connect_to_TM(char *in_dbname)
{
    return connect_to_TM_auth(in_dbname, "", "");
}

int connect_to_TM_auth(char *in_dbname, char *in_username, char
    *in_password)
{
    SQL_STRUCTURE sqlca sqlca;
    int ConnectSQLCODE = 0;
    EXEC SQL BEGIN DECLARE SECTION;
    char dbname[9];
    char username[129];
    char password[15];
    EXEC SQL END DECLARE SECTION;
    strncpy(dbname,in_dbname,8);
    if (strcmp(in_username,"") == 0)
    {
        EXEC SQL CONNECT TO :dbname IN SHARE MODE;
    } else {
        strncpy(username,in_username,128);
        strncpy(password,in_password,14);
        EXEC SQL CONNECT TO :dbname IN SHARE MODE USER :username
        USING :password;
    }
    ConnectSQLCODE = SQLCODE;
    if (ConnectSQLCODE != 0)
    {
        sqlerror( CLIENT_SQL, "CONNECT", __FILE__, __LINE__, &sqlca);
        return ConnectSQLCODE;
    }
    return 0;
}

int disconnect_from_TM(void)
{
    SQL_STRUCTURE sqlca sqlca;
    int DisconnectSQLCODE = 0;
    EXEC SQL CONNECT RESET;
    DisconnectSQLCODE = SQLCODE;
    if (DisconnectSQLCODE != 0) {
        sqlerror( CLIENT_SQL, "DISCONNECT", __FILE__, __LINE__, &sqlca);
    }
    if (DisconnectSQLCODE) {
        return DisconnectSQLCODE;
    }
    return 0;
}

Src.Common/tpccdbg.c

/*****
*****
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2004
** All Rights Reserved.
**

```

```

** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****
*****/
/*
 * tcdbg.c - Debugging Routines
 */
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <ctype.h>
#include <time.h>
#include "sqlca.h"
#include "sql.h"
#include "db2tpcc.h"
#include "tpccdbg.h"
#define DEBUG_FILENAME_SZ 128
#define DEBUG_PATH_SIZE 128
void del_print();
void new_print();
void ord_print();
void pay_print();
void stk_print();
void current_tmstamp(char *buf);
static int debugInit = 0;
static char debugPath[DEBUG_PATH_SIZE] = "";
/*-----*/
/* InitializeDebug */
/*-----*/
__inline void InitializeDebug(void) {
    if (debugInit == 0) {
        char *p = getenv("TPCC_DEBUGDIR");
        if (p) {
            strncpy(debugPath, p, DEBUG_PATH_SIZE);
        } else {
            strcpy(debugPath, "/tmp");
        }
        strcat(debugPath, "/");
    }
    debugInit = 1;
}
/*-----*/
/* sqlerror */
/*-----*/
void sqlerror(int tranType, char *msg, char *file, int line, SQL_STRUCTURE
sqlca *psqlca)
{
    FILE *err_fn = NULL;
    char err_fn[DEBUG_PATH_SIZE + DEBUG_FILENAME_SZ];
    char tranName[16];
    int j,k;
    char timeStamp[27];
    char errStr[512] = "";
    InitializeDebug();
    strncpy(err_fn, debugPath, DEBUG_PATH_SIZE);
    current_tmstamp(&timeStamp[0]);
    timeStamp[19] = (char)NULL;
    switch(tranType)
    {
        case NEWORD_SQL:
            // sprintf(err_fn, "%d.err.out", getpid());
            strcat(err_fn, "new.err.out");
            strcpy(tranName, "NEW_ORDER");
            break;
        case DELIVERY_SQL:
            // sprintf(err_fn, "%d.err.out", getpid());
            strcat(err_fn, "del.err.out");
            strcpy(tranName, "DELIVERY");

```

```

            break;
        case PAYMENT_SQL:
            // sprintf(err_fn, "%d.err.out", getpid());
            strcat(err_fn, "pay.err.out");
            strcpy(tranName, "PAYMENT");
            break;
        case ORDSTAT_SQL:
            // sprintf(err_fn, "%d.err.out", getpid());
            strcat(err_fn, "ord.err.out");
            strcpy(tranName, "ORDER_STAT");
            break;
        case STOCKLEV_SQL:
            //sprintf(err_fn, "%d.err.out", getpid());
            strcat(err_fn, "stk.err.out");
            strcpy(tranName, "STOCK_LVL");
            break;
        case 0:
            strcat(err_fn, "cli.err.out");
            strcpy(tranName, "CLIENT");
            break;
        default:
            return;
    }
}
/* Generate Formatted Error Message */
sqlaintp(errStr, 512, 78, psqlca);
err_fn = fopen(err_fn, "a+");
fprintf(err_fn, "-----\n");
fprintf(err_fn, "Transaction: %s (%s)\n", tranName, msg);
fprintf(err_fn, "FILE %s (%u)\n", file, line);
fprintf(err_fn, "SQLCODE %d ", psqlca->sqlcode);
fprintf(err_fn, "PID %d ", getpid());
fprintf(err_fn, "TIME %s\n", timeStamp);
fprintf(err_fn, "-----\n");
fprintf(err_fn, "%s", errStr);
fprintf(err_fn, "-----\n");
if (psqlca->sqlerrmc[0] != '' || psqlca->sqlerrmc[1] != '')
{
    fprintf(err_fn, "slerrmc: ");
    for(j = 0; j < 5; j++)
    {
        for(k = 0; k < 16; k++) {
            int pos = j * 16 + k;
            if (pos < 70) fprintf(err_fn, "%02x ", psqlca->sqlerrmc[pos]);
            else fprintf(err_fn, " ");
        }
        fprintf(err_fn, " |");
        for(k = 0; k < 16; k++) {
            int pos = j * 16 + k;
            char c = ' ';
            if (pos < 70) {
                c = psqlca->sqlerrmc[pos];
                if (!isprint(c)) c = ' ';
            }
            fprintf(err_fn, "%c", c);
        }
        fprintf(err_fn, "\n");
        if (j < 4) fprintf(err_fn, " ");
    }
}
fprintf(err_fn, "sqlerrp: ");
for(j = 0; j < 8; j++)
    fprintf(err_fn, "%c", psqlca->sqlerrp[j]);
fprintf(err_fn, "\n");
fprintf(err_fn, "sqlerrd: ");
for(j = 0; j < 6; j++)
    fprintf(err_fn, " %d", psqlca->sqlerrd[j]);
fprintf(err_fn, "\n");
if (psqlca->sqlwarn[0] != '')

```

```

{
    fprintf(err_fp, "sqlwarn: ");
    for(j = 0; j < 8; j++)
        fprintf(err_fp, "%c ", psqlca->sqlwarn[j]);
    fprintf(err_fp, "\n");
}
fprintf(err_fp, "\n");
fclose(err_fp);
}

/*-----*/
/* del_debug */
/*-----*/
void del_debug (struct out_delivery_struct *delivery_ptr,
                struct in_delivery_struct *in_delivery,
                char *msg)
{
    char debug_fn[DEBUG_PATH_SIZE + DEBUG_FILENAME_SZ];

    InitializeDebug();
    strncpy(debug_fn, debugPath, DEBUG_PATH_SIZE);
    strcat(debug_fn, "del.debug.out");
    del_print(delivery_ptr, in_delivery, debug_fn, msg);
}

/*-----*/
/* del_print */
/*-----*/
void del_print (struct out_delivery_struct *delivery_ptr,
                struct in_delivery_struct *in_delivery,
                char *filename,
                char *msg)
{
    FILE *debug_fp;
    char timeStamp[27];
    int j;
    current_tmstamp(&timeStamp[0]);
    timeStamp[19] = (char)NULL;
    if ((debug_fp = fopen(filename, "a+")) == NULL)
    {
        return;
    }
    fprintf(debug_fp, "Delivery debug information follows %s (%s)\n",
timeStamp, msg);
    fprintf(debug_fp, " PID %d ", getpid());

    fprintf(debug_fp, "\n=====
=====\\n");
    fprintf(debug_fp, "in_delivery_struct {\n");
    fprintf(debug_fp, "ts_W_ID = %d (%X)\n",
in_delivery->s_W_ID, in_delivery->s_W_ID);
    fprintf(debug_fp, "ts_O_CARRIER_ID = %d (%X)\n",
in_delivery->s_O_CARRIER_ID, in_delivery->s_O_CARRIER_ID);
    fprintf(debug_fp, "ts_O_DELIVERY_D = %lld (%lX)\n",
in_delivery->s_O_DELIVERY_D_time,
in_delivery->s_O_DELIVERY_D_time);
    fprintf(debug_fp, "}\n");
    fprintf(debug_fp, "out_delivery_struct {\n");
    fprintf(debug_fp, "ts_transtatus = %d (%X)\n",
delivery_ptr->s_transtatus, delivery_ptr->s_transtatus);
    fprintf(debug_fp, "tdeadlocks = %d (%X)\n",
delivery_ptr->deadlocks, delivery_ptr->deadlocks);
    for (j = 0; j < 10; j++) {
        fprintf(debug_fp, "ts_O_ID[%d] = %d\n",
j, delivery_ptr->s_O_ID[j]);
    }
    fprintf(debug_fp, "t}\n}\n");
    fclose(debug_fp);
}

/*-----*/
/* new_debug */
/*-----*/
void new_debug (struct out_neword_struct *neword_ptr,
                struct in_neword_struct *in_neword,
                char *msg)
{
    char debug_fn[DEBUG_PATH_SIZE + DEBUG_FILENAME_SZ];

    InitializeDebug();
    strncpy(debug_fn, debugPath, DEBUG_PATH_SIZE);
    strcat(debug_fn, "new.debug.out");
    new_print(neword_ptr, in_neword, debug_fn, msg);
}

/*-----*/
/* new_print */
/*-----*/
void new_print (struct out_neword_struct *neword_ptr,
                struct in_neword_struct *in_neword,
                char *filename,
                char *msg)
{
    FILE *debug_fp;
    char timeStamp[27];
    int j, items;
    current_tmstamp(&timeStamp[0]);
    timeStamp[19] = (char)NULL;
    if ((debug_fp = fopen(filename, "a+")) == NULL)
    {
        return;
    }
    fprintf(debug_fp, "New order debug information follows %s (%s)\n",
timeStamp, msg);
    fprintf(debug_fp, " PID %d ", getpid());

    fprintf(debug_fp, "\n=====
=====\\n");
    fprintf(debug_fp, "in_neword_struct {\n");
    fprintf(debug_fp, "ts_C_ID = %d (%X)\n",
in_neword->s_C_ID, in_neword->s_C_ID);
    fprintf(debug_fp, "ts_W_ID = %d (%X)\n",
in_neword->s_W_ID, in_neword->s_W_ID);
    fprintf(debug_fp, "ts_D_ID = %d (%X)\n",
in_neword->s_D_ID, in_neword->s_D_ID);
    fprintf(debug_fp, "ts_O_OL_CNT = %d (%X)\n",
in_neword->s_O_OL_CNT, in_neword->s_O_OL_CNT);
    fprintf(debug_fp, "ts_all_local = %d (%X)\n",
in_neword->s_all_local, in_neword->s_all_local);
    fprintf(debug_fp, "ts_O_ENTRY_D = %lld (%lX)\n",
in_neword->s_O_ENTRY_D_time, in_neword->s_O_ENTRY_D_time);
    // fprintf(debug_fp, "ts_transtatus = %d (%X)\n",
// in_neword->s_transtatus, in_neword->s_transtatus);
    // fprintf(debug_fp, "tduplicate_items= %d (%X)\n",
// in_neword->duplicate_items, in_neword->duplicate_items);
    fprintf(debug_fp, "titems {\n");
    items = in_neword->s_O_OL_CNT;
    for (j=0; j<items; j++) {
        if(j != 0)
            fprintf(debug_fp, "\n");
        fprintf(debug_fp, "ts_OL_I_ID[%d] = %d (%X)\n",
j, in_neword->in_item[j].s_OL_I_ID,
in_neword->in_item[j].s_OL_I_ID);
        fprintf(debug_fp, "ts_OL_SUPPLY_W_ID[%d] = %d (%X)\n",
j, in_neword->in_item[j].s_OL_SUPPLY_W_ID,
in_neword->in_item[j].s_OL_SUPPLY_W_ID);
        fprintf(debug_fp, "ts_OL_QUANTITY[%d] = %d (%X)\n",

```

```

        j, in_neword->in_item[j].s_OL_QUANTITY,
in_neword->in_item[j].s_OL_QUANTITY);
    }
    fprintf(debug_fp, "\n");
    fprintf(debug_fp, "out_neword_struct {\n");
    fprintf(debug_fp, "ts_C_LAST = %s\n",
        neword_ptr->s_C_LAST);
    fprintf(debug_fp, "ts_C_CREDIT = %s\n",
        neword_ptr->s_C_CREDIT);
    fprintf(debug_fp, "ts_W_TAX = %d\n",
        neword_ptr->s_W_TAX);
    fprintf(debug_fp, "ts_D_TAX = %d\n",
        neword_ptr->s_D_TAX);
    fprintf(debug_fp, "ts_C_DISCOUNT = %d\n",
        neword_ptr->s_C_DISCOUNT);
    fprintf(debug_fp, "ts_O_ID = %d (%X)\n",
        neword_ptr->s_O_ID, neword_ptr->s_O_ID);
    fprintf(debug_fp, "ts_O_OL_CNT = %d (%X)\n",
        neword_ptr->s_O_OL_CNT, neword_ptr->s_O_OL_CNT);
    fprintf(debug_fp, "ts_O_ENTRY_D = %lld (%lX)\n",
        neword_ptr->s_O_ENTRY_D_time,
neword_ptr->s_O_ENTRY_D_time);
    fprintf(debug_fp, "ts_total_amount = %d\n",
        neword_ptr->s_total_amount);
    fprintf(debug_fp, "ts_transtatus = %d (%X)\n",
        neword_ptr->s_transtatus, neword_ptr->s_transtatus);
    fprintf(debug_fp, "tdeadlocks = %d (%X)\n",
        neword_ptr->deadlocks, neword_ptr->deadlocks);
// fprintf(debug_fp, "ts_W_ID = %d (%X)\n",
//     neword_ptr->s_W_ID, neword_ptr->s_W_ID);
// fprintf(debug_fp, "ts_D_ID = %d (%X)\n",
//     neword_ptr->s_D_ID, neword_ptr->s_D_ID);
// fprintf(debug_fp, "ts_all_local = %d (%X)\n",
//     neword_ptr->s_all_local, neword_ptr->s_all_local);
// fprintf(debug_fp, "tduplicate_items= %d (%X)\n",
//     neword_ptr->duplicate_items, neword_ptr->duplicate_items);
    fprintf(debug_fp, "titems {\n");
    items = neword_ptr->s_O_OL_CNT;
    for (j=0; j<items; j++) {
        if(j != 0)
            fprintf(debug_fp, "\n");
        fprintf(debug_fp, "tts_I_NAME[%d] = %s\n",
            j, neword_ptr->item[j].s_I_NAME);
        fprintf(debug_fp, "tts_I_PRICE[%d] = %d\n",
            j, neword_ptr->item[j].s_I_PRICE);
        fprintf(debug_fp, "tts_OL_AMOUNT[%d] = %d\n",
            j, neword_ptr->item[j].s_OL_AMOUNT);
        fprintf(debug_fp, "tts_S_QUANTITY[%d] = %d (%X)\n",
            j, neword_ptr->item[j].s_S_QUANTITY,
neword_ptr->item[j].s_S_QUANTITY);
        fprintf(debug_fp, "tts_brand_generic[%d] = %c\n",
            j, neword_ptr->item[j].s_brand_generic);
    }
    fprintf(debug_fp, "\n");
    fclose(debug_fp);
}

/*-----*/
/* ord_debug */
/*-----*/
void ord_debug (struct out_ordstat_struct *ordstat_ptr,
    struct in_ordstat_struct *in_ordstat,
    char *msg)
{
    char debug_fn[DEBUG_PATH_SIZE + DEBUG_FILENAME_SZ];

    InitializeDebug();

```

```

    strncpy(debug_fn, debugPath, DEBUG_PATH_SIZE);
    strcat(debug_fn, "ord.debug.out");
    ord_print(ordstat_ptr, in_ordstat, debug_fn, msg);
}
/*-----*/
/* ord_print */
/*-----*/
void ord_print (struct out_ordstat_struct *ordstat_ptr,
    struct in_ordstat_struct *in_ordstat,
    char *filename,
    char *msg)
{
    FILE *debug_fp;
    char timeStamp[27];
    int j, items;
    current_tmstamp(&timeStamp[0]);
    timeStamp[19] = (char)NULL;
    if ((debug_fp = fopen(filename, "a+")) == NULL)
    {
        return;
    }
    fprintf(debug_fp, "Order status debug information follows %s (%s)\n",
timeStamp, msg);
    fprintf(debug_fp, " PID %d ", getpid());

    fprintf(debug_fp, "\n=====
=====
\n");
    fprintf(debug_fp, "in_ordstat_struct {\n");
    fprintf(debug_fp, "ts_W_ID = %d (%X)\n",
        in_ordstat->s_W_ID, in_ordstat->s_W_ID);
    fprintf(debug_fp, "ts_D_ID = %d (%X)\n",
        in_ordstat->s_D_ID, in_ordstat->s_D_ID);
    fprintf(debug_fp, "ts_C_ID = %d (%X)\n",
        in_ordstat->s_C_ID, in_ordstat->s_C_ID);
    fprintf(debug_fp, "ts_C_LAST = %s\n",
        in_ordstat->s_C_LAST);
    fprintf(debug_fp, "}\n");
    fprintf(debug_fp, "out_ordstat_struct {\n");
    fprintf(debug_fp, "ts_C_ID = %d (%X)\n",
        ordstat_ptr->s_C_ID, ordstat_ptr->s_C_ID);
    fprintf(debug_fp, "ts_C_FIRST = %s\n",
        ordstat_ptr->s_C_FIRST);
    fprintf(debug_fp, "ts_C_MIDDLE = %s\n",
        ordstat_ptr->s_C_MIDDLE);
    fprintf(debug_fp, "ts_C_LAST = %s\n",
        ordstat_ptr->s_C_LAST);
    fprintf(debug_fp, "ts_C_BALANCE = %lld\n",
        ordstat_ptr->s_C_BALANCE);
    fprintf(debug_fp, "ts_O_ID = %d (%X)\n",
        ordstat_ptr->s_O_ID, ordstat_ptr->s_O_ID);
    fprintf(debug_fp, "ts_O_ENTRY_D = %lld (%lX)\n",
        ordstat_ptr->s_O_ENTRY_D_time, ordstat_ptr->s_O_ENTRY_D_time);
    fprintf(debug_fp, "ts_O_CARRIER_ID = %d (%X)\n",
        ordstat_ptr->s_O_CARRIER_ID, ordstat_ptr->s_O_CARRIER_ID);
    fprintf(debug_fp, "ts_ol_cnt = %d (%X)\n",
        ordstat_ptr->s_ol_cnt, ordstat_ptr->s_ol_cnt);
    fprintf(debug_fp, "ts_transtatus = %d (%X)\n",
        ordstat_ptr->s_transtatus, ordstat_ptr->s_transtatus);
    fprintf(debug_fp, "tdeadlocks = %d (%X)\n",
        ordstat_ptr->deadlocks, ordstat_ptr->deadlocks);
    fprintf(debug_fp, "titems {\n");
    items = ordstat_ptr->s_ol_cnt;
    for (j = 0; j < items; j++) {
        if(j != 0)
            fprintf(debug_fp, "\n");
        fprintf(debug_fp, "tts_OL_SUPPLY_W_ID[%d] = %d (%X)\n",
            j, ordstat_ptr->item[j].s_OL_SUPPLY_W_ID,
ordstat_ptr->item[j].s_OL_SUPPLY_W_ID);

```



```

        struct in_stocklev_struct *in_stocklev,
        char *msg)
{
    char debug_fn[DEBUG_PATH_SIZE + DEBUG_FILENAME_SZ];

    InitializeDebug();
    strncpy(debug_fn, debugPath, DEBUG_PATH_SIZE);
    strcat(debug_fn, "stk.debug.out");
    stk_print(stocklev, in_stocklev, debug_fn, msg);
}
/*-----*/
/* stk_print */
/*-----*/
void stk_print (struct out_stocklev_struct *stocklev,
                struct in_stocklev_struct *in_stocklev,
                char *filename,
                char *msg)
{
    FILE *debug_fp;
    char timeStamp[27];
    current_tmstamp(&timeStamp[0]);
    timeStamp[19] = (char)NULL;
    if ((debug_fp = fopen(filename, "a+")) == NULL)
    {
        return;
    }
    fprintf(debug_fp, "Stock level debug information follows %s (%s)\n",
timeStamp, msg);
    fprintf(debug_fp, " PID %d ", getpid());

fprintf(debug_fp, "\n=====
=====");
    fprintf(debug_fp, "in_stocklev_struct {\n");
    fprintf(debug_fp, "\ts_W_ID      = %d (%X)\n",
        in_stocklev->s_W_ID, in_stocklev->s_W_ID);
    fprintf(debug_fp, "\ts_D_ID      = %d (%X)\n",
        in_stocklev->s_D_ID, in_stocklev->s_D_ID);
    fprintf(debug_fp, "\ts_threshold = %d (%X)\n",
        in_stocklev->s_threshold, in_stocklev->s_threshold);
    fprintf(debug_fp, "}\n\n");
    fprintf(debug_fp, "out_stocklev_struct {\n");
    fprintf(debug_fp, "\ts_transtatus = %d (%X)\n",
        stocklev->s_transtatus, stocklev->s_transtatus);
    fprintf(debug_fp, "\ts_deadlocks  = %d (%X)\n",
        stocklev->s_deadlocks, stocklev->s_deadlocks);
    fprintf(debug_fp, "\ts_low_stock  = %d (%X)\n",
        stocklev->s_low_stock, stocklev->s_low_stock);
    fprintf(debug_fp, "}\n\n");
    fclose(debug_fp);
}
void current_tmstamp(char *buf)
{
    time_t t = time(NULL);
    strncpy(buf,ctime(&t),19);
}

```

Src.Common/tpccmisc.c

```

/*****
*****
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2004
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or

```

```

** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****
*****/
/*
 * tpccmisc.c - Miscellaneous routines
 */
#include <stdlib.h>
#include <sys/types.h>
#include <sys/time.h>
double current_time_ms(void);
double current_time(void);

/* Current time in SECONDS, precision SECONDS */
double current_time(void)
{
    /* use time() to get seconds */
    return(time(NULL));
}
/* Current time in SECONDS, precision MILLISECONDS */
double current_time_ms(void)
{
    /* gettimeofday() returns seconds and microseconds */
    /* convert to fractional seconds */
    struct timeval t;
    gettimeofday(&t,NULL);
    return (t.tv_sec + (double)t.tv_usec/(1000*1000));
}

```

Src.Srv/Makefile

```

#####
#####
## Licensed Materials - Property of IBM
##
## Governed under the terms of the International
## License Agreement for Non-Warranted Sample Code.
##
## (C) COPYRIGHT International Business Machines Corp. 1996 - 2005
## All Rights Reserved.
##
## US Government Users Restricted Rights - Use, duplication or
## disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
#####
#####
#
# Makefile - Makefile for Src.Srv
#

!include $(TPCC_ROOT)/Makefile.config

#
#####
#####
# Preprocessor, Compiler and Linker Flags
#
#####
#####
BND_OPTS =      GRANT PUBLIC \
                MESSAGES $*.bnd.msg
PRP_OPTS =      BINDFILE \
                EXPLAIN ALL \
                MESSAGES $*.prep.msg

INCLUDES =      -I$(TPCC_SQLLIB)$(SLASH)include
                -I$(TPCC_ROOT)$(SLASH)include

CFLAGS =        $(CFLAGS_OS) $(INCLUDES) $(CFLAGS_DEBUG) \

```

```

-D$(DB2EDITION) -D$(DB2VERSION) \
-DSQLA_NOLINES -DLINT_ARGS

LD_FLAGS = $(LD_FLAGS_STORP) $(LD_FLAGS_LIB)

#
#####
#####
# File Collections
#
#####
#####

STORED_PROCS = new ord del

UTIL_OBJ = $(TPCC_ROOT)/Src.Common/tpccmisc$(OBJEXT) \
$(TPCC_ROOT)/Src.Common/tpccdbg$(OBJEXT)

DLL = rpctpc$(SHLIBEXT)

#
#####
#####
# User Targets
#
#####
#####

all: connect explain catalog $(DLL) install plan disconnect

clean: connect uncatalog unexplain disconnect
      - $(ERASE) $(TPCC_SPDIR)$(SLASH)rpctpc$(SHLIBEXT)
      - $(ERASE) *.bnd *.msg *.out *$(OBJEXT) $(DLL) tpcc_all_sql.c
      - $(ERASE) TPCC_ALL.*.plan

#
#####
#####
# Helper Targets
#
#####
#####

catalog: uncatalog
      - perl $(TPCC_ROOT)$(SLASH)utils$(SLASH)genproc.pl
$(STORED_PROCS)
      - db2 -tvf cat-proc.ddl +o -z cat-proc.out
      - db2 -td$$ -vf cat-func.ddl +o -z cat-func.out

uncatalog:
      - perl $(TPCC_ROOT)$(SLASH)utils$(SLASH)genproc.pl
$(STORED_PROCS)
      - db2 -td$$ -vf uncat-func.ddl +o -z uncat-func.out
      - db2 -tvf uncat-proc.ddl +o -z uncat-proc.out

explain:
      - perl $(TPCC_ROOT)$(SLASH)utils$(SLASH)fixup_explain.pl
      - db2 -tvf
$(TPCC_ROOT)$(SLASH)utils$(SLASH)EXPLAIN.DDL +o -z EXPLAIN.out

unexplain:
      - db2 -tvf
$(TPCC_ROOT)$(SLASH)utils$(SLASH)UNEXPLAIN.DDL +o -z
UNEXPLAIN.out

connect:
      - db2 connect to $(TPCC_DBNAME)

```

```

disconnect:
      - db2 connect reset
      - db2 terminate

# This (environment) variable is used by db2expln
DB2EXPLN_BUFFER=3000000

plan:
      - db2exfmt -d $(TPCC_DBNAME) -e $(TPCC_SCHEMA) -s
$(TPCC_SCHEMA) -w -l -n TPCC_ALL -g # 0 -o TPCC_ALL.exfmt.plan
      - db2expln -d $(TPCC_DBNAME) -c $(TPCC_SCHEMA) -p
TPCC_ALL -s 0 -g -o TPCC_ALL.expln.plan

rebind: connect catalog
      db2 bind tpcc_all_sql.bnd $(BND_OPTS) QUERYOPT 7

#
#####
#####
# Install Targets
#
#####
#####

install: $(DLL)
      - mkdir $(TPCC_SPDIR)
      $(COPY) $(DLL) $(TPCC_SPDIR)

#
#####
#####
# Build Rules
#
#####
#####

.SUFFIXES: $(OBJEXT) .c .sqc

# d230437mt: QUERYOPT 7 required for UNION ALL
# Only stock needs CS , and that can be specified on the SELECT statement
tpcc_all_sql.c:
      @echo "Prepping $*.sqc"
      -db2 prep $*.sqc $(PRP_OPTS) ISOLATION RR
      @echo "Binding $*.bnd"
      db2 bind $*.bnd $(BND_OPTS) QUERYOPT 7

# Stored procedures are built in a special way

tpcc_all_sql$(OBJEXT):
      $(CC) -c tpcc_all_sql.c $(CFLAGS) -D$(TPCC_SPTYPE)
$(CFLAGS_OUT)$@

$(DLL): $(UTIL_OBJ) tpcc_all_sql$(OBJEXT)
      $(LD_STORP) $(LD_FLAGS) $(UTIL_OBJ)
tpcc_all_sql$(OBJEXT) $(LD_FLAGS_OUT)$@

#
#####
#####
# Dependencies
#
#####
#####

# Executables (Stored Procedures)
$(DLL): $(UTIL_OBJ) tpcc_all_sql$(OBJEXT)

# Source

```

tpcc_all_sql\$(OBJEXT): tpcc_all_sql.c

Headers

tpcc_all_sql.c: \$(TPCC_ROOT)/include/db2tpcc.h

Src.Srv/cat-func.ddl

```
-----
-- Licensed Materials - Property of IBM
--
-- Governed under the terms of the International
-- License Agreement for Non-Warranted Sample Code.
--
-- (C) COPYRIGHT International Business Machines Corp. 1996 - 2004
-- All Rights Reserved.
--
-- US Government Users Restricted Rights - Use, duplication or
-- disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
-----
--
-- cat-func.ddl - Create table functions
--
-- DELIVERY
--
CREATE FUNCTION DEL( W_ID INTEGER
                  , D_ID SMALLINT
                  , CARRIER_ID SMALLINT
                  , DELIVERY_D BIGINT
                  )
RETURNS TABLE( O_ID INTEGER )
SPECIFIC DELIVERY
MODIFIES SQL DATA DETERMINISTIC NO EXTERNAL ACTION
LANGUAGE SQL
VAR: BEGIN ATOMIC
DECLARE O_ID INTEGER ;
DECLARE C_ID INTEGER ;
DECLARE AMOUNT INTEGER ;
/* Delete the order from new order table */
SET VAR.O_ID = ( SELECT NO_O_ID
                FROM OLD TABLE ( DELETE
                                FROM ( SELECT NO_O_ID
                                        FROM NEW_ORDER
                                        WHERE NO_W_ID = DEL.W_ID
                                        AND NO_D_ID = DEL.D_ID
                                        ORDER BY NO_O_ID ASC
                                        FETCH FIRST 1 ROW ONLY
                                ) AS NEW_ORDER
                ) AS D
                )
;
/* Update the order as delivered and retrieve the customer id */
SET VAR.C_ID = ( SELECT O_C_ID
                FROM OLD TABLE ( UPDATE ORDERS
                                SET O_CARRIER_ID = DEL.CARRIER_ID
                                WHERE O_W_ID = DEL.W_ID
                                AND O_D_ID = DEL.D_ID
                                AND O_ID = VAR.O_ID
                                ) AS U
                )
;
SET VAR.AMOUNT = ( SELECT SUM(OL_AMOUNT)
                  FROM OLD TABLE ( UPDATE ORDER_LINE
                                SET OL_DELIVERY_D = DEL.DELIVERY_D
                                WHERE OL_W_ID = DEL.W_ID
                                AND OL_D_ID = DEL.D_ID
                                AND OL_O_ID = VAR.O_ID
```

```

                ) AS U
                )
;
/* Charge the customer */
UPDATE CUSTOMER
SET C_BALANCE = C_BALANCE + VAR.AMOUNT
, C_DELIVERY_CNT = C_DELIVERY_CNT + SMALLINT(1)
WHERE C_W_ID = DEL.W_ID
AND C_D_ID = DEL.D_ID
AND C_ID = VAR.C_ID
;
/* Return the order id to the caller (or NULL) */
RETURN VALUES VAR.O_ID ;
END
$
--
-- ORDER STATUS
--
CREATE FUNCTION ORD_C_LAST( W_ID INTEGER
                          , D_ID SMALLINT
                          , C_LAST VARCHAR(16)
                          )
RETURNS TABLE( O_ID INTEGER
              , O_CARRIER_ID SMALLINT
              , O_ENTRY_D BIGINT
              , C_BALANCE BIGINT
              , C_FIRST VARCHAR(16)
              , C_MIDDLE CHAR(2)
              , C_ID INTEGER
              )
SPECIFIC ORD_C_LAST
READS SQL DATA NO EXTERNAL ACTION DETERMINISTIC
LANGUAGE SQL
VAR: BEGIN ATOMIC
DECLARE C_BALANCE BIGINT ;
DECLARE C_FIRST VARCHAR(16) ;
DECLARE C_MIDDLE CHAR(2) ;
DECLARE C_ID INTEGER ;
DECLARE O_ID INTEGER ;
DECLARE O_CARRIER_ID SMALLINT ;
DECLARE O_ENTRY_D BIGINT ;
/* Retrieve the Customer information */
SET ( C_BALANCE, C_FIRST, C_MIDDLE, C_ID )
= ( SELECT C_BALANCE, C_FIRST, C_MIDDLE, C_ID
    FROM ( SELECT C_ID
          , C_BALANCE
          , C_FIRST
          , C_MIDDLE
          , COUNT(*) OVER() AS COUNT
          , ROWNUMBER() OVER (ORDER BY C_FIRST) AS NUM
          FROM CUSTOMER
          WHERE C_W_ID = ORD_C_LAST.W_ID
          AND C_D_ID = ORD_C_LAST.D_ID
          AND C_LAST = ORD_C_LAST.C_LAST
        ) AS V1
    WHERE NUM = (COUNT + BIGINT(1)) / BIGINT(2)
    )
;
/* Take advantage of the index to fetch the first row (and hence max(o_id)) */
SET ( O_ID, O_CARRIER_ID, O_ENTRY_D )
= ( SELECT O_ID
    , O_CARRIER_ID
    , O_ENTRY_D
    FROM ORDERS
    WHERE O_W_ID = ORD_C_LAST.W_ID
    AND O_D_ID = ORD_C_LAST.D_ID
    AND O_C_ID = VAR.C_ID
```

```

ORDER BY O_ID DESC
FETCH FIRST 1 ROW ONLY
)
;

RETURN VALUES ( VAR.O_ID
,VAR.O_CARRIER_ID
,VAR.O_ENTRY_D
,VAR.C_BALANCE
,VAR.C_FIRST
,VAR.C_MIDDLE
,VAR.C_ID
)
;

END $
CREATE FUNCTION ORD_C_ID( W_ID INTEGER
,D_ID SMALLINT
,C_ID INTEGER
)
RETURNS TABLE( O_ID INTEGER
,O_CARRIER_ID SMALLINT
,O_ENTRY_D BIGINT
,C_BALANCE BIGINT
,C_FIRST VARCHAR(16)
,C_MIDDLE CHAR(2)
,C_LAST VARCHAR(16)
)
SPECIFIC ORD_C_ID
READS SQL DATA NO EXTERNAL ACTION DETERMINISTIC
LANGUAGE SQL
VAR: BEGIN ATOMIC
DECLARE C_BALANCE BIGINT ;
DECLARE C_FIRST VARCHAR(16);
DECLARE C_MIDDLE CHAR(2);
DECLARE C_LAST VARCHAR(16);
DECLARE O_ID INTEGER;
DECLARE O_CARRIER_ID SMALLINT;
DECLARE O_ENTRY_D BIGINT;
/* Retrieve the Customer information */
SET ( C_BALANCE, C_FIRST, C_MIDDLE, C_LAST )
= ( SELECT C_BALANCE, C_FIRST, C_MIDDLE, C_LAST
FROM CUSTOMER
WHERE C_ID = ORD_C_ID.C_ID
AND C_W_ID = ORD_C_ID.W_ID
AND C_D_ID = ORD_C_ID.D_ID
)
;
SET (O_ID, O_CARRIER_ID, O_ENTRY_D)
= ( SELECT O_ID
,O_CARRIER_ID
,O_ENTRY_D
FROM ORDERS
WHERE O_W_ID = ORD_C_ID.W_ID
AND O_D_ID = ORD_C_ID.D_ID
AND O_C_ID = ORD_C_ID.C_ID
ORDER BY O_ID DESC
FETCH FIRST 1 ROW ONLY
)
;
RETURN VALUES ( VAR.O_ID
,VAR.O_CARRIER_ID
,VAR.O_ENTRY_D
,VAR.C_BALANCE
,VAR.C_FIRST
,VAR.C_MIDDLE
,VAR.C_LAST
);

```

```

END $
--
-- PAYMENT
--
CREATE FUNCTION PAY_C_LAST( W_ID INTEGER
,D_ID SMALLINT
,C_W_ID INTEGER
,C_D_ID SMALLINT
,C_LAST VARCHAR(16)
,H_DATE BIGINT
,H_AMOUNT BIGINT
,BAD_CREDIT_PREFIX VARCHAR(28)
)
RETURNS TABLE( W_STREET_1 CHAR(20)
,W_STREET_2 CHAR(20)
,W_CITY CHAR(20)
,W_STATE CHAR(2)
,W_ZIP CHAR(9)
,D_STREET_1 CHAR(20)
,D_STREET_2 CHAR(20)
,D_CITY CHAR(20)
,D_STATE CHAR(2)
,D_ZIP CHAR(9)
,C_ID INTEGER
,C_FIRST VARCHAR(16)
,C_MIDDLE CHAR(2)
,C_STREET_1 VARCHAR(20)
,C_STREET_2 VARCHAR(20)
,C_CITY VARCHAR(20)
,C_STATE CHAR(2)
,C_ZIP CHAR(9)
,C_PHONE CHAR(16)
,C_SINCE BIGINT
,C_CREDIT CHAR(2)
,C_CREDIT_LIM BIGINT
,C_DISCOUNT INTEGER
,C_BALANCE BIGINT
,C_DATA CHAR(200)
)
SPECIFIC PAY_C_LAST
MODIFIES SQL DATA DETERMINISTIC NO EXTERNAL ACTION
LANGUAGE SQL
VAR: BEGIN ATOMIC
DECLARE W_NAME CHAR(10);
DECLARE D_NAME CHAR(10);
DECLARE W_STREET_1 CHAR(20);
DECLARE W_STREET_2 CHAR(20);
DECLARE W_CITY CHAR(20);
DECLARE W_STATE CHAR(2);
DECLARE W_ZIP CHAR(9);
DECLARE D_STREET_1 CHAR(20);
DECLARE D_STREET_2 CHAR(20);
DECLARE D_CITY CHAR(20);
DECLARE D_STATE CHAR(2);
DECLARE D_ZIP CHAR(9);
DECLARE C_ID INTEGER;
DECLARE C_FIRST VARCHAR(16);
DECLARE C_MIDDLE CHAR(2);
DECLARE C_STREET_1 VARCHAR(20);
DECLARE C_STREET_2 VARCHAR(20);
DECLARE C_CITY VARCHAR(20);
DECLARE C_STATE CHAR(2);
DECLARE C_ZIP CHAR(9);
DECLARE C_PHONE CHAR(16);
DECLARE C_SINCE BIGINT;
DECLARE C_CREDIT CHAR(2);
DECLARE C_CREDIT_LIM BIGINT;

```

```

DECLARE C_DISCOUNT INTEGER ;
DECLARE C_BALANCE BIGINT ;
DECLARE C_DATA CHAR(200) ;
/* Update District and retrieve its data */
SET ( D_NAME, D_STREET_1, D_STREET_2, D_CITY, D_STATE,
D_ZIP )
= ( SELECT D_NAME, D_STREET_1, D_STREET_2, D_CITY,
D_STATE, D_ZIP
FROM OLD TABLE ( UPDATE DISTRICT
SET D_YTD = D_YTD + PAY_C_LAST.H_AMOUNT
WHERE D_W_ID = PAY_C_LAST.W_ID
AND D_ID = PAY_C_LAST.D_ID
) AS U
)
;
/* Determine the C_ID */
SET ( C_ID )
= ( SELECT C_ID
FROM ( SELECT C_ID
, COUNT(*) OVER() AS COUNT
, ROWNUMBER() OVER (ORDER BY C_FIRST) AS NUM
FROM CUSTOMER
WHERE C_LAST = PAY_C_LAST.C_LAST
AND C_W_ID = PAY_C_LAST.C_W_ID
AND C_D_ID = PAY_C_LAST.C_D_ID
) AS T
WHERE NUM = (COUNT + BIGINT(1)) / BIGINT(2)
)
;
/* Update the middle customer */
SET ( C_ID, 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, C_DATA )
= ( SELECT C_ID, 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
, CASE WHEN C_CREDIT = 'BC' THEN SUBSTR(C_DATA, 1,
200) ELSE NULL END AS C_DATA
FROM NEW TABLE ( UPDATE CUSTOMER
SET C_BALANCE = C_BALANCE -
PAY_C_LAST.H_AMOUNT
, C_YTD_PAYMENT = C_YTD_PAYMENT +
PAY_C_LAST.H_AMOUNT
, C_PAYMENT_CNT = C_PAYMENT_CNT +
SMALLINT(1)
, C_DATA = CASE WHEN C_CREDIT = 'BC'
THEN CHAR(C_ID) -- 11 bytes long
||BAD_CREDIT_PREFIX -- 28 bytes
long
||SUBSTR(C_DATA, 1, 461) -- 461 + 39 =
500
ELSE C_DATA
END
WHERE C_W_ID = PAY_C_LAST.C_W_ID
AND C_D_ID = PAY_C_LAST.C_D_ID
AND C_ID = VAR.C_ID
) AS U
)
;
/* Update the warehouse */
SET ( W_NAME, W_STREET_1, W_STREET_2, W_CITY, W_STATE,
W_ZIP )
= ( SELECT W_NAME, W_STREET_1, W_STREET_2, W_CITY,
W_STATE, W_ZIP
FROM OLD TABLE ( UPDATE WAREHOUSE

```

```

SET W_YTD = W_YTD + PAY_C_LAST.H_AMOUNT
WHERE W_ID = PAY_C_LAST.W_ID
) AS U
)
;
/* Finally insert into the warehouse */
INSERT
INTO HISTORY ( H_C_ID, H_C_D_ID, H_C_W_ID, H_D_ID, H_W_ID,
H_DATA, H_DATE, H_AMOUNT )
VALUES ( VAR.C_ID
, PAY_C_LAST.C_D_ID
, PAY_C_LAST.C_W_ID
, PAY_C_LAST.D_ID
, PAY_C_LAST.W_ID
, VAR.W_NAME || CHAR(' ', 4) || VAR.D_NAME
, PAY_C_LAST.H_DATE
, PAY_C_LAST.H_AMOUNT
)
;
/* Done - return the collected data */
RETURN VALUES ( 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_ID, 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, C_DATA
)
;
END
$
CREATE FUNCTION PAY_C_ID( W_ID INTEGER
, D_ID SMALLINT
, C_W_ID INTEGER
, C_D_ID SMALLINT
, C_ID INTEGER
, H_DATE BIGINT
, H_AMOUNT BIGINT
, BAD_CREDIT_PREFIX VARCHAR(34)
)
RETURNS TABLE( W_STREET_1 CHAR(20)
, W_STREET_2 CHAR(20)
, W_CITY CHAR(20)
, W_STATE CHAR(2)
, W_ZIP CHAR(9)
, D_STREET_1 CHAR(20)
, D_STREET_2 CHAR(20)
, D_CITY CHAR(20)
, D_STATE CHAR(2)
, D_ZIP CHAR(9)
, C_LAST VARCHAR(16)
, C_FIRST VARCHAR(16)
, C_MIDDLE CHAR(2)
, C_STREET_1 VARCHAR(20)
, C_STREET_2 VARCHAR(20)
, C_CITY VARCHAR(20)
, C_STATE CHAR(2)
, C_ZIP CHAR(9)
, C_PHONE CHAR(16)
, C_SINCE BIGINT
, C_CREDIT CHAR(2)
, C_CREDIT_LIM BIGINT
, C_DISCOUNT INTEGER
, C_BALANCE BIGINT
, C_DATA CHAR(200)
)
SPECIFIC PAY_C_ID

```

```

MODIFIES SQL DATA DETERMINISTIC NO EXTERNAL ACTION
LANGUAGE SQL
VAR: BEGIN ATOMIC
DECLARE W_NAME CHAR(10);
DECLARE D_NAME CHAR(10);
DECLARE W_STREET_1 CHAR(20);
DECLARE W_STREET_2 CHAR(20);
DECLARE W_CITY CHAR(20);
DECLARE W_STATE CHAR(2);
DECLARE W_ZIP CHAR(9);
DECLARE D_STREET_1 CHAR(20);
DECLARE D_STREET_2 CHAR(20);
DECLARE D_CITY CHAR(20);
DECLARE D_STATE CHAR(2);
DECLARE D_ZIP CHAR(9);
DECLARE C_LAST VARCHAR(16);
DECLARE C_FIRST VARCHAR(16);
DECLARE C_MIDDLE CHAR(2);
DECLARE C_STREET_1 VARCHAR(20);
DECLARE C_STREET_2 VARCHAR(20);
DECLARE C_CITY VARCHAR(20);
DECLARE C_STATE CHAR(2);
DECLARE C_ZIP CHAR(9);
DECLARE C_PHONE CHAR(16);
DECLARE C_SINCE BIGINT;
DECLARE C_CREDIT CHAR(2);
DECLARE C_CREDIT_LIM BIGINT;
DECLARE C_DISCOUNT INTEGER;
DECLARE C_BALANCE BIGINT;
DECLARE C_DATA CHAR(200);

/* Update District and retrieve its data */
SET ( D_NAME, D_STREET_1, D_STREET_2, D_CITY, D_STATE, D_ZIP
)
= ( SELECT D_NAME, D_STREET_1, D_STREET_2, D_CITY,
D_STATE, D_ZIP
FROM OLD TABLE ( UPDATE DISTRICT
SET D_YTD = D_YTD + PAY_C_ID.H_AMOUNT
WHERE D_W_ID = PAY_C_ID.W_ID
AND D_ID = PAY_C_ID.D_ID
) AS U
)
;
/* Update the middle customer */
SET ( C_LAST, 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, C_DATA )
= ( SELECT C_LAST, 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
, CASE WHEN C_CREDIT = 'BC' THEN SUBSTR(C_DATA, 1,
200) ELSE NULL END AS C_DATA
FROM NEW TABLE ( UPDATE CUSTOMER
SET C_BALANCE = C_BALANCE -
PAY_C_ID.H_AMOUNT
, C_YTD_PAYMENT = C_YTD_PAYMENT +
PAY_C_ID.H_AMOUNT
, C_PAYMENT_CNT = C_PAYMENT_CNT +
SMALLINT( 1 )
, C_DATA = CASE WHEN C_CREDIT = 'BC'
THEN BAD_CREDIT_PREFIX -- 34
bytes long
SUBSTR( C_DATA, 1, 466 ) -- 466 + 34 =
500 bytes
ELSE C_DATA

```

```

END
WHERE C_W_ID = PAY_C_ID.C_W_ID
AND C_D_ID = PAY_C_ID.C_D_ID
AND C_ID = PAY_C_ID.C_ID
) AS U
)
;
/* Update the warehouse */
SET ( W_NAME, W_STREET_1, W_STREET_2, W_CITY, W_STATE,
W_ZIP )
= ( SELECT W_NAME, W_STREET_1, W_STREET_2, W_CITY,
W_STATE, W_ZIP
FROM OLD TABLE ( UPDATE WAREHOUSE
SET W_YTD = W_YTD + PAY_C_ID.H_AMOUNT
WHERE W_ID = PAY_C_ID.W_ID
) AS U
)
;
/* Finally insert into the warehouse */
INSERT
INTO HISTORY ( H_C_ID, H_C_D_ID, H_C_W_ID, H_D_ID, H_W_ID,
H_DATA, H_DATE, H_AMOUNT )
VALUES ( PAY_C_ID.C_ID
, PAY_C_ID.C_D_ID
, PAY_C_ID.C_W_ID
, PAY_C_ID.D_ID
, PAY_C_ID.W_ID
, VAR.W_NAME || CHAR( ' ', 4 ) || VAR.D_NAME
, PAY_C_ID.H_DATE
, PAY_C_ID.H_AMOUNT
)
;
/* Done - return the collected data */
RETURN VALUES ( 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_LAST, 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, C_DATA
)
;
END
$
--
-- NEW ORDER
--
CREATE FUNCTION NEW_OL_ALL( I_ID INT
, I_QTY SMALLINT
, W_ID INT
, SUPP_W_ID INT
, O_ID INT
, D_ID SMALLINT
)
RETURNS TABLE( I_PRICE INTEGER
, I_NAME CHAR(24)
, I_DATA VARCHAR(50)
, OL_DIST_INFO CHAR(24)
, S_DATA VARCHAR(50)
, S_QUANTITY SMALLINT
)
SPECIFIC NEW_OL_ALL
MODIFIES SQL DATA DETERMINISTIC NO EXTERNAL ACTION
LANGUAGE SQL
VAR: BEGIN ATOMIC
DECLARE I_PRICE INTEGER ;
DECLARE I_NAME CHAR(24) ;
DECLARE I_DATA VARCHAR(50) ;

```

```

DECLARE OL_DIST_INFO CHAR(24) ;
DECLARE S_DATA VARCHAR(50) ;
DECLARE S_QUANTITY SMALLINT ;

SET ( I_PRICE , I_NAME , I_DATA )
= ( SELECT
      I_PRICE
      , I_NAME
      , I_DATA
    FROM ITEM
    WHERE ITEM.I_ID = NEW_OL_ALL.I_ID
  );
SET ( OL_DIST_INFO , S_DATA , S_QUANTITY )
= ( SELECT OL_DIST_INFO
      , S_DATA
      , S_QUANTITY
    FROM NEW TABLE ( UPDATE STOCK
                      INCLUDE ( OL_DIST_INFO CHAR( 24 ) )
                      SET S_QUANTITY = CASE WHEN S_QUANTITY -
NEW_OL_ALL.I_QTY >= 10
NEW_OL_ALL.I_QTY
NEW_OL_ALL.I_QTY
NEW_OL_ALL.I_QTY + 91
END
, S_ORDER_CNT = S_ORDER_CNT +
SMALLINT( 1 )
, S_YTD = S_YTD + NEW_OL_ALL.I_QTY
, S_REMOTE_CNT = CASE WHEN
NEW_OL_ALL.SUPP_W_ID = NEW_OL_ALL.W_ID
THEN S_REMOTE_CNT
ELSE S_REMOTE_CNT +
SMALLINT( 1 )
END
, OL_DIST_INFO = CASE D_ID WHEN
SMALLINT( 1 ) THEN S_DIST_01
WHEN SMALLINT( 2 )
THEN S_DIST_02
WHEN SMALLINT( 3 )
THEN S_DIST_03
WHEN SMALLINT( 4 )
THEN S_DIST_04
WHEN SMALLINT( 5 )
THEN S_DIST_05
WHEN SMALLINT( 6 )
THEN S_DIST_06
WHEN SMALLINT( 7 )
THEN S_DIST_07
WHEN SMALLINT( 8 )
THEN S_DIST_08
WHEN SMALLINT( 9 )
THEN S_DIST_09
WHEN SMALLINT( 10 )
END
WHERE S_I_ID = NEW_OL_ALL.I_ID
AND S_W_ID = NEW_OL_ALL.SUPP_W_ID
) AS U
);
RETURN VALUES( VAR.I_PRICE
, VAR.I_NAME
, VAR.I_DATA
, VAR.OL_DIST_INFO
, VAR.S_DATA
, VAR.S_QUANTITY
);
END
$
CREATE FUNCTION NEW_OL_LOCAL( I_ID INT
, I_QTY SMALLINT
, W_ID INT
, O_ID INT
, D_ID SMALLINT
)
RETURNS TABLE( I_PRICE INTEGER
, I_NAME CHAR(24)
, I_DATA VARCHAR(50)
, OL_DIST_INFO CHAR(24)
, S_DATA VARCHAR(50)
, S_QUANTITY SMALLINT
)
SPECIFIC NEW_OL_LOCAL
MODIFIES SQL DATA DETERMINISTIC NO EXTERNAL ACTION
LANGUAGE SQL
VAR: BEGIN ATOMIC
DECLARE I_PRICE INTEGER ;
DECLARE I_NAME CHAR(24) ;
DECLARE I_DATA VARCHAR(50) ;
DECLARE OL_DIST_INFO CHAR(24) ;
DECLARE S_DATA VARCHAR(50) ;
DECLARE S_QUANTITY SMALLINT ;

SET ( I_PRICE , I_NAME , I_DATA )
= ( SELECT
      I_PRICE
      , I_NAME
      , I_DATA
    FROM ITEM
    WHERE ITEM.I_ID = NEW_OL_LOCAL.I_ID
  );
SET ( OL_DIST_INFO , S_DATA , S_QUANTITY )
= ( SELECT OL_DIST_INFO
      , S_DATA
      , S_QUANTITY
    FROM NEW TABLE ( UPDATE STOCK
                      INCLUDE ( OL_DIST_INFO CHAR( 24 ) )
                      SET S_QUANTITY = CASE WHEN S_QUANTITY -
NEW_OL_LOCAL.I_QTY >= 10
NEW_OL_LOCAL.I_QTY
NEW_OL_LOCAL.I_QTY
NEW_OL_LOCAL.I_QTY + 91
END
, S_ORDER_CNT = S_ORDER_CNT +
SMALLINT( 1 )
, S_YTD = S_YTD +
NEW_OL_LOCAL.I_QTY
, OL_DIST_INFO = CASE D_ID WHEN
SMALLINT( 1 ) THEN S_DIST_01
WHEN SMALLINT( 2 )
THEN S_DIST_02
WHEN SMALLINT( 3 )
THEN S_DIST_03
WHEN SMALLINT( 4 )
THEN S_DIST_04
WHEN SMALLINT( 5 )
THEN S_DIST_05
WHEN SMALLINT( 6 )
THEN S_DIST_06
WHEN SMALLINT( 7 )
THEN S_DIST_07
WHEN SMALLINT( 8 )

```

```

        WHEN SMALLINT( 9 )
THEN S_DIST_09
        WHEN SMALLINT( 10 )
THEN S_DIST_10
        END
        WHERE S_I_ID = NEW_OL_LOCAL_I_ID
        AND S_W_ID = NEW_OL_LOCAL.W_ID
    ) AS U
)
;
RETURN VALUES( VAR_I_PRICE
, VAR_I_NAME
, VAR_I_DATA
, VAR_OL_DIST_INFO
, VAR_S_DATA
, VAR_S_QUANTITY
)
;
END
$

```

```

CREATE FUNCTION NEW_WH( O_ID INTEGER
, W_ID INTEGER
, D_ID SMALLINT
, C_ID INTEGER
, O_ENTRY_D BIGINT
, O_OL_CNT SMALLINT
, O_ALL_LOCAL SMALLINT
)
RETURNS TABLE( W_TAX INTEGER
, C_DISCOUNT INTEGER
, C_LAST VARCHAR(16)
, C_CREDIT CHAR(2)
)
SPECIFIC NEW_WH
MODIFIES SQL DATA DETERMINISTIC NO EXTERNAL ACTION
LANGUAGE SQL
VAR: BEGIN ATOMIC
DECLARE C_DISCOUNT INTEGER ;
DECLARE C_LAST VARCHAR(16);
DECLARE C_CREDIT CHAR(2);
DECLARE W_TAX INTEGER;
INSERT
INTO NEW_ORDER ( NO_O_ID, NO_D_ID, NO_W_ID )
VALUES ( O_ID
, D_ID
, W_ID
)
;
INSERT
INTO ORDERS ( O_C_ID, O_ENTRY_D, O_CARRIER_ID, O_OL_CNT,
O_ALL_LOCAL, O_ID, O_W_ID, O_D_ID )
VALUES ( C_ID
, O_ENTRY_D
, 0
, O_OL_CNT
, O_ALL_LOCAL
, O_ID
, W_ID
, D_ID
)
;
SET ( C_DISCOUNT, C_LAST, C_CREDIT )
= ( SELECT C_DISCOUNT, C_LAST, C_CREDIT
FROM CUSTOMER
WHERE C_ID = NEW_WH.C_ID

```

```

AND C_W_ID = W_ID
AND C_D_ID = D_ID
)
;
SET W_TAX
= ( SELECT W_TAX
FROM WAREHOUSE
WHERE W_ID = NEW_WH.W_ID
)
;
RETURN VALUES ( W_TAX , C_DISCOUNT , C_LAST , C_CREDIT );
END
$

```

Src.Srv/cat-proc.ddl

```

CREATE PROCEDURE news
(in new_in varchar(270) FOR BIT DATA,
out new_out varchar(662) FOR BIT DATA)
LANGUAGE C
PARAMETER STYLE GENERAL
EXTERNAL NAME 'c:\SQLLIB\function\lrpctpcc!news'
not fenced;

CREATE PROCEDURE ords
(in ord_in varchar(42) FOR BIT DATA,
out ord_out varchar(446) FOR BIT DATA)
LANGUAGE C
PARAMETER STYLE GENERAL
EXTERNAL NAME 'c:\SQLLIB\function\lrpctpcc!ords'
not fenced;

CREATE PROCEDURE dels
(in del_in varchar(22) FOR BIT DATA,
out del_out varchar(50) FOR BIT DATA)
LANGUAGE C
PARAMETER STYLE GENERAL
EXTERNAL NAME 'c:\SQLLIB\function\lrpctpcc!dels'
not fenced;

```

Src.Srv/tpcc_all_sql.sqc

```

/*****
*****
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2004
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****
*****/
/*
* tpcc_all_sql.sqc - Client/Server code for TPCC
*/
#include <stdlib.h>
#include <errno.h>
#include "db2tpcc.h"
#include "tpccapp.h"
#include "tpccdbg.h"
#include "sqlca.h"
#include "sql.h"

```



```

#include "Ival.h"
// -----
// New Order SERVER
// -----
int static is_ORIGINAL( char *string, short length );
SQL_API_RC new_order_internal( char *pin, char *pout )
{
    struct out_neword_struct *neword;
    struct in_neword_struct *in_neword;
    struct sqlca sqlca ;
    int fbadItemDetected = 0 ;
    EXEC SQL BEGIN DECLARE SECTION;
    char c_last [ 16 ] ;
    char c_credit [ 2 ] ;
    sqlint32 c_discount ;
    sqlint32 dist_tax ;
    sqlint32 ware_tax ;
    sqlint32 w_id ;
    short d_id ;
    sqlint32 c_id ;
    sqlint32 next_o_id ;
    short s_quantity ;
    sqlint32 supply_w_id ;
    short inputItemCnt ;
    char stockDistrictInformation [ 24 ] ;
    char item_name[ 24 ] ;
    sqlint64 o_entry_d ;

    short allLocal ;

    sqlint32 item_price ;

    struct i_data_type { short len ; char data[ 50 ] ; } i_data ;
    struct s_data_type { short len ; char data[ 50 ] ; } s_data ;
    sqlint32 id0, id1, id2, id3, id4, id5, id6, id7 ;
    sqlint32 id8, id9, id10, id11, id12, id13, id14 ;
    sqlint32 supply_w_id0, supply_w_id1, supply_w_id2, supply_w_id3 ;
    sqlint32 supply_w_id4, supply_w_id5, supply_w_id6, supply_w_id7 ;
    sqlint32 supply_w_id8, supply_w_id9, supply_w_id10, supply_w_id11 ;
    sqlint32 supply_w_id12, supply_w_id13, supply_w_id14 ;
    short ol_quantity0, ol_quantity1, ol_quantity2, ol_quantity3 ;
    short ol_quantity4, ol_quantity5, ol_quantity6, ol_quantity7 ;
    short ol_quantity8, ol_quantity9, ol_quantity10, ol_quantity11 ;
    short ol_quantity12, ol_quantity13, ol_quantity14 ;

    EXEC SQL END DECLARE SECTION;
    int storedProcRc ;
    int inputItemArrayIndex ;
    char stockDistrictInformationArray [15][25];
    #define stockDistrictInformationArray[
inputItemArrayIndex ]
    // Redirected input fields
    #define w_id in_neword->s_W_ID
    #define d_id in_neword->s_D_ID
    #define c_id in_neword->s_C_ID
    #define o_entry_d in_neword->s_O_ENTRY_D_time
    #define inputItemCnt in_neword->s_O_OL_CNT
    #define allLocal in_neword->s_all_local
    // Redirected output fields
    #define c_last neword->s_C_LAST
    #define c_credit neword->s_C_CREDIT
    #define c_discount neword->s_C_DISCOUNT
    #define ware_tax neword->s_W_TAX
    #define dist_tax neword->s_D_TAX
    #define s_quantity neword->item[ inputItemArrayIndex ].s_S_QUANTITY
    // This output field becomes an input field to order_line

```

```

#define next_o_id neword->s_O_ID

#define item_name neword->item[ inputItemArrayIndex ].s_I_NAME
// item_price holds the integer version of this value. If the return structure was
// an integer this would not be necessary.
sqlint32 i_priceArray[ 15 ] ;
#define item_price i_priceArray[ inputItemArrayIndex ]
// Handle the generic/brand distinction
struct i_data_type i_dataArray[ 15 ] ;
struct s_data_type s_dataArray[ 15 ] ;
#define i_data i_dataArray[ inputItemArrayIndex ]
#define s_data s_dataArray[ inputItemArrayIndex ]

// Redirect hostvars to input structure
#define id0 in_neword->in_item[0].s_OL_I_ID
#define id1 in_neword->in_item[1].s_OL_I_ID
#define id2 in_neword->in_item[2].s_OL_I_ID
#define id3 in_neword->in_item[3].s_OL_I_ID
#define id4 in_neword->in_item[4].s_OL_I_ID
#define id5 in_neword->in_item[5].s_OL_I_ID
#define id6 in_neword->in_item[6].s_OL_I_ID
#define id7 in_neword->in_item[7].s_OL_I_ID
#define id8 in_neword->in_item[8].s_OL_I_ID
#define id9 in_neword->in_item[9].s_OL_I_ID
#define id10 in_neword->in_item[10].s_OL_I_ID
#define id11 in_neword->in_item[11].s_OL_I_ID
#define id12 in_neword->in_item[12].s_OL_I_ID
#define id13 in_neword->in_item[13].s_OL_I_ID
#define id14 in_neword->in_item[14].s_OL_I_ID
#define ol_quantity0 in_neword->in_item[ 0 ].s_OL_QUANTITY
#define ol_quantity1 in_neword->in_item[ 1 ].s_OL_QUANTITY
#define ol_quantity2 in_neword->in_item[ 2 ].s_OL_QUANTITY
#define ol_quantity3 in_neword->in_item[ 3 ].s_OL_QUANTITY
#define ol_quantity4 in_neword->in_item[ 4 ].s_OL_QUANTITY
#define ol_quantity5 in_neword->in_item[ 5 ].s_OL_QUANTITY
#define ol_quantity6 in_neword->in_item[ 6 ].s_OL_QUANTITY
#define ol_quantity7 in_neword->in_item[ 7 ].s_OL_QUANTITY
#define ol_quantity8 in_neword->in_item[ 8 ].s_OL_QUANTITY
#define ol_quantity9 in_neword->in_item[ 9 ].s_OL_QUANTITY
#define ol_quantity10 in_neword->in_item[ 10 ].s_OL_QUANTITY
#define ol_quantity11 in_neword->in_item[ 11 ].s_OL_QUANTITY
#define ol_quantity12 in_neword->in_item[ 12 ].s_OL_QUANTITY
#define ol_quantity13 in_neword->in_item[ 13 ].s_OL_QUANTITY
#define ol_quantity14 in_neword->in_item[ 14 ].s_OL_QUANTITY
#define supply_w_id0 in_neword->in_item[ 0 ].s_OL_SUPPLY_W_ID
#define supply_w_id1 in_neword->in_item[ 1 ].s_OL_SUPPLY_W_ID
#define supply_w_id2 in_neword->in_item[ 2 ].s_OL_SUPPLY_W_ID
#define supply_w_id3 in_neword->in_item[ 3 ].s_OL_SUPPLY_W_ID
#define supply_w_id4 in_neword->in_item[ 4 ].s_OL_SUPPLY_W_ID
#define supply_w_id5 in_neword->in_item[ 5 ].s_OL_SUPPLY_W_ID
#define supply_w_id6 in_neword->in_item[ 6 ].s_OL_SUPPLY_W_ID
#define supply_w_id7 in_neword->in_item[ 7 ].s_OL_SUPPLY_W_ID
#define supply_w_id8 in_neword->in_item[ 8 ].s_OL_SUPPLY_W_ID
#define supply_w_id9 in_neword->in_item[ 9 ].s_OL_SUPPLY_W_ID
#define supply_w_id10 in_neword->in_item[ 10 ].s_OL_SUPPLY_W_ID
#define supply_w_id11 in_neword->in_item[ 11 ].s_OL_SUPPLY_W_ID
#define supply_w_id12 in_neword->in_item[ 12 ].s_OL_SUPPLY_W_ID
#define supply_w_id13 in_neword->in_item[ 13 ].s_OL_SUPPLY_W_ID
#define supply_w_id14 in_neword->in_item[ 14 ].s_OL_SUPPLY_W_ID
EXEC SQL DECLARE ISOL_Remote_1 CURSOR FOR
    WITH DATA AS ( SELECT O_ID
        , D_ID
        , W_ID
        , OL_NUMBER
        , I_ID
        , I_SUPPLY_W_ID
        , 0 AS OL_DELIVERY_D

```

```

,I_QTY
,(I_PRICE * I_QTY) AS TOTAL_PRICE
,OL_DIST_INFO
,I_PRICE,I_NAME,I_DATA,S_DATA,S_QUANTITY
FROM ( SELECT :next_o_id as O_ID
      ,:w_id AS W_ID
      ,:d_id as D_ID
      ,OL_NUMBER
      ,I_ID
      ,I_SUPPLY_W_ID
      ,I_QTY
FROM Table( VALUES
      ( SMALLINT(1) ,:id0 ,:ol_quantity0
, :supply_w_id0 )
      ) AS X (OL_NUMBER , I_ID , I_QTY
) AS ITEMLIST
, TABLE(NEW_OL_ALL( I_ID
      ,I_QTY
      ,W_ID
      ,I_SUPPLY_W_ID
      ,O_ID
      ,D_ID
      )
) AS NEW_OL_ALL
WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
)
SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA ,
S_QUANTITY
FROM NEW TABLE ( INSERT INTO ORDER_LINE
( OL_O_ID
, OL_D_ID
, OL_W_ID
, OL_NUMBER
, OL_I_ID
, OL_SUPPLY_W_ID
, OL_DELIVERY_D
, OL_QUANTITY
, OL_AMOUNT
, OL_DIST_INFO
)
INCLUDE( I_PRICE INTEGER
, I_NAME CHAR(24)
, I_DATA VARCHAR(50)
, S_DATA VARCHAR(50)
, S_QUANTITY SMALLINT )
SELECT O_ID
, D_ID
, W_ID
, OL_NUMBER
, I_ID
, I_SUPPLY_W_ID
, OL_DELIVERY_D
, I_QTY
, TOTAL_PRICE
, OL_DIST_INFO
, I_PRICE, I_NAME, I_DATA, S_DATA,
S_QUANTITY
FROM DATA
) AS INS
;
EXEC SQL DECLARE ISOL_Remote_2 CURSOR FOR
WITH DATA AS ( SELECT O_ID
, D_ID
, W_ID
, OL_NUMBER

```

```

,I_ID
,I_SUPPLY_W_ID
,0 AS OL_DELIVERY_D
,I_QTY
,(I_PRICE * I_QTY) AS TOTAL_PRICE
,OL_DIST_INFO
,I_PRICE,I_NAME,I_DATA,S_DATA,S_QUANTITY
FROM ( SELECT :next_o_id as O_ID
      ,:w_id AS W_ID
      ,:d_id as D_ID
      ,OL_NUMBER
      ,I_ID
      ,I_SUPPLY_W_ID
      ,I_QTY
FROM Table( VALUES
      ( SMALLINT(1) ,:id0 ,:ol_quantity0
, :supply_w_id0 )
      , ( SMALLINT(2) ,:id1 ,:ol_quantity1
) AS X (OL_NUMBER , I_ID , I_QTY
) AS ITEMLIST
, TABLE(NEW_OL_ALL( I_ID
      ,I_QTY
      ,W_ID
      ,I_SUPPLY_W_ID
      ,O_ID
      ,D_ID
      )
) AS NEW_OL_ALL
WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
)
SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA ,
S_QUANTITY
FROM NEW TABLE ( INSERT INTO ORDER_LINE
( OL_O_ID
, OL_D_ID
, OL_W_ID
, OL_NUMBER
, OL_I_ID
, OL_SUPPLY_W_ID
, OL_DELIVERY_D
, OL_QUANTITY
, OL_AMOUNT
, OL_DIST_INFO
)
INCLUDE( I_PRICE INTEGER
, I_NAME CHAR(24)
, I_DATA VARCHAR(50)
, S_DATA VARCHAR(50)
, S_QUANTITY SMALLINT )
SELECT O_ID
, D_ID
, W_ID
, OL_NUMBER
, I_ID
, I_SUPPLY_W_ID
, OL_DELIVERY_D
, I_QTY
, TOTAL_PRICE
, OL_DIST_INFO
, I_PRICE, I_NAME, I_DATA, S_DATA,
S_QUANTITY
FROM DATA
) AS INS
;
EXEC SQL DECLARE ISOL_Remote_3 CURSOR FOR

```

```

WITH DATA AS ( SELECT O_ID
, D_ID
, W_ID
, OL_NUMBER
, I_ID
, I_SUPPLY_W_ID
, 0 AS OL_DELIVERY_D
, I_QTY
, (I_PRICE * I_QTY) AS TOTAL_PRICE
, OL_DIST_INFO
, I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY
FROM ( SELECT :next_o_id as O_ID
, :w_id AS W_ID
, :d_id as D_ID
, OL_NUMBER
, I_ID
, I_SUPPLY_W_ID
, I_QTY
FROM Table( VALUES
( SMALLINT(1) , :id0 , :ol_quantity0
, :supply_w_id0 )
, ( SMALLINT(2) , :id1 , :ol_quantity1
, :supply_w_id1 )
, ( SMALLINT(3) , :id2 , :ol_quantity2
, :supply_w_id2 )
) AS X (OL_NUMBER, I_ID, I_QTY
) AS ITEMLIST
, TABLE(NEW_OL_ALL( I_ID
, I_QTY
, W_ID
, I_SUPPLY_W_ID
, O_ID
, D_ID
)
) AS NEW_OL_ALL
WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
)
SELECT I_PRICE, I_NAME, I_DATA, OL_DIST_INFO, S_DATA,
S_QUANTITY
FROM NEW TABLE ( INSERT INTO ORDER_LINE
( OL_O_ID
, OL_D_ID
, OL_W_ID
, OL_NUMBER
, OL_I_ID
, OL_SUPPLY_W_ID
, OL_DELIVERY_D
, OL_QUANTITY
, OL_AMOUNT
, OL_DIST_INFO
)
INCLUDE ( I_PRICE INTEGER
, I_NAME CHAR(24)
, I_DATA VARCHAR(50)
, S_DATA VARCHAR(50)
, S_QUANTITY SMALLINT )
SELECT O_ID
, D_ID
, W_ID
, OL_NUMBER
, I_ID
, I_SUPPLY_W_ID
, OL_DELIVERY_D
, I_QTY
, TOTAL_PRICE
, OL_DIST_INFO

```

```

, I_PRICE, I_NAME, I_DATA, S_DATA,
S_QUANTITY
FROM DATA
) AS INS
;
EXEC SQL DECLARE ISOL_Remote_4 CURSOR FOR
WITH DATA AS ( SELECT O_ID
, D_ID
, W_ID
, OL_NUMBER
, I_ID
, I_SUPPLY_W_ID
, 0 AS OL_DELIVERY_D
, I_QTY
, (I_PRICE * I_QTY) AS TOTAL_PRICE
, OL_DIST_INFO
, I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY
FROM ( SELECT :next_o_id as O_ID
, :w_id AS W_ID
, :d_id as D_ID
, OL_NUMBER
, I_ID
, I_SUPPLY_W_ID
, I_QTY
FROM Table( VALUES
( SMALLINT(1) , :id0 , :ol_quantity0
, :supply_w_id0 )
, ( SMALLINT(2) , :id1 , :ol_quantity1
, :supply_w_id1 )
, ( SMALLINT(3) , :id2 , :ol_quantity2
, :supply_w_id2 )
, ( SMALLINT(4) , :id3 , :ol_quantity3
, :supply_w_id3 )
) AS X (OL_NUMBER, I_ID, I_QTY
) AS ITEMLIST
, TABLE(NEW_OL_ALL( I_ID
, I_QTY
, W_ID
, I_SUPPLY_W_ID
, O_ID
, D_ID
)
) AS NEW_OL_ALL
WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
)
SELECT I_PRICE, I_NAME, I_DATA, OL_DIST_INFO, S_DATA,
S_QUANTITY
FROM NEW TABLE ( INSERT INTO ORDER_LINE
( OL_O_ID
, OL_D_ID
, OL_W_ID
, OL_NUMBER
, OL_I_ID
, OL_SUPPLY_W_ID
, OL_DELIVERY_D
, OL_QUANTITY
, OL_AMOUNT
, OL_DIST_INFO
)
INCLUDE ( I_PRICE INTEGER
, I_NAME CHAR(24)
, I_DATA VARCHAR(50)
, S_DATA VARCHAR(50)
, S_QUANTITY SMALLINT )
SELECT O_ID
, D_ID

```

```

        ,W_ID
        ,OL_NUMBER
        ,I_ID
        ,I_SUPPLY_W_ID
        ,OL_DELIVERY_D
        ,I_QTY
        ,TOTAL_PRICE
        ,OL_DIST_INFO
        ,I_PRICE, I_NAME, I_DATA, S_DATA,
S_QUANTITY
        FROM DATA
        ) AS INS
;
EXEC SQL DECLARE ISOL_Remote_5 CURSOR FOR
WITH DATA AS ( SELECT O_ID
        ,D_ID
        ,W_ID
        ,OL_NUMBER
        ,I_ID
        ,I_SUPPLY_W_ID
        ,O AS OL_DELIVERY_D
        ,I_QTY
        ,(I_PRICE * I_QTY) AS TOTAL_PRICE
        ,OL_DIST_INFO
        ,I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY
        FROM ( SELECT :next_o_id as O_ID
                ,:w_id AS W_ID
                ,:d_id as D_ID
                ,OL_NUMBER
                ,I_ID
                ,I_SUPPLY_W_ID
                ,I_QTY
                FROM Table( VALUES
                        ( SMALLINT(1) ,:id0 ,:ol_quantity0
, :supply_w_id0 )
                        , ( SMALLINT(2) ,:id1 ,:ol_quantity1
, :supply_w_id1 )
                        , ( SMALLINT(3) ,:id2 ,:ol_quantity2
, :supply_w_id2 )
                        , ( SMALLINT(4) ,:id3 ,:ol_quantity3
, :supply_w_id3 )
                        , ( SMALLINT(5) ,:id4 ,:ol_quantity4
, :supply_w_id4 )
                ) AS X (OL_NUMBER , I_ID , I_QTY
        ) AS ITEMLIST
        , TABLE(NEW_OL_ALL( I_ID
                ,I_QTY
                ,W_ID
                ,I_SUPPLY_W_ID
                ,O_ID
                ,D_ID
                )
        ) AS NEW_OL_ALL
        WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
        )
        SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA ,
S_QUANTITY
        FROM NEW TABLE ( INSERT INTO ORDER_LINE
        ( OL_O_ID
        ,OL_D_ID
        ,OL_W_ID
        ,OL_NUMBER
        ,OL_I_ID
        ,OL_SUPPLY_W_ID
        ,OL_DELIVERY_D
        ,OL_QUANTITY

```

```

        ,OL_AMOUNT
        ,OL_DIST_INFO
        )
        INCLUDE( I_PRICE INTEGER
        ,I_NAME CHAR(24)
        ,I_DATA VARCHAR(50)
        ,S_DATA VARCHAR(50)
        ,S_QUANTITY SMALLINT)
        SELECT O_ID
        ,D_ID
        ,W_ID
        ,OL_NUMBER
        ,I_ID
        ,I_SUPPLY_W_ID
        ,OL_DELIVERY_D
        ,I_QTY
        ,TOTAL_PRICE
        ,OL_DIST_INFO
        ,I_PRICE, I_NAME, I_DATA, S_DATA,
S_QUANTITY
        FROM DATA
        ) AS INS
;
EXEC SQL DECLARE ISOL_Remote_6 CURSOR FOR
WITH DATA AS ( SELECT O_ID
        ,D_ID
        ,W_ID
        ,OL_NUMBER
        ,I_ID
        ,I_SUPPLY_W_ID
        ,O AS OL_DELIVERY_D
        ,I_QTY
        ,(I_PRICE * I_QTY) AS TOTAL_PRICE
        ,OL_DIST_INFO
        ,I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY
        FROM ( SELECT :next_o_id as O_ID
                ,:w_id AS W_ID
                ,:d_id as D_ID
                ,OL_NUMBER
                ,I_ID
                ,I_SUPPLY_W_ID
                ,I_QTY
                FROM Table( VALUES
                        ( SMALLINT(1) ,:id0 ,:ol_quantity0
, :supply_w_id0 )
                        , ( SMALLINT(2) ,:id1 ,:ol_quantity1
, :supply_w_id1 )
                        , ( SMALLINT(3) ,:id2 ,:ol_quantity2
, :supply_w_id2 )
                        , ( SMALLINT(4) ,:id3 ,:ol_quantity3
, :supply_w_id3 )
                        , ( SMALLINT(5) ,:id4 ,:ol_quantity4
, :supply_w_id4 )
                        , ( SMALLINT(6) ,:id5 ,:ol_quantity5
, :supply_w_id5 )
                ) AS X (OL_NUMBER , I_ID , I_QTY
        ) AS ITEMLIST
        , TABLE(NEW_OL_ALL( I_ID
                ,I_QTY
                ,W_ID
                ,I_SUPPLY_W_ID
                ,O_ID
                ,D_ID
                )
        ) AS NEW_OL_ALL
        WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
        )

```

```

SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA ,
S_QUANTITY
FROM NEW TABLE ( INSERT INTO ORDER_LINE
( OL_O_ID
, OL_D_ID
, OL_W_ID
, OL_NUMBER
, OL_I_ID
, OL_SUPPLY_W_ID
, OL_DELIVERY_D
, OL_QUANTITY
, OL_AMOUNT
, OL_DIST_INFO
)
INCLUDE ( I_PRICE INTEGER
, I_NAME CHAR(24)
, I_DATA VARCHAR(50)
, S_DATA VARCHAR(50)
, S_QUANTITY SMALLINT )
SELECT O_ID
, D_ID
, W_ID
, OL_NUMBER
, I_ID
, I_SUPPLY_W_ID
, OL_DELIVERY_D
, I_QTY
, TOTAL_PRICE
, OL_DIST_INFO
, I_PRICE, I_NAME, I_DATA, S_DATA,
S_QUANTITY
FROM DATA
) AS INS
;
EXEC SQL DECLARE ISOL_Remote_7 CURSOR FOR
WITH DATA AS ( SELECT O_ID
, D_ID
, W_ID
, OL_NUMBER
, I_ID
, I_SUPPLY_W_ID
, 0 AS OL_DELIVERY_D
, I_QTY
, (I_PRICE * I_QTY) AS TOTAL_PRICE
, OL_DIST_INFO
, I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY
FROM ( SELECT :next_o_id as O_ID
, :w_id AS W_ID
, :d_id as D_ID
, OL_NUMBER
, I_ID
, I_SUPPLY_W_ID
, I_QTY
FROM Table( VALUES
( SMALLINT(1) , :id0 , :ol_quantity0
, :supply_w_id0 )
, ( SMALLINT(2) , :id1 , :ol_quantity1
, :supply_w_id1 )
, ( SMALLINT(3) , :id2 , :ol_quantity2
, :supply_w_id2 )
, ( SMALLINT(4) , :id3 , :ol_quantity3
, :supply_w_id3 )
, ( SMALLINT(5) , :id4 , :ol_quantity4
, :supply_w_id4 )
, ( SMALLINT(6) , :id5 , :ol_quantity5

```

```

, ( SMALLINT(7) , :id6 , :ol_quantity6
, :supply_w_id6 )
) AS X (OL_NUMBER , I_ID , I_QTY
, I_SUPPLY_W_ID )
) AS ITEMLIST
, TABLE(NEW_OL_ALL( I_ID
, I_QTY
, W_ID
, I_SUPPLY_W_ID
, O_ID
, D_ID
)
) AS NEW_OL_ALL
WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
)
SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA ,
S_QUANTITY
FROM NEW TABLE ( INSERT INTO ORDER_LINE
( OL_O_ID
, OL_D_ID
, OL_W_ID
, OL_NUMBER
, OL_I_ID
, OL_SUPPLY_W_ID
, OL_DELIVERY_D
, OL_QUANTITY
, OL_AMOUNT
, OL_DIST_INFO
)
INCLUDE ( I_PRICE INTEGER
, I_NAME CHAR(24)
, I_DATA VARCHAR(50)
, S_DATA VARCHAR(50)
, S_QUANTITY SMALLINT )
SELECT O_ID
, D_ID
, W_ID
, OL_NUMBER
, I_ID
, I_SUPPLY_W_ID
, OL_DELIVERY_D
, I_QTY
, TOTAL_PRICE
, OL_DIST_INFO
, I_PRICE, I_NAME, I_DATA, S_DATA,
S_QUANTITY
FROM DATA
) AS INS
;
EXEC SQL DECLARE ISOL_Remote_8 CURSOR FOR
WITH DATA AS ( SELECT O_ID
, D_ID
, W_ID
, OL_NUMBER
, I_ID
, I_SUPPLY_W_ID
, 0 AS OL_DELIVERY_D
, I_QTY
, (I_PRICE * I_QTY) AS TOTAL_PRICE
, OL_DIST_INFO
, I_PRICE, I_NAME, I_DATA, S_DATA,
S_QUANTITY
FROM DATA
) AS INS
;
EXEC SQL DECLARE ISOL_Remote_8 CURSOR FOR
WITH DATA AS ( SELECT O_ID
, D_ID
, W_ID
, OL_NUMBER
, I_ID
, I_SUPPLY_W_ID
, 0 AS OL_DELIVERY_D
, I_QTY
, (I_PRICE * I_QTY) AS TOTAL_PRICE
, OL_DIST_INFO
, I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY
FROM ( SELECT :next_o_id as O_ID
, :w_id AS W_ID
, :d_id as D_ID
, OL_NUMBER

```

```

        ,I_ID
        ,I_SUPPLY_W_ID
        ,I_QTY
    FROM Table( VALUES
        ( SMALLINT(1)      ,:id0 ,:ol_quantity0
, :supply_w_id0 )
        , ( SMALLINT(2)      ,:id1 ,:ol_quantity1
, :supply_w_id1 )
        , ( SMALLINT(3)      ,:id2 ,:ol_quantity2
, :supply_w_id2 )
        , ( SMALLINT(4)      ,:id3 ,:ol_quantity3
, :supply_w_id3 )
        , ( SMALLINT(5)      ,:id4 ,:ol_quantity4
, :supply_w_id4 )
        , ( SMALLINT(6)      ,:id5 ,:ol_quantity5
, :supply_w_id5 )
        , ( SMALLINT(7)      ,:id6 ,:ol_quantity6
, :supply_w_id6 )
        , ( SMALLINT(8)      ,:id7 ,:ol_quantity7
, :supply_w_id7 )
        ) AS X (OL_NUMBER , I_ID , I_QTY
, I_SUPPLY_W_ID )
    ) AS ITEM LIST
    , TABLE(NEW_OL_ALL( I_ID
        , I_QTY
        , W_ID
        , I_SUPPLY_W_ID
        , O_ID
        , D_ID
        )
    ) AS NEW_OL_ALL
    WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
)

SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA ,
S_QUANTITY
FROM NEW TABLE ( INSERT INTO ORDER_LINE
    ( OL_O_ID
    , OL_D_ID
    , OL_W_ID
    , OL_NUMBER
    , OL_I_ID
    , OL_SUPPLY_W_ID
    , OL_DELIVERY_D
    , OL_QUANTITY
    , OL_AMOUNT
    , OL_DIST_INFO
    )
    INCLUDE ( I_PRICE INTEGER
    , I_NAME CHAR(24)
    , I_DATA VARCHAR(50)
    , S_DATA VARCHAR(50)
    , S_QUANTITY SMALLINT )
    SELECT O_ID
    , D_ID
    , W_ID
    , OL_NUMBER
    , I_ID
    , I_SUPPLY_W_ID
    , OL_DELIVERY_D
    , I_QTY
    , TOTAL_PRICE
    , OL_DIST_INFO
    , I_PRICE, I_NAME, I_DATA, S_DATA,
S_QUANTITY
    FROM DATA
    ) AS INS
;

```

```

EXEC SQL DECLARE ISOL_Remote_9 CURSOR FOR
WITH DATA AS ( SELECT O_ID
    , D_ID
    , W_ID
    , OL_NUMBER
    , I_ID
    , I_SUPPLY_W_ID
    , 0 AS OL_DELIVERY_D
    , I_QTY
    , (I_PRICE * I_QTY) AS TOTAL_PRICE
    , OL_DIST_INFO
    , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY
    FROM ( SELECT :next_o_id as O_ID
        , :w_id AS W_ID
        , :d_id as D_ID
        , OL_NUMBER
        , I_ID
        , I_SUPPLY_W_ID
        , I_QTY
    FROM Table( VALUES
        ( SMALLINT(1)      ,:id0 ,:ol_quantity0
, :supply_w_id0 )
        , ( SMALLINT(2)      ,:id1 ,:ol_quantity1
, :supply_w_id1 )
        , ( SMALLINT(3)      ,:id2 ,:ol_quantity2
, :supply_w_id2 )
        , ( SMALLINT(4)      ,:id3 ,:ol_quantity3
, :supply_w_id3 )
        , ( SMALLINT(5)      ,:id4 ,:ol_quantity4
, :supply_w_id4 )
        , ( SMALLINT(6)      ,:id5 ,:ol_quantity5
, :supply_w_id5 )
        , ( SMALLINT(7)      ,:id6 ,:ol_quantity6
, :supply_w_id6 )
        , ( SMALLINT(8)      ,:id7 ,:ol_quantity7
, :supply_w_id7 )
        , ( SMALLINT(9)      ,:id8 ,:ol_quantity8
, :supply_w_id8 )
        ) AS X (OL_NUMBER , I_ID , I_QTY
, I_SUPPLY_W_ID )
    ) AS ITEM LIST
    , TABLE(NEW_OL_ALL( I_ID
        , I_QTY
        , W_ID
        , I_SUPPLY_W_ID
        , O_ID
        , D_ID
        )
    ) AS NEW_OL_ALL
    WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
)

SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA ,
S_QUANTITY
FROM NEW TABLE ( INSERT INTO ORDER_LINE
    ( OL_O_ID
    , OL_D_ID
    , OL_W_ID
    , OL_NUMBER
    , OL_I_ID
    , OL_SUPPLY_W_ID
    , OL_DELIVERY_D
    , OL_QUANTITY
    , OL_AMOUNT
    , OL_DIST_INFO
    )
    INCLUDE ( I_PRICE INTEGER
    , I_NAME CHAR(24)

```

```

, I_DATA VARCHAR(50)
, S_DATA VARCHAR(50)
, S_QUANTITY SMALLINT )
SELECT O_ID
, D_ID
, W_ID
, OL_NUMBER
, I_ID
, I_SUPPLY_W_ID
, OL_DELIVERY_D
, I_QTY
, TOTAL_PRICE
, OL_DIST_INFO
, I_PRICE, I_NAME, I_DATA, S_DATA,
S_QUANTITY
FROM DATA
) AS INS
;
EXEC SQL DECLARE ISOL_Remote_10 CURSOR FOR
WITH DATA AS ( SELECT O_ID
, D_ID
, W_ID
, OL_NUMBER
, I_ID
, I_SUPPLY_W_ID
, 0 AS OL_DELIVERY_D
, I_QTY
, (I_PRICE * I_QTY) AS TOTAL_PRICE
, OL_DIST_INFO
, I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY
FROM ( SELECT :next_o_id as O_ID
, :w_id AS W_ID
, :d_id as D_ID
, OL_NUMBER
, I_ID
, I_SUPPLY_W_ID
, I_QTY
FROM Table( VALUES
( SMALLINT( 1) , :id0 , :ol_quantity0
, :supply_w_id0 )
, ( SMALLINT( 2) , :id1 , :ol_quantity1
, :supply_w_id1 )
, ( SMALLINT( 3) , :id2 , :ol_quantity2
, :supply_w_id2 )
, ( SMALLINT( 4) , :id3 , :ol_quantity3
, :supply_w_id3 )
, ( SMALLINT( 5) , :id4 , :ol_quantity4
, :supply_w_id4 )
, ( SMALLINT( 6) , :id5 , :ol_quantity5
, :supply_w_id5 )
, ( SMALLINT( 7) , :id6 , :ol_quantity6
, :supply_w_id6 )
, ( SMALLINT( 8) , :id7 , :ol_quantity7
, :supply_w_id7 )
, ( SMALLINT( 9) , :id8 , :ol_quantity8
, :supply_w_id8 )
, ( SMALLINT( 10) , :id9 ,
:ol_quantity9 , :supply_w_id9 )
) AS X ( OL_NUMBER , I_ID , I_QTY
, I_SUPPLY_W_ID )
) AS ITEMLIST
, TABLE(NEW_OL_ALL( I_ID
, I_QTY
, W_ID
, I_SUPPLY_W_ID
, O_ID
, D_ID
)

```

```

) AS NEW_OL_ALL
WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
)
SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA ,
S_QUANTITY
FROM NEW TABLE ( INSERT INTO ORDER_LINE
( OL_O_ID
, OL_D_ID
, OL_W_ID
, OL_NUMBER
, OL_I_ID
, OL_SUPPLY_W_ID
, OL_DELIVERY_D
, OL_QUANTITY
, OL_AMOUNT
, OL_DIST_INFO
)
INCLUDE ( I_PRICE INTEGER
, I_NAME CHAR(24)
, I_DATA VARCHAR(50)
, S_DATA VARCHAR(50)
, S_QUANTITY SMALLINT )
SELECT O_ID
, D_ID
, W_ID
, OL_NUMBER
, I_ID
, I_SUPPLY_W_ID
, OL_DELIVERY_D
, I_QTY
, TOTAL_PRICE
, OL_DIST_INFO
, I_PRICE, I_NAME, I_DATA, S_DATA,
S_QUANTITY
FROM DATA
) AS INS
;
EXEC SQL DECLARE ISOL_Remote_11 CURSOR FOR
WITH DATA AS ( SELECT O_ID
, D_ID
, W_ID
, OL_NUMBER
, I_ID
, I_SUPPLY_W_ID
, 0 AS OL_DELIVERY_D
, I_QTY
, (I_PRICE * I_QTY) AS TOTAL_PRICE
, OL_DIST_INFO
, I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY
FROM ( SELECT :next_o_id as O_ID
, :w_id AS W_ID
, :d_id as D_ID
, OL_NUMBER
, I_ID
, I_SUPPLY_W_ID
, I_QTY
FROM Table( VALUES
( SMALLINT( 1) , :id0 , :ol_quantity0
, :supply_w_id0 )
, ( SMALLINT( 2) , :id1 , :ol_quantity1
, :supply_w_id1 )
, ( SMALLINT( 3) , :id2 , :ol_quantity2
, :supply_w_id2 )
, ( SMALLINT( 4) , :id3 , :ol_quantity3
, :supply_w_id3 )
, ( SMALLINT( 5) , :id4 , :ol_quantity4
, :supply_w_id4 )
)

```

```

, :supply_w_id5 ) , ( SMALLINT( 6 ) , :id5 , :ol_quantity5
, :supply_w_id6 ) , ( SMALLINT( 7 ) , :id6 , :ol_quantity6
, :supply_w_id7 ) , ( SMALLINT( 8 ) , :id7 , :ol_quantity7
, :supply_w_id8 ) , ( SMALLINT( 9 ) , :id8 , :ol_quantity8
:ol_quantity9 , :supply_w_id9 ) , ( SMALLINT( 10 ) , :id9 ,
:ol_quantity10 , :supply_w_id10 ) , ( SMALLINT( 11 ) , :id10 ,
, I_SUPPLY_W_ID ) ) AS X ( OL_NUMBER , I_ID , I_QTY
) AS ITEMLIST
, TABLE( NEW_OL_ALL( I_ID
, I_QTY
, W_ID
, I_SUPPLY_W_ID
, O_ID
, D_ID
)
) AS NEW_OL_ALL
WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
)
SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA ,
S_QUANTITY
FROM NEW TABLE ( INSERT INTO ORDER_LINE
( OL_O_ID
, OL_D_ID
, OL_W_ID
, OL_NUMBER
, OL_I_ID
, OL_SUPPLY_W_ID
, OL_DELIVERY_D
, OL_QUANTITY
, OL_AMOUNT
, OL_DIST_INFO
)
INCLUDE( I_PRICE INTEGER
, I_NAME CHAR(24)
, I_DATA VARCHAR(50)
, S_DATA VARCHAR(50)
, S_QUANTITY SMALLINT )
SELECT O_ID
, D_ID
, W_ID
, OL_NUMBER
, I_ID
, I_SUPPLY_W_ID
, OL_DELIVERY_D
, I_QTY
, TOTAL_PRICE
, OL_DIST_INFO
, I_PRICE, I_NAME, I_DATA, S_DATA,
S_QUANTITY
FROM DATA
) AS INS
;
EXEC SQL DECLARE ISOL_Remote_12 CURSOR FOR
WITH DATA AS ( SELECT O_ID
, D_ID
, W_ID
, OL_NUMBER
, I_ID
, I_SUPPLY_W_ID
, 0 AS OL_DELIVERY_D
, I_QTY
, ( I_PRICE * I_QTY ) AS TOTAL_PRICE
, OL_DIST_INFO
, I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY
FROM ( SELECT :next_o_id as O_ID
, :w_id AS W_ID
, :d_id as D_ID
, OL_NUMBER
, I_ID
, I_SUPPLY_W_ID
, I_QTY
FROM Table( VALUES
( SMALLINT( 1 ) , :id0 , :ol_quantity0
, :supply_w_id0 )
, ( SMALLINT( 2 ) , :id1 , :ol_quantity1
, :supply_w_id1 )
, ( SMALLINT( 3 ) , :id2 , :ol_quantity2
, :supply_w_id2 )
, ( SMALLINT( 4 ) , :id3 , :ol_quantity3
, :supply_w_id3 )
, ( SMALLINT( 5 ) , :id4 , :ol_quantity4
, :supply_w_id4 )
, ( SMALLINT( 6 ) , :id5 , :ol_quantity5
, :supply_w_id5 )
, ( SMALLINT( 7 ) , :id6 , :ol_quantity6
, :supply_w_id6 )
, ( SMALLINT( 8 ) , :id7 , :ol_quantity7
, :supply_w_id7 )
, ( SMALLINT( 9 ) , :id8 , :ol_quantity8
, :supply_w_id8 )
, ( SMALLINT( 10 ) , :id9 ,
:ol_quantity9 , :supply_w_id9 )
, ( SMALLINT( 11 ) , :id10 ,
:ol_quantity10 , :supply_w_id10 )
, ( SMALLINT( 12 ) , :id11 ,
:ol_quantity11 , :supply_w_id11 )
) AS X ( OL_NUMBER , I_ID , I_QTY
, I_SUPPLY_W_ID )
) AS ITEMLIST
, TABLE( NEW_OL_ALL( I_ID
, I_QTY
, W_ID
, I_SUPPLY_W_ID
, O_ID
, D_ID
)
) AS NEW_OL_ALL
WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
)
SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA ,
S_QUANTITY
FROM NEW TABLE ( INSERT INTO ORDER_LINE
( OL_O_ID
, OL_D_ID
, OL_W_ID
, OL_NUMBER
, OL_I_ID
, OL_SUPPLY_W_ID
, OL_DELIVERY_D
, OL_QUANTITY
, OL_AMOUNT
, OL_DIST_INFO
)
INCLUDE( I_PRICE INTEGER
, I_NAME CHAR(24)
, I_DATA VARCHAR(50)
, S_DATA VARCHAR(50)
)

```



```

        ,S_QUANTITY SMALLINT )
SELECT O_ID
      ,D_ID
      ,W_ID
      ,OL_NUMBER
      ,I_ID
      ,I_SUPPLY_W_ID
      ,OL_DELIVERY_D
      ,I_QTY
      ,TOTAL_PRICE
      ,OL_DIST_INFO
      ,I_PRICE,I_NAME,I_DATA,S_DATA,
S_QUANTITY
FROM DATA
) AS INS
;
EXEC SQL DECLARE ISOL_Remote_13 CURSOR FOR
WITH DATA AS ( SELECT O_ID
      ,D_ID
      ,W_ID
      ,OL_NUMBER
      ,I_ID
      ,I_SUPPLY_W_ID
      ,0 AS OL_DELIVERY_D
      ,I_QTY
      ,(I_PRICE * I_QTY) AS TOTAL_PRICE
      ,OL_DIST_INFO
      ,I_PRICE,I_NAME,I_DATA,S_DATA,S_QUANTITY
FROM ( SELECT :next_o_id as O_ID
      ,:w_id AS W_ID
      ,:d_id as D_ID
      ,OL_NUMBER
      ,I_ID
      ,I_SUPPLY_W_ID
      ,I_QTY
FROM Table( VALUES
      ( SMALLINT( 1) ,:id0 ,:ol_quantity0
, :supply_w_id0 )
      ,( SMALLINT( 2) ,:id1 ,:ol_quantity1
, :supply_w_id1 )
      ,( SMALLINT( 3) ,:id2 ,:ol_quantity2
, :supply_w_id2 )
      ,( SMALLINT( 4) ,:id3 ,:ol_quantity3
, :supply_w_id3 )
      ,( SMALLINT( 5) ,:id4 ,:ol_quantity4
, :supply_w_id4 )
      ,( SMALLINT( 6) ,:id5 ,:ol_quantity5
, :supply_w_id5 )
      ,( SMALLINT( 7) ,:id6 ,:ol_quantity6
, :supply_w_id6 )
      ,( SMALLINT( 8) ,:id7 ,:ol_quantity7
, :supply_w_id7 )
      ,( SMALLINT( 9) ,:id8 ,:ol_quantity8
, :supply_w_id8 )
      ,( SMALLINT( 10) ,:id9 ,
:ol_quantity9 ,:supply_w_id9 )
      ,( SMALLINT( 11) ,:id10 ,
:ol_quantity10 ,:supply_w_id10 )
      ,( SMALLINT( 12) ,:id11 ,
:ol_quantity11 ,:supply_w_id11 )
      ,( SMALLINT( 13) ,:id12 ,
:ol_quantity12 ,:supply_w_id12 )
) AS X (OL_NUMBER ,I_ID ,I_QTY
,I_SUPPLY_W_ID )
) AS ITEMLIST
, TABLE(NEW_OL_ALL( I_ID
      ,I_QTY
      ,W_ID

```

```

      ,I_SUPPLY_W_ID
      ,O_ID
      ,D_ID
)
) AS NEW_OL_ALL
WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
)
SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA ,
S_QUANTITY
FROM NEW TABLE ( INSERT INTO ORDER_LINE
( OL_O_ID
, OL_D_ID
, OL_W_ID
, OL_NUMBER
, OL_I_ID
, OL_SUPPLY_W_ID
, OL_DELIVERY_D
, OL_QUANTITY
, OL_AMOUNT
, OL_DIST_INFO
)
INCLUDE( I_PRICE INTEGER
, I_NAME CHAR(24)
, I_DATA VARCHAR(50)
, S_DATA VARCHAR(50)
, S_QUANTITY SMALLINT )
SELECT O_ID
      ,D_ID
      ,W_ID
      ,OL_NUMBER
      ,I_ID
      ,I_SUPPLY_W_ID
      ,OL_DELIVERY_D
      ,I_QTY
      ,TOTAL_PRICE
      ,OL_DIST_INFO
      ,I_PRICE,I_NAME,I_DATA,S_DATA,
S_QUANTITY
FROM DATA
) AS INS
;
EXEC SQL DECLARE ISOL_Remote_14 CURSOR FOR
WITH DATA AS ( SELECT O_ID
      ,D_ID
      ,W_ID
      ,OL_NUMBER
      ,I_ID
      ,I_SUPPLY_W_ID
      ,0 AS OL_DELIVERY_D
      ,I_QTY
      ,(I_PRICE * I_QTY) AS TOTAL_PRICE
      ,OL_DIST_INFO
      ,I_PRICE,I_NAME,I_DATA,S_DATA,S_QUANTITY
FROM ( SELECT :next_o_id as O_ID
      ,:w_id AS W_ID
      ,:d_id as D_ID
      ,OL_NUMBER
      ,I_ID
      ,I_SUPPLY_W_ID
      ,I_QTY
FROM Table( VALUES
      ( SMALLINT( 1) ,:id0 ,:ol_quantity0
, :supply_w_id0 )
      ,( SMALLINT( 2) ,:id1 ,:ol_quantity1
, :supply_w_id1 )
      ,( SMALLINT( 3) ,:id2 ,:ol_quantity2
, :supply_w_id2 )

```

```

, :supply_w_id3 )          , ( SMALLINT( 4 )      , :id3 , :ol_quantity3
, :supply_w_id4 )          , ( SMALLINT( 5 )      , :id4 , :ol_quantity4
, :supply_w_id5 )          , ( SMALLINT( 6 )      , :id5 , :ol_quantity5
, :supply_w_id6 )          , ( SMALLINT( 7 )      , :id6 , :ol_quantity6
, :supply_w_id7 )          , ( SMALLINT( 8 )      , :id7 , :ol_quantity7
, :supply_w_id8 )          , ( SMALLINT( 9 )      , :id8 , :ol_quantity8
:ol_quantity9 , :supply_w_id9 )
:ol_quantity10 , :supply_w_id10 )
:ol_quantity11 , :supply_w_id11 )
:ol_quantity12 , :supply_w_id12 )
:ol_quantity13 , :supply_w_id13 )
) AS X (OL_NUMBER , I_ID , I_QTY
, I_SUPPLY_W_ID )
) AS ITEMLIST
, TABLE(NEW_OL_ALL( I_ID
, I_QTY
, W_ID
, I_SUPPLY_W_ID
, O_ID
, D_ID
)
) AS NEW_OL_ALL
WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
)
SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA ,
S_QUANTITY
FROM NEW TABLE ( INSERT INTO ORDER_LINE
( OL_O_ID
, OL_D_ID
, OL_W_ID
, OL_NUMBER
, OL_I_ID
, OL_SUPPLY_W_ID
, OL_DELIVERY_D
, OL_QUANTITY
, OL_AMOUNT
, OL_DIST_INFO
)
INCLUDE( I_PRICE INTEGER
, I_NAME CHAR(24)
, I_DATA VARCHAR(50)
, S_DATA VARCHAR(50)
, S_QUANTITY SMALLINT )
SELECT O_ID
, D_ID
, W_ID
, OL_NUMBER
, I_ID
, I_SUPPLY_W_ID
, OL_DELIVERY_D
, I_QTY
, TOTAL_PRICE
, OL_DIST_INFO
, I_PRICE, I_NAME, I_DATA, S_DATA,
S_QUANTITY
FROM DATA
) AS INS
;
EXEC SQL DECLARE ISOL_Remote_15 CURSOR FOR
WITH DATA AS ( SELECT O_ID
, D_ID
, W_ID
, OL_NUMBER
, I_ID
, I_SUPPLY_W_ID
, 0 AS OL_DELIVERY_D
, I_QTY
, (I_PRICE * I_QTY) AS TOTAL_PRICE
, OL_DIST_INFO
, I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY
FROM ( SELECT :next_o_id as O_ID
, :w_id AS W_ID
, :d_id as D_ID
, OL_NUMBER
, I_ID
, I_SUPPLY_W_ID
, I_QTY
FROM Table( VALUES
( SMALLINT( 1 )      , :id0 , :ol_quantity0
, :supply_w_id0 )
, ( SMALLINT( 2 )      , :id1 , :ol_quantity1
, :supply_w_id1 )
, ( SMALLINT( 3 )      , :id2 , :ol_quantity2
, :supply_w_id2 )
, ( SMALLINT( 4 )      , :id3 , :ol_quantity3
, :supply_w_id3 )
, ( SMALLINT( 5 )      , :id4 , :ol_quantity4
, :supply_w_id4 )
, ( SMALLINT( 6 )      , :id5 , :ol_quantity5
, :supply_w_id5 )
, ( SMALLINT( 7 )      , :id6 , :ol_quantity6
, :supply_w_id6 )
, ( SMALLINT( 8 )      , :id7 , :ol_quantity7
, :supply_w_id7 )
, ( SMALLINT( 9 )      , :id8 , :ol_quantity8
, :supply_w_id8 )
, ( SMALLINT( 10 )     , :id9 ,
:ol_quantity9 , :supply_w_id9 )
, ( SMALLINT( 11 )     , :id10 ,
:ol_quantity10 , :supply_w_id10 )
, ( SMALLINT( 12 )     , :id11 ,
:ol_quantity11 , :supply_w_id11 )
, ( SMALLINT( 13 )     , :id12 ,
:ol_quantity12 , :supply_w_id12 )
, ( SMALLINT( 14 )     , :id13 ,
:ol_quantity13 , :supply_w_id13 )
, ( SMALLINT( 15 )     , :id14 ,
:ol_quantity14 , :supply_w_id14 )
) AS X (OL_NUMBER , I_ID , I_QTY
, I_SUPPLY_W_ID )
) AS ITEMLIST
, TABLE(NEW_OL_ALL( I_ID
, I_QTY
, W_ID
, I_SUPPLY_W_ID
, O_ID
, D_ID
)
) AS NEW_OL_ALL
WHERE NEW_OL_ALL.I_PRICE IS NOT NULL
)
SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA ,
S_QUANTITY

```

```

FROM NEW TABLE ( INSERT INTO ORDER_LINE
  ( OL_O_ID
  ,OL_D_ID
  ,OL_W_ID
  ,OL_NUMBER
  ,OL_I_ID
  ,OL_SUPPLY_W_ID
  ,OL_DELIVERY_D
  ,OL_QUANTITY
  ,OL_AMOUNT
  ,OL_DIST_INFO
  )
  INCLUDE( I_PRICE INTEGER
  ,I_NAME CHAR(24)
  ,I_DATA VARCHAR(50)
  ,S_DATA VARCHAR(50)
  ,S_QUANTITY SMALLINT )
  SELECT O_ID
  ,D_ID
  ,W_ID
  ,OL_NUMBER
  ,I_ID
  ,I_SUPPLY_W_ID
  ,OL_DELIVERY_D
  ,I_QTY
  ,TOTAL_PRICE
  ,OL_DIST_INFO
  ,I_PRICE, I_NAME, I_DATA, S_DATA,
S_QUANTITY
  FROM DATA
  ) AS INS
;
EXEC SQL DECLARE ISOL_Local_1 CURSOR FOR

WITH DATA AS ( SELECT O_ID
  ,D_ID
  ,W_ID
  ,OL_NUMBER
  ,I_ID
  ,W_ID AS I_SUPPLY_W_ID
  ,0 AS OL_DELIVERY_D
  ,I_QTY
  ,(I_PRICE * I_QTY) AS TOTAL_PRICE
  ,OL_DIST_INFO
  ,I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY
  FROM ( SELECT :next_o_id as O_ID
  ,:w_id AS W_ID

  ,:d_id as D_ID
  ,OL_NUMBER
  ,I_ID
  ,I_QTY
  FROM Table( VALUES
  ( SMALLINT(1) , :id0 , :ol_quantity0 )
  ) AS X (OL_NUMBER , I_ID , I_QTY
  )
  ) AS ITEMLIST
  ,TABLE(NEW_OL_LOCAL( I_ID
  ,I_QTY
  ,W_ID
  ,O_ID
  ,D_ID
  )
  ) AS NEW_OL_LOCAL
  WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
  )

```

```

SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA ,
S_QUANTITY
FROM NEW TABLE ( INSERT INTO ORDER_LINE
  ( OL_O_ID
  ,OL_D_ID
  ,OL_W_ID
  ,OL_NUMBER
  ,OL_I_ID
  ,OL_SUPPLY_W_ID
  ,OL_DELIVERY_D
  ,OL_QUANTITY
  ,OL_AMOUNT
  ,OL_DIST_INFO
  )
  INCLUDE( I_PRICE INTEGER
  ,I_NAME CHAR(24)
  ,I_DATA VARCHAR(50)
  ,S_DATA VARCHAR(50)
  ,S_QUANTITY SMALLINT )
  SELECT O_ID
  ,D_ID
  ,W_ID
  ,OL_NUMBER
  ,I_ID
  ,I_SUPPLY_W_ID
  ,OL_DELIVERY_D
  ,I_QTY
  ,TOTAL_PRICE
  ,OL_DIST_INFO
  ,I_PRICE, I_NAME, I_DATA, S_DATA,
S_QUANTITY
  FROM DATA
  ) AS INS
;
EXEC SQL DECLARE ISOL_Local_2 CURSOR FOR

WITH DATA AS ( SELECT O_ID
  ,D_ID

  ,W_ID
  ,OL_NUMBER
  ,I_ID
  ,W_ID AS I_SUPPLY_W_ID
  ,0 AS OL_DELIVERY_D
  ,I_QTY
  ,(I_PRICE * I_QTY) AS TOTAL_PRICE
  ,OL_DIST_INFO
  ,I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY
  FROM ( SELECT :next_o_id as O_ID
  ,:w_id AS W_ID
  ,:d_id as D_ID
  ,OL_NUMBER
  ,I_ID
  ,I_QTY
  FROM Table( VALUES
  ( SMALLINT(1) , :id0 , :ol_quantity0 )
  ,( SMALLINT(2) , :id1 , :ol_quantity1 )
  ) AS X (OL_NUMBER , I_ID , I_QTY
  )
  ) AS ITEMLIST
  ,TABLE(NEW_OL_LOCAL( I_ID
  ,I_QTY
  ,W_ID
  ,O_ID
  ,D_ID
  )
  ) AS NEW_OL_LOCAL
  WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
  )

```

```

)
SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA ,
S_QUANTITY
FROM NEW TABLE ( INSERT INTO ORDER_LINE
( OL_O_ID
, OL_D_ID
, OL_W_ID
, OL_NUMBER
, OL_I_ID
, OL_SUPPLY_W_ID
, OL_DELIVERY_D
, OL_QUANTITY
, OL_AMOUNT
, OL_DIST_INFO
)
INCLUDE( I_PRICE INTEGER
, I_NAME CHAR(24)
, I_DATA VARCHAR(50)
, S_DATA VARCHAR(50)
, S_QUANTITY SMALLINT )
SELECT O_ID
, D_ID
, W_ID
, OL_NUMBER
, I_ID
, I_SUPPLY_W_ID
, OL_DELIVERY_D
, I_QTY
, TOTAL_PRICE
, OL_DIST_INFO
, I_PRICE, I_NAME, I_DATA, S_DATA,
S_QUANTITY
FROM DATA
) AS INS
;
EXEC SQL DECLARE ISOL_Local_3 CURSOR FOR
WITH DATA AS ( SELECT O_ID
, D_ID
, W_ID
, OL_NUMBER
, I_ID
, W_ID AS I_SUPPLY_W_ID
, 0 AS OL_DELIVERY_D
, I_QTY
, (I_PRICE * I_QTY) AS TOTAL_PRICE
, OL_DIST_INFO
, I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY
FROM ( SELECT :next_o_id as O_ID
, :w_id AS W_ID
, :d_id as D_ID
, OL_NUMBER
, I_ID
, I_QTY
FROM Table( VALUES
( SMALLINT(1) , :id0 , :ol_quantity0 )
, ( SMALLINT(2) , :id1 , :ol_quantity1 )
, ( SMALLINT(3) , :id2 , :ol_quantity2 )
) AS X (OL_NUMBER , I_ID , I_QTY
)
) AS ITEMLIST
, TABLE(NEW_OL_LOCAL( I_ID
, I_QTY
, W_ID
, O_ID
, D_ID

```

```

)
) AS NEW_OL_LOCAL
WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
)
SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA ,
S_QUANTITY
FROM NEW TABLE ( INSERT INTO ORDER_LINE
( OL_O_ID
, OL_D_ID
, OL_W_ID
, OL_NUMBER
, OL_I_ID
, OL_SUPPLY_W_ID
, OL_DELIVERY_D
, OL_QUANTITY
, OL_AMOUNT
, OL_DIST_INFO
)
INCLUDE( I_PRICE INTEGER
, I_NAME CHAR(24)
, I_DATA VARCHAR(50)
, S_DATA VARCHAR(50)
, S_QUANTITY SMALLINT )
SELECT O_ID
, D_ID
, W_ID
, OL_NUMBER
, I_ID
, I_SUPPLY_W_ID
, OL_DELIVERY_D
, I_QTY
, TOTAL_PRICE
, OL_DIST_INFO
, I_PRICE, I_NAME, I_DATA, S_DATA,
S_QUANTITY
FROM DATA
) AS INS
;
EXEC SQL DECLARE ISOL_Local_4 CURSOR FOR
WITH DATA AS ( SELECT O_ID
, D_ID
, W_ID
, OL_NUMBER
, I_ID
, W_ID AS I_SUPPLY_W_ID
, 0 AS OL_DELIVERY_D
, I_QTY
, (I_PRICE * I_QTY) AS TOTAL_PRICE
, OL_DIST_INFO
, I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY
FROM ( SELECT :next_o_id as O_ID
, :w_id AS W_ID
, :d_id as D_ID
, OL_NUMBER
, I_ID
, I_QTY
FROM Table( VALUES
( SMALLINT(1) , :id0 , :ol_quantity0 )
, ( SMALLINT(2) , :id1 , :ol_quantity1 )
, ( SMALLINT(3) , :id2 , :ol_quantity2 )
, ( SMALLINT(4) , :id3 , :ol_quantity3 )
) AS X (OL_NUMBER , I_ID , I_QTY
)
) AS ITEMLIST
, TABLE(NEW_OL_LOCAL( I_ID

```

```

                ,I_QTY
                ,W_ID
                ,O_ID
                ,D_ID
            )
        ) AS NEW_OL_LOCAL
        WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
    )
SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA ,
S_QUANTITY
FROM NEW TABLE ( INSERT INTO ORDER_LINE
    ( OL_O_ID
    , OL_D_ID
    , OL_W_ID
    , OL_NUMBER
    , OL_I_ID
    , OL_SUPPLY_W_ID
    , OL_DELIVERY_D
    , OL_QUANTITY
    , OL_AMOUNT
    , OL_DIST_INFO
    )
INCLUDE ( I_PRICE INTEGER
    , I_NAME CHAR(24)
    , I_DATA VARCHAR(50)
    , S_DATA VARCHAR(50)
    , S_QUANTITY SMALLINT )
SELECT O_ID
    , D_ID
    , W_ID
    , OL_NUMBER
    , I_ID
    , I_SUPPLY_W_ID
    , OL_DELIVERY_D
    , I_QTY
    , TOTAL_PRICE
    , OL_DIST_INFO
    , I_PRICE, I_NAME, I_DATA, S_DATA,
S_QUANTITY
FROM DATA
) AS INS
;
EXEC SQL DECLARE ISOL_Local_5 CURSOR FOR
WITH DATA AS ( SELECT O_ID
    , D_ID
    , W_ID
    , OL_NUMBER
    , I_ID
    , W_ID AS I_SUPPLY_W_ID
    , 0 AS OL_DELIVERY_D
    , I_QTY
    , (I_PRICE * I_QTY) AS TOTAL_PRICE
    , OL_DIST_INFO
    , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY
    FROM ( SELECT :next_o_id as O_ID
        , :w_id AS W_ID
        , :d_id as D_ID
        , OL_NUMBER
        , I_ID
        , I_QTY
        FROM Table( VALUES
            ( SMALLINT( 1 ) , :id0 , :ol_quantity0 )
            , ( SMALLINT( 2 ) , :id1 , :ol_quantity1 )
            , ( SMALLINT( 3 ) , :id2 , :ol_quantity2 )

```

```

                , ( SMALLINT( 4 ) , :id3 , :ol_quantity3 )
                , ( SMALLINT( 5 ) , :id4 , :ol_quantity4 )
            ) AS X ( OL_NUMBER , I_ID , I_QTY
        ) AS ITEM LIST
    , TABLE( NEW_OL_LOCAL( I_ID
        , I_QTY
        , W_ID
        , O_ID
        , D_ID
        )
        ) AS NEW_OL_LOCAL
        WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
    )
SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA ,
S_QUANTITY
FROM NEW TABLE ( INSERT INTO ORDER_LINE
    ( OL_O_ID
    , OL_D_ID
    , OL_W_ID
    , OL_NUMBER
    , OL_I_ID
    , OL_SUPPLY_W_ID
    , OL_DELIVERY_D
    , OL_QUANTITY
    , OL_AMOUNT
    , OL_DIST_INFO
    )
INCLUDE ( I_PRICE INTEGER
    , I_NAME CHAR(24)
    , I_DATA VARCHAR(50)
    , S_DATA VARCHAR(50)
    , S_QUANTITY SMALLINT )
SELECT O_ID
    , D_ID
    , W_ID
    , OL_NUMBER
    , I_ID
    , I_SUPPLY_W_ID
    , OL_DELIVERY_D
    , I_QTY
    , TOTAL_PRICE
    , OL_DIST_INFO
    , I_PRICE, I_NAME, I_DATA, S_DATA,
S_QUANTITY
FROM DATA
) AS INS
;
EXEC SQL DECLARE ISOL_Local_6 CURSOR FOR
WITH DATA AS ( SELECT O_ID
    , D_ID
    , W_ID
    , OL_NUMBER
    , I_ID
    , W_ID AS I_SUPPLY_W_ID
    , 0 AS OL_DELIVERY_D
    , I_QTY
    , (I_PRICE * I_QTY) AS TOTAL_PRICE
    , OL_DIST_INFO
    , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY
    FROM ( SELECT :next_o_id as O_ID
        , :w_id AS W_ID
        , :d_id as D_ID
        , OL_NUMBER
        , I_ID
        FROM ( SELECT :next_o_id as O_ID
            , :w_id AS W_ID
            , :d_id as D_ID
            , OL_NUMBER
            , I_ID

```

```

        ,I_QTY
    FROM Table( VALUES
        ( SMALLINT(1) , :id0 , :ol_quantity0 )
        , ( SMALLINT(2) , :id1 , :ol_quantity1 )
        , ( SMALLINT(3) , :id2 , :ol_quantity2 )
        , ( SMALLINT(4) , :id3 , :ol_quantity3 )
        , ( SMALLINT(5) , :id4 , :ol_quantity4 )
        , ( SMALLINT(6) , :id5 , :ol_quantity5 )
    ) AS X (OL_NUMBER , I_ID , I_QTY
)
    ) AS ITEMLIST
    , TABLE(NEW_OL_LOCAL( I_ID
        , I_QTY
        , W_ID
        , O_ID
        , D_ID
    )
    ) AS NEW_OL_LOCAL
    WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
)
SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA ,
S_QUANTITY
FROM NEW TABLE ( INSERT INTO ORDER_LINE
    ( OL_O_ID
    , OL_D_ID
    , OL_W_ID
    , OL_NUMBER
    , OL_I_ID
    , OL_SUPPLY_W_ID
    , OL_DELIVERY_D
    , OL_QUANTITY
    , OL_AMOUNT
    , OL_DIST_INFO
    )
    INCLUDE( I_PRICE INTEGER
    , I_NAME CHAR(24)
    , I_DATA VARCHAR(50)
    , S_DATA VARCHAR(50)
    , S_QUANTITY SMALLINT )
    SELECT O_ID
    , D_ID
    , W_ID
    , OL_NUMBER
    , I_ID
    , I_SUPPLY_W_ID
    , OL_DELIVERY_D
    , I_QTY
    , TOTAL_PRICE
    , OL_DIST_INFO
    , I_PRICE, I_NAME, I_DATA, S_DATA,
S_QUANTITY
    FROM DATA
    ) AS INS
;
EXEC SQL DECLARE ISOL_Local_7 CURSOR FOR

WITH DATA AS ( SELECT O_ID
    , D_ID
    , W_ID
    , OL_NUMBER
    , I_ID
    , W_ID AS I_SUPPLY_W_ID
    , O AS OL_DELIVERY_D
    , I_QTY
    , (I_PRICE * I_QTY) AS TOTAL_PRICE
    , OL_DIST_INFO
    , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

```

```

FROM ( SELECT :next_o_id as O_ID
    , :w_id AS W_ID
    , :d_id as D_ID
    , OL_NUMBER
    , I_ID
    , I_QTY
    FROM Table( VALUES
        ( SMALLINT(1) , :id0 , :ol_quantity0 )
        , ( SMALLINT(2) , :id1 , :ol_quantity1 )
        , ( SMALLINT(3) , :id2 , :ol_quantity2 )
        , ( SMALLINT(4) , :id3 , :ol_quantity3 )
        , ( SMALLINT(5) , :id4 , :ol_quantity4 )
        , ( SMALLINT(6) , :id5 , :ol_quantity5 )
        , ( SMALLINT(7) , :id6 , :ol_quantity6 )
    ) AS X (OL_NUMBER , I_ID , I_QTY
)
    ) AS ITEMLIST
    , TABLE(NEW_OL_LOCAL( I_ID
        , I_QTY
        , W_ID
        , O_ID
        , D_ID
    )
    ) AS NEW_OL_LOCAL
    WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
)
SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA ,
S_QUANTITY
FROM NEW TABLE ( INSERT INTO ORDER_LINE
    ( OL_O_ID
    , OL_D_ID
    , OL_W_ID
    , OL_NUMBER
    , OL_I_ID
    , OL_SUPPLY_W_ID
    , OL_DELIVERY_D
    , OL_QUANTITY
    , OL_AMOUNT
    , OL_DIST_INFO
    )
    INCLUDE( I_PRICE INTEGER
    , I_NAME CHAR(24)
    , I_DATA VARCHAR(50)
    , S_DATA VARCHAR(50)
    , S_QUANTITY SMALLINT )
    SELECT O_ID
    , D_ID
    , W_ID
    , OL_NUMBER
    , I_ID
    , I_SUPPLY_W_ID
    , OL_DELIVERY_D
    , I_QTY
    , TOTAL_PRICE
    , OL_DIST_INFO
    , I_PRICE, I_NAME, I_DATA, S_DATA,
S_QUANTITY
    FROM DATA
    ) AS INS
;
EXEC SQL DECLARE ISOL_Local_8 CURSOR FOR

WITH DATA AS ( SELECT O_ID
    , D_ID
    , W_ID
    , OL_NUMBER

```

```

,I_ID
,W_ID AS I_SUPPLY_W_ID
,0 AS OL_DELIVERY_D
,I_QTY
,(I_PRICE * I_QTY) AS TOTAL_PRICE
,OL_DIST_INFO
,I_PRICE,I_NAME,I_DATA,S_DATA,S_QUANTITY

FROM ( SELECT :next_o_id as O_ID
      ,:w_id AS W_ID
      ,:d_id as D_ID
      ,OL_NUMBER
      ,I_ID
      ,I_QTY
FROM Table( VALUES
      ( SMALLINT(1) ,:id0 ,:ol_quantity0 )
      ,( SMALLINT(2) ,:id1 ,:ol_quantity1 )
      ,( SMALLINT(3) ,:id2 ,:ol_quantity2 )
      ,( SMALLINT(4) ,:id3 ,:ol_quantity3 )
      ,( SMALLINT(5) ,:id4 ,:ol_quantity4 )
      ,( SMALLINT(6) ,:id5 ,:ol_quantity5 )
      ,( SMALLINT(7) ,:id6 ,:ol_quantity6 )
      ,( SMALLINT(8) ,:id7 ,:ol_quantity7 )
) AS X (OL_NUMBER , I_ID , I_QTY
)

) AS ITEMLIST
, TABLE(NEW_OL_LOCAL( I_ID
      ,I_QTY
      ,W_ID
      ,O_ID
      ,D_ID
)
) AS NEW_OL_LOCAL
WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
)

SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA ,
S_QUANTITY
FROM NEW TABLE ( INSERT INTO ORDER_LINE
( OL_O_ID
, OL_D_ID
, OL_W_ID
, OL_NUMBER
, OL_I_ID
, OL_SUPPLY_W_ID
, OL_DELIVERY_D
, OL_QUANTITY
, OL_AMOUNT
, OL_DIST_INFO
)
INCLUDE( I_PRICE INTEGER
,I_NAME CHAR(24)
,I_DATA VARCHAR(50)
,S_DATA VARCHAR(50)
,S_QUANTITY SMALLINT )
SELECT O_ID
,D_ID
,W_ID
,OL_NUMBER
,I_ID
,I_SUPPLY_W_ID
,OL_DELIVERY_D
,I_QTY
,TOTAL_PRICE

, OL_DIST_INFO
,I_PRICE,I_NAME,I_DATA,S_DATA,
S_QUANTITY

```

```

FROM DATA
) AS INS
;
EXEC SQL DECLARE ISOL_Local_9 CURSOR FOR

WITH DATA AS ( SELECT O_ID
      ,D_ID
      ,W_ID
      ,OL_NUMBER
      ,I_ID
      ,W_ID AS I_SUPPLY_W_ID
      ,0 AS OL_DELIVERY_D
      ,I_QTY
      ,(I_PRICE * I_QTY) AS TOTAL_PRICE
      ,OL_DIST_INFO
      ,I_PRICE,I_NAME,I_DATA,S_DATA,S_QUANTITY

FROM ( SELECT :next_o_id as O_ID
      ,:w_id AS W_ID
      ,:d_id as D_ID
      ,OL_NUMBER
      ,I_ID
      ,I_QTY
FROM Table( VALUES
      ( SMALLINT(1) ,:id0 ,:ol_quantity0 )
      ,( SMALLINT(2) ,:id1 ,:ol_quantity1 )
      ,( SMALLINT(3) ,:id2 ,:ol_quantity2 )
      ,( SMALLINT(4) ,:id3 ,:ol_quantity3 )
      ,( SMALLINT(5) ,:id4 ,:ol_quantity4 )
      ,( SMALLINT(6) ,:id5 ,:ol_quantity5 )
      ,( SMALLINT(7) ,:id6 ,:ol_quantity6 )
      ,( SMALLINT(8) ,:id7 ,:ol_quantity7 )
      ,( SMALLINT(9) ,:id8 ,:ol_quantity8 )
) AS X (OL_NUMBER , I_ID , I_QTY
)

) AS ITEMLIST
, TABLE(NEW_OL_LOCAL( I_ID
      ,I_QTY
      ,W_ID
      ,O_ID
      ,D_ID
)
) AS NEW_OL_LOCAL
WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
)

SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA ,
S_QUANTITY
FROM NEW TABLE ( INSERT INTO ORDER_LINE
( OL_O_ID
, OL_D_ID
, OL_W_ID
, OL_NUMBER
, OL_I_ID
, OL_SUPPLY_W_ID
, OL_DELIVERY_D
, OL_QUANTITY
, OL_AMOUNT
, OL_DIST_INFO
)
INCLUDE( I_PRICE INTEGER
,I_NAME CHAR(24)
,I_DATA VARCHAR(50)
,S_DATA VARCHAR(50)
,S_QUANTITY SMALLINT )
SELECT O_ID
,D_ID
,W_ID

```

```

        ,OL_NUMBER
        ,I_ID
        ,I_SUPPLY_W_ID
        ,OL_DELIVERY_D
        ,I_QTY
        ,TOTAL_PRICE
        ,OL_DIST_INFO
        ,I_PRICE, I_NAME, I_DATA, S_DATA,
S_QUANTITY
    ) AS INS
;
EXEC SQL DECLARE ISOL_Local_10 CURSOR FOR

WITH DATA AS ( SELECT O_ID
        ,D_ID
        ,W_ID
        ,OL_NUMBER
        ,I_ID
        ,W_ID AS I_SUPPLY_W_ID
        ,0 AS OL_DELIVERY_D
        ,I_QTY
        ,(I_PRICE * I_QTY) AS TOTAL_PRICE
        ,OL_DIST_INFO
        ,I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

    FROM ( SELECT :next_o_id as O_ID
        ,:w_id AS W_ID
        ,:d_id as D_ID
        ,OL_NUMBER
        ,I_ID
        ,I_QTY
    FROM Table( VALUES
        ( SMALLINT(1) ,:id0 ,:ol_quantity0 )
        ,( SMALLINT(2) ,:id1 ,:ol_quantity1 )
        ,( SMALLINT(3) ,:id2 ,:ol_quantity2 )
        ,( SMALLINT(4) ,:id3 ,:ol_quantity3 )
        ,( SMALLINT(5) ,:id4 ,:ol_quantity4 )
        ,( SMALLINT(6) ,:id5 ,:ol_quantity5 )
        ,( SMALLINT(7) ,:id6 ,:ol_quantity6 )
        ,( SMALLINT(8) ,:id7 ,:ol_quantity7 )
        ,( SMALLINT(9) ,:id8 ,:ol_quantity8 )
        ,( SMALLINT(10) ,:id9 ,:ol_quantity9 )
    ) AS X (OL_NUMBER , I_ID , I_QTY

    )
    ) AS ITEM_LIST
    ,TABLE(NEW_OL_LOCAL( I_ID
        ,I_QTY
        ,W_ID
        ,O_ID
        ,D_ID
    )
    ) AS NEW_OL_LOCAL
    WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
)

SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA ,
S_QUANTITY
FROM NEW TABLE ( INSERT INTO ORDER_LINE
    ( OL_O_ID
    ,OL_D_ID
    ,OL_W_ID
    ,OL_NUMBER
    ,OL_I_ID
    ,OL_SUPPLY_W_ID
    ,OL_DELIVERY_D
    ,OL_QUANTITY
    ,OL_AMOUNT

```

```

        ,OL_DIST_INFO
    )
    INCLUDE ( I_PRICE INTEGER
        ,I_NAME CHAR(24)
        ,I_DATA VARCHAR(50)
        ,S_DATA VARCHAR(50)
        ,S_QUANTITY SMALLINT )
    SELECT O_ID
        ,D_ID
        ,W_ID
        ,OL_NUMBER
        ,I_ID
        ,I_SUPPLY_W_ID
        ,OL_DELIVERY_D
        ,I_QTY
        ,TOTAL_PRICE
        ,OL_DIST_INFO
        ,I_PRICE, I_NAME, I_DATA, S_DATA,
S_QUANTITY
    ) AS INS
;
EXEC SQL DECLARE ISOL_Local_11 CURSOR FOR

WITH DATA AS ( SELECT O_ID
        ,D_ID
        ,W_ID
        ,OL_NUMBER
        ,I_ID
        ,W_ID AS I_SUPPLY_W_ID
        ,0 AS OL_DELIVERY_D
        ,I_QTY
        ,(I_PRICE * I_QTY) AS TOTAL_PRICE
        ,OL_DIST_INFO
        ,I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY
    FROM ( SELECT :next_o_id as O_ID
        ,:w_id AS W_ID
        ,:d_id as D_ID
        ,OL_NUMBER
        ,I_ID
        ,I_QTY
    FROM Table( VALUES
        ( SMALLINT(1) ,:id0 ,:ol_quantity0 )
        ,( SMALLINT(2) ,:id1 ,:ol_quantity1 )
        ,( SMALLINT(3) ,:id2 ,:ol_quantity2 )
        ,( SMALLINT(4) ,:id3 ,:ol_quantity3 )
        ,( SMALLINT(5) ,:id4 ,:ol_quantity4 )
        ,( SMALLINT(6) ,:id5 ,:ol_quantity5 )
        ,( SMALLINT(7) ,:id6 ,:ol_quantity6 )
        ,( SMALLINT(8) ,:id7 ,:ol_quantity7 )
        ,( SMALLINT(9) ,:id8 ,:ol_quantity8 )
        ,( SMALLINT(10) ,:id9 ,:ol_quantity9 )
        ,( SMALLINT(11) ,:id10 ,:ol_quantity10 )
    ) AS X (OL_NUMBER , I_ID , I_QTY

    )
    ) AS ITEM_LIST
    ,TABLE(NEW_OL_LOCAL( I_ID
        ,I_QTY
        ,W_ID
        ,O_ID
        ,D_ID
    )
    ) AS NEW_OL_LOCAL
    WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
)

SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA ,
S_QUANTITY

```



```

FROM NEW TABLE ( INSERT INTO ORDER_LINE
  ( OL_O_ID
  , OL_D_ID
  , OL_W_ID
  , OL_NUMBER
  , OL_I_ID
  , OL_SUPPLY_W_ID
  , OL_DELIVERY_D
  , OL_QUANTITY
  , OL_AMOUNT
  , OL_DIST_INFO
  )
  INCLUDE( I_PRICE INTEGER
  , I_NAME CHAR(24)
  , I_DATA VARCHAR(50)
  , S_DATA VARCHAR(50)
  , S_QUANTITY SMALLINT )
  SELECT O_ID
  , D_ID
  , W_ID
  , OL_NUMBER
  , I_ID
  , I_SUPPLY_W_ID
  , OL_DELIVERY_D
  , I_QTY
  , TOTAL_PRICE
  , OL_DIST_INFO
  , I_PRICE, I_NAME, I_DATA, S_DATA,
S_QUANTITY
  FROM DATA
  ) AS INS
;
EXEC SQL DECLARE ISOL_Local_12 CURSOR FOR
  WITH DATA AS ( SELECT O_ID
  , D_ID
  , W_ID
  , OL_NUMBER
  , I_ID
  , W_ID AS I_SUPPLY_W_ID
  , 0 AS OL_DELIVERY_D
  , I_QTY
  , (I_PRICE * I_QTY) AS TOTAL_PRICE
  , OL_DIST_INFO
  , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

  FROM ( SELECT :next_o_id as O_ID
  , :w_id AS W_ID
  , :d_id as D_ID
  , OL_NUMBER
  , I_ID
  , I_QTY
  FROM Table( VALUES
    ( SMALLINT( 1 ) , :id0 , :ol_quantity0 )
  , ( SMALLINT( 2 ) , :id1 , :ol_quantity1 )
  , ( SMALLINT( 3 ) , :id2 , :ol_quantity2 )
  , ( SMALLINT( 4 ) , :id3 , :ol_quantity3 )
  , ( SMALLINT( 5 ) , :id4 , :ol_quantity4 )
  , ( SMALLINT( 6 ) , :id5 , :ol_quantity5 )
  , ( SMALLINT( 7 ) , :id6 , :ol_quantity6 )
  , ( SMALLINT( 8 ) , :id7 , :ol_quantity7 )
  , ( SMALLINT( 9 ) , :id8 , :ol_quantity8 )
  , ( SMALLINT( 10 ) , :id9 , :ol_quantity9 )
  , ( SMALLINT( 11 ) , :id10 , :ol_quantity10 )
  , ( SMALLINT( 12 ) , :id11 , :ol_quantity11 )
  ) AS X ( OL_NUMBER , I_ID , I_QTY
  )
  ) AS ITEMLIST
  , TABLE(NEW_OL_LOCAL( I_ID

```

```

  , I_QTY
  , W_ID
  , O_ID
  , D_ID
  )
  ) AS NEW_OL_LOCAL
  WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
  )
  SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA ,
S_QUANTITY
  FROM NEW TABLE ( INSERT INTO ORDER_LINE
  ( OL_O_ID
  , OL_D_ID
  , OL_W_ID
  , OL_NUMBER
  , OL_I_ID
  , OL_SUPPLY_W_ID
  , OL_DELIVERY_D
  , OL_QUANTITY
  , OL_AMOUNT
  , OL_DIST_INFO
  )
  INCLUDE( I_PRICE INTEGER
  , I_NAME CHAR(24)
  , I_DATA VARCHAR(50)
  , S_DATA VARCHAR(50)
  , S_QUANTITY SMALLINT )
  SELECT O_ID
  , D_ID
  , W_ID
  , OL_NUMBER
  , I_ID
  , I_SUPPLY_W_ID
  , OL_DELIVERY_D
  , I_QTY
  , TOTAL_PRICE
  , OL_DIST_INFO
  , I_PRICE, I_NAME, I_DATA, S_DATA,
S_QUANTITY
  FROM DATA
  ) AS INS
;
EXEC SQL DECLARE ISOL_Local_13 CURSOR FOR
  WITH DATA AS ( SELECT O_ID
  , D_ID
  , W_ID
  , OL_NUMBER
  , I_ID
  , W_ID AS I_SUPPLY_W_ID
  , 0 AS OL_DELIVERY_D
  , I_QTY
  , (I_PRICE * I_QTY) AS TOTAL_PRICE
  , OL_DIST_INFO
  , I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

  FROM ( SELECT :next_o_id as O_ID
  , :w_id AS W_ID
  , :d_id as D_ID
  , OL_NUMBER
  , I_ID
  , I_QTY
  FROM Table( VALUES
    ( SMALLINT( 1 ) , :id0 , :ol_quantity0 )
  , ( SMALLINT( 2 ) , :id1 , :ol_quantity1 )
  , ( SMALLINT( 3 ) , :id2 , :ol_quantity2 )
  , ( SMALLINT( 4 ) , :id3 , :ol_quantity3 )

```

```

        , ( SMALLINT( 5 ) , :id4 , :ol_quantity4 )
        , ( SMALLINT( 6 ) , :id5 , :ol_quantity5 )
        , ( SMALLINT( 7 ) , :id6 , :ol_quantity6 )
        , ( SMALLINT( 8 ) , :id7 , :ol_quantity7 )
        , ( SMALLINT( 9 ) , :id8 , :ol_quantity8 )
        , ( SMALLINT( 10 ) , :id9 , :ol_quantity9 )
        , ( SMALLINT( 11 ) , :id10 , :ol_quantity10 )
        , ( SMALLINT( 12 ) , :id11 , :ol_quantity11 )
        , ( SMALLINT( 13 ) , :id12 , :ol_quantity12 )
    ) AS X ( OL_NUMBER , I_ID , I_QTY
)
    ) AS ITEMLIST
    , TABLE( NEW_OL_LOCAL( I_ID
        , I_QTY
        , W_ID
        , O_ID
        , D_ID
    )
    ) AS NEW_OL_LOCAL
WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
)
SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA ,
S_QUANTITY
FROM NEW TABLE ( INSERT INTO ORDER_LINE
    ( OL_O_ID
    , OL_D_ID
    , OL_W_ID
    , OL_NUMBER
    , OL_I_ID
    , OL_SUPPLY_W_ID
    , OL_DELIVERY_D
    , OL_QUANTITY
    , OL_AMOUNT
    , OL_DIST_INFO
    )
INCLUDE ( I_PRICE INTEGER
    , I_NAME CHAR(24)
    , I_DATA VARCHAR(50)
    , S_DATA VARCHAR(50)
    , S_QUANTITY SMALLINT )
SELECT O_ID
    , D_ID
    , W_ID
    , OL_NUMBER
    , I_ID
    , I_SUPPLY_W_ID
    , OL_DELIVERY_D
    , I_QTY
    , TOTAL_PRICE
    , OL_DIST_INFO
    , I_PRICE, I_NAME, I_DATA, S_DATA,
S_QUANTITY
FROM DATA
) AS INS
;
EXEC SQL DECLARE ISOL_Local_14 CURSOR FOR

WITH DATA AS ( SELECT O_ID
    , D_ID
    , W_ID
    , OL_NUMBER
    , I_ID
    , W_ID AS I_SUPPLY_W_ID
    , 0 AS OL_DELIVERY_D
    , I_QTY
    , ( I_PRICE * I_QTY ) AS TOTAL_PRICE
    , OL_DIST_INFO

```

```

, I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

FROM ( SELECT :next_o_id as O_ID
    , :w_id AS W_ID
    , :d_id as D_ID
    , OL_NUMBER
    , I_ID
    , I_QTY
FROM Table( VALUES
    ( SMALLINT( 1 ) , :id0 , :ol_quantity0 )
    , ( SMALLINT( 2 ) , :id1 , :ol_quantity1 )
    , ( SMALLINT( 3 ) , :id2 , :ol_quantity2 )
    , ( SMALLINT( 4 ) , :id3 , :ol_quantity3 )
    , ( SMALLINT( 5 ) , :id4 , :ol_quantity4 )
    , ( SMALLINT( 6 ) , :id5 , :ol_quantity5 )
    , ( SMALLINT( 7 ) , :id6 , :ol_quantity6 )
    , ( SMALLINT( 8 ) , :id7 , :ol_quantity7 )
    , ( SMALLINT( 9 ) , :id8 , :ol_quantity8 )
    , ( SMALLINT( 10 ) , :id9 , :ol_quantity9 )
    , ( SMALLINT( 11 ) , :id10 , :ol_quantity10 )
    , ( SMALLINT( 12 ) , :id11 , :ol_quantity11 )
    , ( SMALLINT( 13 ) , :id12 , :ol_quantity12 )
    , ( SMALLINT( 14 ) , :id13 , :ol_quantity13 )
) AS X ( OL_NUMBER , I_ID , I_QTY
)
) AS ITEMLIST
, TABLE( NEW_OL_LOCAL( I_ID
    , I_QTY
    , W_ID
    , O_ID
    , D_ID
)
) AS NEW_OL_LOCAL
WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
)
SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA ,
S_QUANTITY
FROM NEW TABLE ( INSERT INTO ORDER_LINE
    ( OL_O_ID
    , OL_D_ID
    , OL_W_ID
    , OL_NUMBER
    , OL_I_ID
    , OL_SUPPLY_W_ID
    , OL_DELIVERY_D
    , OL_QUANTITY
    , OL_AMOUNT
    , OL_DIST_INFO
    )
INCLUDE ( I_PRICE INTEGER
    , I_NAME CHAR(24)
    , I_DATA VARCHAR(50)
    , S_DATA VARCHAR(50)
    , S_QUANTITY SMALLINT )
SELECT O_ID
    , D_ID
    , W_ID
    , OL_NUMBER
    , I_ID
    , I_SUPPLY_W_ID
    , OL_DELIVERY_D
    , I_QTY
    , TOTAL_PRICE
    , OL_DIST_INFO
    , I_PRICE, I_NAME, I_DATA, S_DATA,
S_QUANTITY
FROM DATA

```

```

) AS INS
;
EXEC SQL DECLARE ISOL_Local_15 CURSOR FOR

WITH DATA AS ( SELECT O_ID
, D_ID
, W_ID
, OL_NUMBER
, I_ID
, W_ID AS I_SUPPLY_W_ID
, 0 AS OL_DELIVERY_D
, I_QTY
, (I_PRICE * I_QTY) AS TOTAL_PRICE
, OL_DIST_INFO
, I_PRICE, I_NAME, I_DATA, S_DATA, S_QUANTITY

FROM ( SELECT :next_o_id as O_ID
, :w_id AS W_ID
, :d_id as D_ID
, OL_NUMBER
, I_ID
, I_QTY
FROM Table( VALUES
( SMALLINT( 1 ) , :id0 , :ol_quantity0 )
, ( SMALLINT( 2 ) , :id1 , :ol_quantity1 )
, ( SMALLINT( 3 ) , :id2 , :ol_quantity2 )
, ( SMALLINT( 4 ) , :id3 , :ol_quantity3 )
, ( SMALLINT( 5 ) , :id4 , :ol_quantity4 )
, ( SMALLINT( 6 ) , :id5 , :ol_quantity5 )
, ( SMALLINT( 7 ) , :id6 , :ol_quantity6 )
, ( SMALLINT( 8 ) , :id7 , :ol_quantity7 )
, ( SMALLINT( 9 ) , :id8 , :ol_quantity8 )
, ( SMALLINT( 10 ) , :id9 , :ol_quantity9 )
, ( SMALLINT( 11 ) , :id10 , :ol_quantity10 )
, ( SMALLINT( 12 ) , :id11 , :ol_quantity11 )
, ( SMALLINT( 13 ) , :id12 , :ol_quantity12 )
, ( SMALLINT( 14 ) , :id13 , :ol_quantity13 )
, ( SMALLINT( 15 ) , :id14 , :ol_quantity14 )
) AS X (OL_NUMBER , I_ID , I_QTY

)
) AS ITEMLIST
, TABLE(NEW_OL_LOCAL( I_ID
, I_QTY
, W_ID
, O_ID
, D_ID
)
) AS NEW_OL_LOCAL
WHERE NEW_OL_LOCAL.I_PRICE IS NOT NULL
)

SELECT I_PRICE , I_NAME , I_DATA , OL_DIST_INFO , S_DATA ,
S_QUANTITY
FROM NEW TABLE ( INSERT INTO ORDER_LINE
( OL_O_ID
, OL_D_ID
, OL_W_ID
, OL_NUMBER
, OL_I_ID
, OL_SUPPLY_W_ID
, OL_DELIVERY_D
, OL_QUANTITY
, OL_AMOUNT
, OL_DIST_INFO
)
INCLUDE ( I_PRICE INTEGER
, I_NAME CHAR(24)
, I_DATA VARCHAR(50)

```

```

, S_DATA VARCHAR(50)
, S_QUANTITY SMALLINT )

SELECT O_ID
, D_ID
, W_ID
, OL_NUMBER
, I_ID
, I_SUPPLY_W_ID
, OL_DELIVERY_D
, I_QTY
, TOTAL_PRICE
, OL_DIST_INFO
, I_PRICE, I_NAME, I_DATA, S_DATA,
S_QUANTITY
FROM DATA
) AS INS
;
// Start processing
in_neword = (struct in_neword_struct *) pin ;
neword = (struct out_neword_struct *) pout ;
#ifdef DEBUGIT
new_debug( neword, in_neword, "SP upon entry");
#endif
// Using I_PRICE == 0 as a flag to the client that the ITEM was not fetched
(hence bad).
for ( inputItemArrayIndex = 0 ; inputItemArrayIndex <
in_neword->s_O_OL_CNT ; inputItemArrayIndex++ )
{
i_priceArray[ inputItemArrayIndex ] = 0 ;
}
neword->deadlocks = -1 ;
retry_tran:

neword->deadlocks++ ;
EXEC SQL
SELECT D_TAX, D_NEXT_O_ID INTO :dist_tax , :next_o_id
FROM OLD TABLE ( UPDATE DISTRICT
SET D_NEXT_O_ID = D_NEXT_O_ID + 1
WHERE D_W_ID = :w_id
AND D_ID = :d_id
) AS OT
;
if ( sqlca.sqlcode != 0 )
{
DLCHK( retry_tran );
sqlerror( NEWORD_SQL, "DISTRICT", __FILE__, __LINE__, &sqlca ) ;
goto ferror;
}
// Invalid I_ID will give a +100, now that we've changed the cursor definitions
// to include a 'WHERE I_PRICE NOT NULL' clause.
#define NEW_CURSOR_OPEN_ERROR \
{ \
if( sqlca.sqlcode != 0 ) \
{ \
goto sql_error ; \
} \
}
#define NEW_CURSOR_ERROR \
{ \
if( sqlca.sqlcode == 0 ) \
{ \
neword->s_O_OL_CNT ++ ; \
} \
else \
if( sqlca.sqlcode == +100 ) \
{ \
break ; \
} \
}

```

```

}
else
  goto sql_error ;
}
if( allLocal )
{
  switch( inputItemCount )
  {
    case 1:
      EXEC SQL OPEN ISOL_Local_1 ;
      NEW_CURSOR_OPEN_ERROR
      for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
      {
        EXEC SQL FETCH ISOL_Local_1 INTO :item_price, :item_name,
:i_data, :stockDistrictInformation, :s_data, :s_quantity ;
        NEW_CURSOR_ERROR
      }
      break ;
    case 2:
      EXEC SQL OPEN ISOL_Local_2 ;
      NEW_CURSOR_OPEN_ERROR
      for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
      {
        EXEC SQL FETCH ISOL_Local_2 INTO :item_price, :item_name,
:i_data, :stockDistrictInformation, :s_data, :s_quantity ;
        NEW_CURSOR_ERROR
      }
      break ;
    case 3:
      EXEC SQL OPEN ISOL_Local_3 ;
      NEW_CURSOR_OPEN_ERROR
      for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
      {
        EXEC SQL FETCH ISOL_Local_3 INTO :item_price, :item_name,
:i_data, :stockDistrictInformation, :s_data, :s_quantity ;
        NEW_CURSOR_ERROR
      }
      break ;
    case 4:
      EXEC SQL OPEN ISOL_Local_4 ;
      NEW_CURSOR_OPEN_ERROR
      for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
      {
        EXEC SQL FETCH ISOL_Local_4 INTO :item_price, :item_name,
:i_data, :stockDistrictInformation, :s_data, :s_quantity ;
        NEW_CURSOR_ERROR
      }
      break ;
    case 5:
      EXEC SQL OPEN ISOL_Local_5 ;
      NEW_CURSOR_OPEN_ERROR
      for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
      {
        EXEC SQL FETCH ISOL_Local_5 INTO :item_price, :item_name,
:i_data, :stockDistrictInformation, :s_data, :s_quantity ;
        NEW_CURSOR_ERROR
      }
      break ;
    case 6:
      EXEC SQL OPEN ISOL_Local_6 ;
      NEW_CURSOR_OPEN_ERROR
      for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
      {
        EXEC SQL FETCH ISOL_Local_6 INTO :item_price, :item_name,
:i_data, :stockDistrictInformation, :s_data, :s_quantity ;
        NEW_CURSOR_ERROR
      }
      break ;
    case 7:
      EXEC SQL OPEN ISOL_Local_7 ;
      NEW_CURSOR_OPEN_ERROR
      for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
      {
        EXEC SQL FETCH ISOL_Local_7 INTO :item_price, :item_name,
:i_data, :stockDistrictInformation, :s_data, :s_quantity ;
        NEW_CURSOR_ERROR
      }
      break ;
    case 8:
      EXEC SQL OPEN ISOL_Local_8 ;
      NEW_CURSOR_OPEN_ERROR
      for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
      {
        EXEC SQL FETCH ISOL_Local_8 INTO :item_price, :item_name,
:i_data, :stockDistrictInformation, :s_data, :s_quantity ;
        NEW_CURSOR_ERROR
      }
      break ;
    case 9:
      EXEC SQL OPEN ISOL_Local_9 ;
      NEW_CURSOR_OPEN_ERROR
      for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
      {
        EXEC SQL FETCH ISOL_Local_9 INTO :item_price, :item_name,
:i_data, :stockDistrictInformation, :s_data, :s_quantity ;
        NEW_CURSOR_ERROR
      }
      break ;
    case 10:
      EXEC SQL OPEN ISOL_Local_10 ;
      NEW_CURSOR_OPEN_ERROR
      for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
      {
        EXEC SQL FETCH ISOL_Local_10 INTO :item_price, :item_name,
:i_data, :stockDistrictInformation, :s_data, :s_quantity ;
        NEW_CURSOR_ERROR
      }
      break ;
    case 11:
      EXEC SQL OPEN ISOL_Local_11 ;
      NEW_CURSOR_OPEN_ERROR
      for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
      {
        EXEC SQL FETCH ISOL_Local_11 INTO :item_price, :item_name,
:i_data, :stockDistrictInformation, :s_data, :s_quantity ;
        NEW_CURSOR_ERROR
      }
      break ;
    case 12:
      EXEC SQL OPEN ISOL_Local_12 ;
      NEW_CURSOR_OPEN_ERROR
      for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
      {

```

```

EXEC SQL FETCH ISOL_Local_12 INTO :item_price, :item_name,
:i_data, :stockDistrictInformation, :s_data, :s_quantity ;
NEW_CURSOR_ERROR
}
break ;
case 13:
EXEC SQL OPEN ISOL_Local_13 ;
NEW_CURSOR_OPEN_ERROR
for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
{
EXEC SQL FETCH ISOL_Local_13 INTO :item_price, :item_name,
:i_data, :stockDistrictInformation, :s_data, :s_quantity ;
NEW_CURSOR_ERROR
}
break ;
case 14:
EXEC SQL OPEN ISOL_Local_14 ;
NEW_CURSOR_OPEN_ERROR
for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
{
EXEC SQL FETCH ISOL_Local_14 INTO :item_price, :item_name,
:i_data, :stockDistrictInformation, :s_data, :s_quantity ;
NEW_CURSOR_ERROR
}
break ;
case 15:
EXEC SQL OPEN ISOL_Local_15 ;
NEW_CURSOR_OPEN_ERROR
for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
{
EXEC SQL FETCH ISOL_Local_15 INTO :item_price, :item_name,
:i_data, :stockDistrictInformation, :s_data, :s_quantity ;
NEW_CURSOR_ERROR
}
break ;

default:
sqlerror(NEWORD_SQL, "Default switch on local
orderline/stock/index", __FILE__, __LINE__, &sqlca);
goto error;
}
}
else
{
switch( inputItemCount )
{
case 1:
EXEC SQL OPEN ISOL_Remote_1 ;
NEW_CURSOR_OPEN_ERROR
for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
{
EXEC SQL FETCH ISOL_Remote_1 INTO :item_price, :item_name,
:i_data, :stockDistrictInformation, :s_data, :s_quantity ;
NEW_CURSOR_ERROR
}
break ;
case 2:
EXEC SQL OPEN ISOL_Remote_2 ;
NEW_CURSOR_OPEN_ERROR
for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
{
EXEC SQL FETCH ISOL_Remote_2 INTO :item_price, :item_name,
:i_data, :stockDistrictInformation, :s_data, :s_quantity ;

```

```

NEW_CURSOR_ERROR
}
break ;
case 3:
EXEC SQL OPEN ISOL_Remote_3 ;
NEW_CURSOR_OPEN_ERROR
for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
{
EXEC SQL FETCH ISOL_Remote_3 INTO :item_price, :item_name,
:i_data, :stockDistrictInformation, :s_data, :s_quantity ;
NEW_CURSOR_ERROR
}
break ;
case 4:
EXEC SQL OPEN ISOL_Remote_4 ;
NEW_CURSOR_OPEN_ERROR
for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
{
EXEC SQL FETCH ISOL_Remote_4 INTO :item_price, :item_name,
:i_data, :stockDistrictInformation, :s_data, :s_quantity ;
NEW_CURSOR_ERROR
}
break ;
case 5:
EXEC SQL OPEN ISOL_Remote_5 ;
NEW_CURSOR_OPEN_ERROR
for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
{
EXEC SQL FETCH ISOL_Remote_5 INTO :item_price, :item_name,
:i_data, :stockDistrictInformation, :s_data, :s_quantity ;
NEW_CURSOR_ERROR
}
break ;
case 6:
EXEC SQL OPEN ISOL_Remote_6 ;
NEW_CURSOR_OPEN_ERROR
for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
{
EXEC SQL FETCH ISOL_Remote_6 INTO :item_price, :item_name,
:i_data, :stockDistrictInformation, :s_data, :s_quantity ;
NEW_CURSOR_ERROR
}
break ;
case 7:
EXEC SQL OPEN ISOL_Remote_7 ;
NEW_CURSOR_OPEN_ERROR
for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
{
EXEC SQL FETCH ISOL_Remote_7 INTO :item_price, :item_name,
:i_data, :stockDistrictInformation, :s_data, :s_quantity ;
NEW_CURSOR_ERROR
}
break ;
case 8:
EXEC SQL OPEN ISOL_Remote_8 ;
NEW_CURSOR_OPEN_ERROR
for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
{
EXEC SQL FETCH ISOL_Remote_8 INTO :item_price, :item_name,
:i_data, :stockDistrictInformation, :s_data, :s_quantity ;
NEW_CURSOR_ERROR
}
}
}
}
}
}

```

```

break ;
case 9:
EXEC SQL OPEN ISOL_Remote_9 ;
NEW_CURSOR_OPEN_ERROR
for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
{
EXEC SQL FETCH ISOL_Remote_9 INTO :item_price, :item_name,
:i_data, :stockDistrictInformation, :s_data, :s_quantity ;
NEW_CURSOR_ERROR
}
break ;
case 10:
EXEC SQL OPEN ISOL_Remote_10 ;
NEW_CURSOR_OPEN_ERROR
for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
{
EXEC SQL FETCH ISOL_Remote_10 INTO :item_price,
:item_name, :i_data, :stockDistrictInformation, :s_data, :s_quantity ;
NEW_CURSOR_ERROR
}
break ;
case 11:
EXEC SQL OPEN ISOL_Remote_11 ;
NEW_CURSOR_OPEN_ERROR
for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
{
EXEC SQL FETCH ISOL_Remote_11 INTO :item_price,
:item_name, :i_data, :stockDistrictInformation, :s_data, :s_quantity ;
NEW_CURSOR_ERROR
}
break ;
case 12:
EXEC SQL OPEN ISOL_Remote_12 ;
NEW_CURSOR_OPEN_ERROR
for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
{
EXEC SQL FETCH ISOL_Remote_12 INTO :item_price,
:item_name, :i_data, :stockDistrictInformation, :s_data, :s_quantity ;
NEW_CURSOR_ERROR
}
break ;
case 13:
EXEC SQL OPEN ISOL_Remote_13 ;
NEW_CURSOR_OPEN_ERROR
for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
{
EXEC SQL FETCH ISOL_Remote_13 INTO :item_price,
:item_name, :i_data, :stockDistrictInformation, :s_data, :s_quantity ;
NEW_CURSOR_ERROR
}
break ;
case 14:
EXEC SQL OPEN ISOL_Remote_14 ;
NEW_CURSOR_OPEN_ERROR
for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
{
EXEC SQL FETCH ISOL_Remote_14 INTO :item_price,
:item_name, :i_data, :stockDistrictInformation, :s_data, :s_quantity ;
NEW_CURSOR_ERROR
}
break ;
case 15:
EXEC SQL OPEN ISOL_Remote_15 ;
NEW_CURSOR_OPEN_ERROR
for ( inputItemArrayIndex = 0 ; inputItemArrayIndex < inputItemCount ;
inputItemArrayIndex++ )
{
EXEC SQL FETCH ISOL_Remote_15 INTO :item_price,
:item_name, :i_data, :stockDistrictInformation, :s_data, :s_quantity ;
NEW_CURSOR_ERROR
}
break ;
default:
sqlerror(NEWORD_SQL, "Default switch on remote
orderline/stock/index", __FILE__, __LINE__, &sqlca);
goto error;
}
}
for ( inputItemArrayIndex = 0 ;
inputItemArrayIndex < in_neword->s_O_OL_CNT // from input
&& i_priceArray[ inputItemArrayIndex ] != 0 ;
inputItemArrayIndex++ )
{
//s_I_NAME, and s_S_QUANTITY already set as output host variables
neword->item[ inputItemArrayIndex ].s_I_PRICE = i_priceArray[
inputItemArrayIndex ] ;

if( is_ORIGINAL( s_dataArray[ inputItemArrayIndex ].data,
s_dataArray[ inputItemArrayIndex ].len )
&& is_ORIGINAL( i_dataArray[ inputItemArrayIndex ].data,
i_dataArray[ inputItemArrayIndex ].len ) )
{
neword->item[ inputItemArrayIndex ].s_brand_generic = 'B';
}
else
{
neword->item[ inputItemArrayIndex ].s_brand_generic = 'G';
}
}
EXEC SQL
SELECT W_TAX, C_DISCOUNT, C_LAST, C_CREDIT
INTO :ware_tax, :c_discount, :c_last, :c_credit

FROM TABLE ( NEW_WH ( :next_o_id
, :w_id
, :d_id
, :c_id
, :o_entry_d
, :inputItemCount
, :allLocal
)
) AS NEW_WH_TABLE
;
if( sqlca.sqlcode == 0 )
{
if( neword->s_O_OL_CNT == in_neword->s_O_OL_CNT )
{
neword->s_transtatus = TRAN_OK ;
EXEC SQL COMMIT;

if( sqlca.sqlcode != 0 )
{
sqlerror(NEWORD_SQL, "COMMIT", __FILE__, __LINE__, &sqlca )
;
goto error;
}
}
else
{

```

```

newword->s_transtatus = INVALID_ITEM ;

EXEC SQL ROLLBACK WORK ;

if ( sqlca.sqlcode != 0 )
{
    newword->s_transtatus = FATAL_SQLERROR;
    sqlerror(NEWORD_SQL, "ROLLBACK FAILED (INVALID ITEM)",
__FILE__, __LINE__, &sqlca);
    // no point in ferror
}
}
else
{
    DLCHK( retry_tran );
    sqlerror(NEWORD_SQL, "NEW_WH", __FILE__, __LINE__, &sqlca);
    goto ferror;
}
/*-----*/
/* Return to client          */
/*-----*/
mexit:
if ( sqlca.sqlcode >= 0 )
{
    storedProcRc = SQLZ_HOLD_PROC ;
}
else
{
    storedProcRc = SQLZ_DISCONNECT_PROC ;
}
#ifdef DEBUGIT
    new_debug( newword, in_newword, "SP prior to return");
#endif
return ( storedProcRc ) ;
sql_error:
{
    char tempstr[ 4096 ] ;
    DLCHK( retry_tran ) ;
    sprintf( tempstr, "inputItemCount=%d, :next_o_id=%d, :d_id=%d,
:w_id=%d", inputItemCount, next_o_id, d_id, w_id ) ;
    sqlerror( NEWORD_SQL, tempstr, __FILE__, __LINE__, &sqlca ) ;
}
ferror:
newword->s_transtatus = FATAL_SQLERROR;
EXEC SQL ROLLBACK WORK;
if ( sqlca.sqlcode != 0 )
{
    sqlerror( NEWORD_SQL, "ROLLBACK FAILED", __FILE__, __LINE__
, &sqlca ) ;
}

goto mexit ;
}
/*
** A little function to search for the string "ORIGINAL" given a string and
** it's length
*/
static unsigned char skip[256] = {8,8,8,8,8,8,8,8, /*0-9*/
    8,8,8,8,8,8,8,8, /*10-19*/
    8,8,8,8,8,8,8,8, /*20-29*/
    8,8,8,8,8,8,8,8, /*30-39*/
    8,8,8,8,8,8,8,8, /*40-49*/
    8,8,8,8,8,8,8,8, /*50-59*/
    8,8,8,8,1,8,8,8, /*60-69*/
    8,4,8,3,8,8,0,8,2,7, /*70-79*/
    8,8,6,8,8,8,8,8,8,8, /*80-89*/
    8,8,8,8,8,8,8,8,8,8, /*90-99*/

```

```

8,8,8,8,8,8,8,8,8, /*100-109*/
8,8,8,8,8,8,8,8,8, /*110-119*/
8,8,8,8,8,8,8,8,8, /*120-129*/
8,8,8,8,8,8,8,8,8, /*130-139*/
8,8,8,8,8,8,8,8,8, /*140-149*/
8,8,8,8,8,8,8,8,8, /*150-159*/
8,8,8,8,8,8,8,8,8, /*160-169*/
8,8,8,8,8,8,8,8,8, /*170-179*/
8,8,8,8,8,8,8,8,8, /*180-189*/
8,8,8,8,8,8,8,8,8, /*190-199*/
8,8,8,8,8,8,8,8,8, /*200-209*/
8,8,8,8,8,8,8,8,8, /*210-219*/
8,8,8,8,8,8,8,8,8, /*220-229*/
8,8,8,8,8,8,8,8,8, /*230-239*/
8,8,8,8,8,8,8,8,8, /*240-249*/
8,8,8,8,8,8,8,8,8, /*250-254*/
static int is_ORIGINAL( char *string, short length )
{
    char *cur_string;
    char *end_string;
    unsigned char *skips;
    int skip_dist;
    int result = 0;
    cur_string = string+7;
    end_string = string + length;
    skips = skip;
    while (cur_string < end_string)
    {
        skip_dist = skips[*cur_string];
        while ( (skip_dist > 0) && (cur_string < end_string) )
        {
            skip_dist = skips[* (cur_string += skip_dist)];
        }
        if (cur_string >= end_string)
            goto exit;
        if ( cur_string[-4] != 'G' )
            goto noMatch;
        if ( memcmp( cur_string-7, "ORIGINAL", 8 ) == 0 )
        {
            result = 1;
            goto exit;
        }
    }
noMatch:
    cur_string += 8;
} /* end while */
exit:
return ( result ) ;
}
// -----
// Order Status SERVER
// -----
#undef w_id
#undef d_id
#undef c_id_input
#undef o_id
#undef o_entry_d
#undef o_carrier_d
#undef c_id
#undef c_first
#undef c_middle
#undef c_last
#undef c_balance
SQL_API_RC order_status_internal( char *pin, char *pout )
{
    struct in_ordstat_struct * in_ordstat = (struct in_ordstat_struct *) pin ;
    struct out_ordstat_struct * ordstat = (struct out_ordstat_struct *) pout ;
    struct sqlca sqlca ;
    EXEC SQL BEGIN DECLARE SECTION;

```

```

// From input values
###sqlint32 w_id;
###short d_id;
sqlint32 c_id_input;
struct s_data_type { short len; char data[ 16 ]; } c_last_input;
// From queries
// From initial query
sqlint32 o_id;
###sqlint32 c_id;
short o_carrier_id;
###sqlint64 o_entry_d;
char c_first[ 16 ];
char c_middle[ 2 ];
###char c_last[ 16 ];
sqlint64 c_balance;
// From cursor
sqlint32 ol_i_id;
sqlint32 ol_supply_w_id;
short ol_quantity;
sqlint32 ol_amount;
sqlint64 ol_delivery_d;
EXEC SQL END DECLARE SECTION;
// NOTE: this varchar would normally live inside the declare section
// but this package already declared the same field higher up. Need the field
// within this scope though.
###struct s_data_type { short len; char data[ 16 ]; } c_last_input;
int storedProcRc;
int itemArrayIndex = 0;
#define w_id in_ordstat->s_W_ID;
#define d_id in_ordstat->s_D_ID;
#define c_id_input in_ordstat->s_C_ID
#define o_id ordstat->s_O_ID
#define o_entry_d ordstat->s_O_ENTRY_D_time
#define o_carrier_id ordstat->s_O_CARRIER_ID
#define c_id ordstat->s_C_ID
#define c_first ordstat->s_C_FIRST
#define c_middle ordstat->s_C_MIDDLE
#define c_last ordstat->s_C_LAST
#define c_balance ordstat->s_C_BALANCE
EXEC SQL DECLARE read_orderline_cur CURSOR FOR
SELECT OL_I_ID, OL_SUPPLY_W_ID, OL_QUANTITY,
OL_AMOUNT, OL_DELIVERY_D
FROM ORDER_LINE
WHERE OL_W_ID = :w_id
AND OL_D_ID = :d_id
AND OL_O_ID = :o_id
FOR FETCH ONLY;
ordstat->deadlocks = -1;
#ifdef DEBUGIT
ord_debug(ordstat, in_ordstat, "SP upon entry");
#endif
retry_tran:
ordstat->deadlocks++;
if ( c_id_input == 0 )
{
c_last_input.len = strlen( in_ordstat->s_C_LAST );
memcpy( c_last_input.data, in_ordstat->s_C_LAST, c_last_input.len );
EXEC SQL
SELECT O_ID, O_CARRIER_ID, O_ENTRY_D, C_BALANCE,
C_FIRST, C_MIDDLE, C_ID
INTO :o_id, :o_carrier_id, :o_entry_d, :c_balance, :c_first, :c_middle,
:c_id

FROM TABLE ( ORD_C_LAST( :w_id
, :d_id
, :c_last_input
)
) AS ORD_C_LAST
;
}
else
{
EXEC SQL
SELECT O_ID, O_CARRIER_ID, O_ENTRY_D, C_BALANCE,
C_FIRST, C_MIDDLE, C_LAST
INTO :o_id, :o_carrier_id, :o_entry_d, :c_balance, :c_first, :c_middle,
:c_last

FROM TABLE ( ORD_C_ID( :w_id
, :d_id
, :c_id_input
)
) AS ORD_C_ID
;
}
if ( sqlca.sqlcode != 0 )
{
DLCHK( retry_tran );
sqlerror( ORDSTAT_SQL, "READ CUST and ORDERS", __FILE__,
__LINE__, &sqlca );
goto ferror;
}
/*-----*/
/* Read ORDER_LINES */
/*-----*/
EXEC SQL OPEN read_orderline_cur;
if ( sqlca.sqlcode != 0 )
{
DLCHK( retry_tran );
sqlerror(ORDSTAT_SQL, "OPEN CURSOR read_orderline_cur",
__FILE__, __LINE__, &sqlca );
goto ferror;
}
itemArrayIndex = 0;
{
do
{
EXEC SQL FETCH read_orderline_cur
INTO :ol_i_id, :ol_supply_w_id, :ol_quantity, :ol_amount,
:ol_delivery_d;
if ( sqlca.sqlcode == 0 )
{
ordstat->item[ itemArrayIndex ].s_OL_I_ID = ol_i_id;
ordstat->item[ itemArrayIndex ].s_OL_SUPPLY_W_ID =
ol_supply_w_id;
ordstat->item[ itemArrayIndex ].s_OL_QUANTITY = ol_quantity;
ordstat->item[ itemArrayIndex ].s_OL_AMOUNT = ol_amount;
ordstat->item[ itemArrayIndex ].s_OL_DELIVERY_D_time =
ol_delivery_d;

itemArrayIndex++;
}
else
if (sqlca.sqlcode < 0)
{
DLCHK( retry_tran );
sqlerror( ORDSTAT_SQL, "FETCH CURSOR read_orderline_cur",
__FILE__, __LINE__, &sqlca );
goto ferror;
}
}
while ( sqlca.sqlcode == 0 );
}
ordstat->s_ol_cnt = itemArrayIndex;
EXEC SQL COMMIT;
if ( sqlca.sqlcode == 0 )

```



```

{
  ordstat->s_transtatus = TRAN_OK ;
}
else
{
  DLCHK( retry_tran );
  sqlerror(ORDSTAT_SQL, "COMMIT", __FILE__, __LINE__, &sqlca);
  goto ferror ;
}
}
mexit:
if ( sqlca.sqlcode >= 0 )
{
  storedProcRc = SQLZ_HOLD_PROC ;
}
else
{
  storedProcRc = SQLZ_DISCONNECT_PROC ;
}
}
#endif
ord_debug(ordstat, in_ordstat, "SP prior to return");
#endif
return ( storedProcRc ) ;
ferror:
ordstat->s_transtatus = FATAL_SQLERROR ;
EXEC SQL ROLLBACK WORK ;
if ( sqlca.sqlcode != 0 )
{
  sqlerror(ORDSTAT_SQL, "ROLLBACK FAILED", __FILE__, __LINE__,
&sqlca);
}
goto mexit;
}
// -----
// Delivery SERVER
// -----

#undef d_id
#undef c_id
#undef w_id
#undef o_carrier_id
#undef ol_delivery_d
SQL_API_RC delivery_internal ( char * pin, char * pout )
{
  struct in_delivery_struct * in_delivery = (struct in_delivery_struct *) pin ;
  struct out_delivery_struct * delivery = (struct out_delivery_struct *) pout ;
  struct sqlca sqlca ;
  int storedProcRc ;
  short district_id ;
  sqlint32 customer_id ;

EXEC SQL BEGIN DECLARE SECTION;
// input
###sqlint32 w_id ;
###short d_id ;
###sqlint32 c_id ;
###short o_carrier_id ;
###sqlint64 ol_delivery_d ;
// output
short no_o_id_indicator = 0 ;
sqlint32 no_o_id ;
EXEC SQL END DECLARE SECTION;
#define d_id district_id
#define c_id customer_id
#define w_id in_delivery->s_W_ID
#define o_carrier_id in_delivery->s_O_CARRIER_ID
#define ol_delivery_d in_delivery->s_O_DELIVERY_D_time
delivery->deadlocks = -1 ;

```

```

#endif
del_debug( delivery, in_delivery, "SP upon entry");
#endif
d_id = 1;
retry_tran:
delivery->deadlocks++;
for ( ; d_id <= DISTRICTS_PER_WAREHOUSE ; d_id++ )
{
  no_o_id = 0 ;
  no_o_id_indicator = 0 ;
  EXEC SQL BEGIN COMPOUND NOT ATOMIC STATIC
  SELECT O_ID

  INTO :no_o_id :no_o_id_indicator

  FROM TABLE ( DEL(:w_id, :d_id, :o_carrier_id, :ol_delivery_d )
AS T ;

  COMMIT ;
  END COMPOUND ;

if ( sqlca.sqlcode == 0 )
{
  /* Refer to clause 2.7.4.2, bullet 3 in spec.*/
  /* Need to report if more than 1 or 1% of */
  /* no_o_id will remain 0 if null returned, so just treat the same way */
  delivery->s_O_ID[ d_id - 1 ] = no_o_id ;
}
else
{
  DLCHK( retry_tran );

  sqlerror( DELIVERY_SQL, "DELIVERY", __FILE__, __LINE__,
&sqlca);
  goto ferror ;
}
}
delivery->s_transtatus = TRAN_OK ;
mexit:
if ( sqlca.sqlcode >= 0 )
{
  storedProcRc = SQLZ_HOLD_PROC ;
}
else
{
  storedProcRc = SQLZ_DISCONNECT_PROC ;
}
#endif
del_debug( delivery, in_delivery, "SP prior to return");
#endif
return ( storedProcRc ) ;
ferror:
delivery->s_transtatus = FATAL_SQLERROR ;
EXEC SQL ROLLBACK WORK ;
if ( sqlca.sqlcode != 0 )
{
  sqlerror( DELIVERY_SQL, "ROLLBACK FAILED", __FILE__, __LINE__,
&sqlca ) ;
}
goto mexit ;
}
// -----
// Stored Procedure Stubs
// -----
SQL_API_RC SQL_API_FN news( char *pin, char *pout )
{

```

```

    return new_order_internal( pin, pout );
}
SQL_API_RC SQL_API_FN ords( char *pin, char *pout )
{
    return order_status_internal( pin, pout );
}
SQL_API_RC SQL_API_FN dels ( char * pin, char * pout )
{
    return delivery_internal( pin, pout );
}

```

Src.Srv/uncat_func.ddl

```

-----
-- Licensed Materials - Property of IBM
--
-- Governed under the terms of the International
-- License Agreement for Non-Warranted Sample Code.
--
-- (C) COPYRIGHT International Business Machines Corp. 1996 - 2004
-- All Rights Reserved.
--
-- US Government Users Restricted Rights - Use, duplication or
-- disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
-----
-- uncat-func.ddl - Drop table function DDL
--
-- STOCK LEVEL
DROP SPECIFIC FUNCTION STOCK_LEVEL $
-- DELIVERY
DROP SPECIFIC FUNCTION DELIVERY $
-- ORDER STATUS
DROP SPECIFIC FUNCTION ORD_C_LAST $
DROP SPECIFIC FUNCTION ORD_C_ID $
-- PAYMENT
DROP SPECIFIC FUNCTION PAY_C_LAST $
DROP SPECIFIC FUNCTION PAY_C_ID $
-- NEW ORDER
DROP SPECIFIC FUNCTION NEW_OL_ALL $
DROP SPECIFIC FUNCTION NEW_OL_LOCAL $
DROP SPECIFIC FUNCTION NEW_WH $

```

Src.Srv/uncat_proc.ddl

```

DROP PROCEDURE news
    (varchar(270),varchar(662));
DROP PROCEDURE news;

DROP PROCEDURE pays;
DROP PROCEDURE ords
    (varchar(42),varchar(446));
DROP PROCEDURE ords;
DROP PROCEDURE dels
    (varchar(22),varchar(50));
DROP PROCEDURE dels;
DROP PROCEDURE stks
    (varchar(18),varchar(14));
DROP PROCEDURE stks;

```

Src.Srv/rpctpcc.def

```

LIBRARY rpctpcc
DESCRIPTION "Library of TPC-C Transactions (Stored Procedures)"
EXPORTS
news

```

```

ords
dels

```

utils/EXPLAIN.ddl

```

-- *- sql *-
--
-- Sample DDL to create Explain tables for Version 5.0
--
-- -> assumes db2start issued
-- -> assumes connection to a database exists
-- -> assumes called by "db2 -tf EXPLAIN.DDL"
--
--
-- To remind users how to use this file!
--
ECHO          ;
ECHO ***** IMPORTANT ***** ;
ECHO          ;
ECHO USAGE: db2 -tf EXPLAIN.DDL ;
ECHO          ;
ECHO ***** IMPORTANT ***** ;
ECHO          ;
ECHO          ;
--
-- Set autocommit off
--
UPDATE COMMAND OPTIONS USING C OFF;
--
-- EXPLAIN INSTANCE
--
-- (must be defined first due to referential integrity defintitions)
--
CREATE TABLE EXPLAIN_INSTANCE ( EXPLAIN_REQUESTER
VARCHAR(128) NOT NULL,
    EXPLAIN_TIME    TIMESTAMP NOT NULL,
    SOURCE_NAME     VARCHAR(128) NOT NULL,
    SOURCE_SCHEMA   VARCHAR(128) NOT NULL,
    SOURCE_VERSION  VARCHAR(64) NOT NULL,
    EXPLAIN_OPTION CHAR(1)  NOT NULL,
    SNAPSHOT_TAKEN CHAR(1)  NOT NULL,
    DB2_VERSION    CHAR(7)  NOT NULL,
    SQL_TYPE       CHAR(1)  NOT NULL,
    QUERYOPT       INTEGER  NOT NULL,
    BLOCK          CHAR(1)  NOT NULL,
    ISOLATION      CHAR(2)  NOT NULL,
    BUFFPAGE       INTEGER  NOT NULL,
    AVG_APPLS     INTEGER  NOT NULL,
    SORTHEAP       INTEGER  NOT NULL,
    LOCKLIST       INTEGER  NOT NULL,
    MAXLOCKS      SMALLINT NOT NULL,
    LOCKS_AVAIL    INTEGER  NOT NULL,
    CPU_SPEED      DOUBLE   NOT NULL,
    REMARKS        VARCHAR(254),
    DBHEAP         INTEGER  NOT NULL,
    COMM_SPEED     DOUBLE   NOT NULL,
    PARALLELISM    CHAR(2)  NOT NULL,
    DATAJOINER    CHAR(1)  NOT NULL,
    PRIMARY KEY (EXPLAIN_REQUESTER,
        EXPLAIN_TIME,
        SOURCE_NAME,
        SOURCE_SCHEMA,
        SOURCE_VERSION))
IN USERSPACE1
INDEX IN USERSPACE1;
--
-- EXPLAIN_STATEMENT

```

```

--
CREATE TABLE EXPLAIN_STATEMENT ( EXPLAIN_REQUESTER
VARCHAR(128) NOT NULL,
    EXPLAIN_TIME    TIMESTAMP NOT NULL,
    SOURCE_NAME     VARCHAR(128) NOT NULL,
    SOURCE_SCHEMA   VARCHAR(128) NOT NULL,
    SOURCE_VERSION  VARCHAR(64) NOT NULL,
    EXPLAIN_LEVEL   CHAR(1) NOT NULL,
    STMTNO         INTEGER NOT NULL,
    SECTNO         INTEGER NOT NULL,
    QUERYNO        INTEGER NOT NULL,
    QUERYTAG       CHAR(20) NOT NULL,
    STATEMENT_TYPE CHAR(2) NOT NULL,
    UPDATABLE      CHAR(1) NOT NULL,
    DELETABLE      CHAR(1) NOT NULL,
    TOTAL_COST     DOUBLE NOT NULL,
    STATEMENT_TEXT CLOB(2M) NOT NULL NOT
LOGGED,
    SNAPSHOT       BLOB(10M) NOT LOGGED,
    QUERY_DEGREE   INTEGER NOT NULL,
    PRIMARY KEY (EXPLAIN_REQUESTER,
        EXPLAIN_TIME,
        SOURCE_NAME,
        SOURCE_SCHEMA,
        SOURCE_VERSION,
        EXPLAIN_LEVEL,
        STMTNO,
        SECTNO),
    FOREIGN KEY (EXPLAIN_REQUESTER,
        EXPLAIN_TIME,
        SOURCE_NAME,
        SOURCE_SCHEMA,
        SOURCE_VERSION)
    REFERENCES EXPLAIN_INSTANCE
    ON DELETE CASCADE)
IN USERSPACE1
INDEX IN USERSPACE1;
--
-- EXPLAIN_ARGUMENTS
--
CREATE TABLE EXPLAIN_ARGUMENT ( EXPLAIN_REQUESTER
VARCHAR(128) NOT NULL,
    EXPLAIN_TIME    TIMESTAMP NOT NULL,
    SOURCE_NAME     VARCHAR(128) NOT NULL,
    SOURCE_SCHEMA   VARCHAR(128) NOT NULL,
    SOURCE_VERSION  VARCHAR(64) NOT NULL,
    EXPLAIN_LEVEL   CHAR(1) NOT NULL,
    STMTNO         INTEGER NOT NULL,
    SECTNO         INTEGER NOT NULL,
    OPERATOR_ID     INTEGER NOT NULL,
    ARGUMENT_TYPE   CHAR(8) NOT NULL,
    ARGUMENT_VALUE  VARCHAR(1024),
    LONG_ARGUMENT_VALUE CLOB(2M) NOT
LOGGED,
    FOREIGN KEY (EXPLAIN_REQUESTER,
        EXPLAIN_TIME,
        SOURCE_NAME,
        SOURCE_SCHEMA,
        SOURCE_VERSION,
        EXPLAIN_LEVEL,
        STMTNO,
        SECTNO)
    REFERENCES EXPLAIN_STATEMENT
    ON DELETE CASCADE)
IN USERSPACE1
INDEX IN USERSPACE1;
--
-- EXPLAIN_OBJECT

```

```

--
CREATE TABLE EXPLAIN_OBJECT ( EXPLAIN_REQUESTER
VARCHAR(128) NOT NULL,
    EXPLAIN_TIME    TIMESTAMP NOT NULL,
    SOURCE_NAME     VARCHAR(128) NOT NULL,
    SOURCE_SCHEMA   VARCHAR(128) NOT NULL,
    SOURCE_VERSION  VARCHAR(64) NOT NULL,
    EXPLAIN_LEVEL   CHAR(1) NOT NULL,
    STMTNO         INTEGER NOT NULL,
    SECTNO         INTEGER NOT NULL,
    OBJECT_SCHEMA   VARCHAR(128) NOT NULL,
    OBJECT_NAME     VARCHAR(128) NOT NULL,
    OBJECT_TYPE     CHAR(2) NOT NULL,
    CREATE_TIME     TIMESTAMP,
    STATISTICS_TIME TIMESTAMP,
    COLUMN_COUNT    SMALLINT NOT NULL,
    ROW_COUNT       BIGINT NOT NULL,
    WIDTH           INTEGER NOT NULL,
    PAGES           INTEGER NOT NULL,
    DISTINCT        CHAR(1) NOT NULL,
    TABLESPACE_NAME VARCHAR(128),
    OVERHEAD        DOUBLE NOT NULL,
    TRANSFER_RATE   DOUBLE NOT NULL,
    PREFETCHSIZE    INTEGER NOT NULL,
    EXTENTSIZE      INTEGER NOT NULL,
    CLUSTER         DOUBLE NOT NULL,
    NLEAF           INTEGER NOT NULL,
    NLEVELS         INTEGER NOT NULL,
    FULLKEYCARD     BIGINT NOT NULL,
    OVERFLOW        INTEGER NOT NULL,
    FIRSTKEYCARD    BIGINT NOT NULL,
    FIRST2KEYCARD   BIGINT NOT NULL,
    FIRST3KEYCARD   BIGINT NOT NULL,
    FIRST4KEYCARD   BIGINT NOT NULL,
    SEQUENTIAL_PAGES INTEGER NOT NULL,
    DENSITY         INTEGER NOT NULL,
    STATS_SRC       CHAR(1) NOT NULL,
    AVERAGE_SEQUENCE_GAP DOUBLE NOT
NULL,
    AVERAGE_SEQUENCE_FETCH_GAP DOUBLE NOT
NULL,
    AVERAGE_SEQUENCE_PAGES DOUBLE NOT
NULL,
    AVERAGE_SEQUENCE_FETCH_PAGES DOUBLE
NOT NULL,
    AVERAGE_RANDOM_PAGES DOUBLE NOT
NULL,
    AVERAGE_RANDOM_FETCH_PAGES DOUBLE NOT
NULL,
    NUMRIDS         BIGINT NOT NULL,
    NUMRIDS_DELETED BIGINT NOT NULL,
    NUM_EMPTY_LEAFS BIGINT NOT NULL,
    ACTIVE_BLOCKS   BIGINT NOT NULL,
    FOREIGN KEY (EXPLAIN_REQUESTER,
        EXPLAIN_TIME,
        SOURCE_NAME,
        SOURCE_SCHEMA,
        SOURCE_VERSION,
        EXPLAIN_LEVEL,
        STMTNO,
        SECTNO)
    REFERENCES EXPLAIN_STATEMENT
    ON DELETE CASCADE)
IN USERSPACE1
INDEX IN USERSPACE1;
--
-- EXPLAIN_OPERATOR
--

```

```

CREATE TABLE EXPLAIN_OPERATOR ( EXPLAIN_REQUESTER
VARCHAR(128) NOT NULL,
    EXPLAIN_TIME    TIMESTAMP NOT NULL,
    SOURCE_NAME     VARCHAR(128) NOT NULL,
    SOURCE_SCHEMA   VARCHAR(128) NOT NULL,
    SOURCE_VERSION  VARCHAR(64) NOT NULL,
    EXPLAIN_LEVEL   CHAR(1) NOT NULL,
    STMTNO         INTEGER NOT NULL,
    SECTNO         INTEGER NOT NULL,
    OPERATOR_ID    INTEGER NOT NULL,
    OPERATOR_TYPE  CHAR(6) NOT NULL,
    TOTAL_COST     DOUBLE NOT NULL,
    IO_COST        DOUBLE NOT NULL,
    CPU_COST       DOUBLE NOT NULL,
    FIRST_ROW_COST DOUBLE NOT NULL,
    RE_TOTAL_COST  DOUBLE NOT NULL,
    RE_IO_COST     DOUBLE NOT NULL,
    RE_CPU_COST    DOUBLE NOT NULL,
    COMM_COST      DOUBLE NOT NULL,
    FIRST_COMM_COST DOUBLE NOT NULL,
    BUFFERS        DOUBLE NOT NULL,
    REMOTE_TOTAL_COST DOUBLE NOT NULL,
    REMOTE_COMM_COST DOUBLE NOT NULL,
    FOREIGN KEY (EXPLAIN_REQUESTER,
                EXPLAIN_TIME,
                SOURCE_NAME,
                SOURCE_SCHEMA,
                SOURCE_VERSION,
                EXPLAIN_LEVEL,
                STMTNO,
                SECTNO)
    REFERENCES EXPLAIN_STATEMENT
    ON DELETE CASCADE)

IN USERSPACE1
INDEX IN USERSPACE1;
--
-- EXPLAIN_PREDICATE
--
CREATE TABLE EXPLAIN_PREDICATE ( EXPLAIN_REQUESTER
VARCHAR(128) NOT NULL,
    EXPLAIN_TIME    TIMESTAMP NOT NULL,
    SOURCE_NAME     VARCHAR(128) NOT NULL,
    SOURCE_SCHEMA   VARCHAR(128) NOT NULL,
    SOURCE_VERSION  VARCHAR(64) NOT NULL,
    EXPLAIN_LEVEL   CHAR(1) NOT NULL,
    STMTNO         INTEGER NOT NULL,
    SECTNO         INTEGER NOT NULL,
    OPERATOR_ID    INTEGER NOT NULL,
    PREDICATE_ID   INTEGER NOT NULL,
    HOW_APPLIED    CHAR(5) NOT NULL,
    WHEN_EVALUATED CHAR(3) NOT NULL,
    RELOP_TYPE     CHAR(2) NOT NULL,
    SUBQUERY       CHAR(1) NOT NULL,
    FILTER_FACTOR  DOUBLE NOT NULL,
    PREDICATE_TEXT CLOB(2M) NOT LOGGED,
    FOREIGN KEY (EXPLAIN_REQUESTER,
                EXPLAIN_TIME,
                SOURCE_NAME,
                SOURCE_SCHEMA,
                SOURCE_VERSION,
                EXPLAIN_LEVEL,
                STMTNO,
                SECTNO)
    REFERENCES EXPLAIN_STATEMENT
    ON DELETE CASCADE)

IN USERSPACE1
INDEX IN USERSPACE1;
--

```

```

-- EXPLAIN_STREAM
--
CREATE TABLE EXPLAIN_STREAM ( EXPLAIN_REQUESTER
VARCHAR(128) NOT NULL,
    EXPLAIN_TIME    TIMESTAMP NOT NULL,
    SOURCE_NAME     VARCHAR(128) NOT NULL,
    SOURCE_SCHEMA   VARCHAR(128) NOT NULL,
    SOURCE_VERSION  VARCHAR(64) NOT NULL,
    EXPLAIN_LEVEL   CHAR(1) NOT NULL,
    STMTNO         INTEGER NOT NULL,
    SECTNO         INTEGER NOT NULL,
    STREAM_ID       INTEGER NOT NULL,
    SOURCE_TYPE     CHAR(1) NOT NULL,
    SOURCE_ID       INTEGER NOT NULL,
    TARGET_TYPE    CHAR(1) NOT NULL,
    TARGET_ID      INTEGER NOT NULL,
    OBJECT_SCHEMA  VARCHAR(128),
    OBJECT_NAME    VARCHAR(128),
    STREAM_COUNT   DOUBLE NOT NULL,
    COLUMN_COUNT   SMALLINT NOT NULL,
    PREDICATE_ID   INTEGER NOT NULL,
    COLUMN_NAMES   CLOB(2M) NOT LOGGED,
    PMID          SMALLINT NOT NULL,
    SINGLE_NODE    CHAR(5),
    PARTITION_COLUMNS CLOB(2M) NOT LOGGED,
    FOREIGN KEY (EXPLAIN_REQUESTER,
                EXPLAIN_TIME,
                SOURCE_NAME,
                SOURCE_SCHEMA,
                SOURCE_VERSION,
                EXPLAIN_LEVEL,
                STMTNO,
                SECTNO)
    REFERENCES EXPLAIN_STATEMENT
    ON DELETE CASCADE)

IN USERSPACE1
INDEX IN USERSPACE1;
--
-- ADVISE TABLES
--
-- ADVISE_INSTANCE
--
-- (must be defined first due to referential integrity defintiions)
--
CREATE TABLE ADVISE_INSTANCE (
    START_TIME    TIMESTAMP NOT NULL WITH DEFAULT
CURRENT TIMESTAMP,
    END_TIME      TIMESTAMP NOT NULL WITH DEFAULT
CURRENT TIMESTAMP,
    MODE          VARCHAR(4) NOT NULL WITH DEFAULT ",
    WKLD_COMPRESSION CHAR(4) NOT NULL WITH DEFAULT
'NONE',
    STATUS        CHAR(9) NOT NULL WITH DEFAULT ",
    PRIMARY KEY (START_TIME))

IN USERSPACE1
INDEX IN USERSPACE1;
--
-- ADVISE_INDEX
--
CREATE TABLE ADVISE_INDEX(
    EXPLAIN_REQUESTER VARCHAR(128) NOT NULL WITH
DEFAULT ",
    EXPLAIN_TIME    TIMESTAMP NOT NULL WITH DEFAULT
CURRENT TIMESTAMP,
    SOURCE_NAME     VARCHAR(128) NOT NULL WITH DEFAULT ",

```

```

SOURCE_SCHEMA VARCHAR(128) NOT NULL WITH DEFAULT
",
SOURCE_VERSION VARCHAR(64) NOT NULL WITH DEFAULT
",
EXPLAIN_LEVEL CHAR(1) NOT NULL WITH DEFAULT ",
STMTNO INTEGER NOT NULL WITH DEFAULT 0,
SECTNO INTEGER NOT NULL WITH DEFAULT 0,
QUERYNO INTEGER NOT NULL WITH DEFAULT 0,
QUERYTAG CHAR(20) NOT NULL WITH DEFAULT ",
NAME VARCHAR(128) NOT NULL,
CREATOR VARCHAR(128) NOT NULL WITH DEFAULT ",
TBNAME VARCHAR(128) NOT NULL,
TBCREATOR VARCHAR(128) NOT NULL WITH DEFAULT ",
COLNAMES CLOB(2M) NOT NULL,
UNIQUERULE CHAR(1) NOT NULL WITH DEFAULT ",
COLCOUNT SMALLINT NOT NULL WITH DEFAULT 0,
IID SMALLINT NOT NULL WITH DEFAULT 0,
NLEAF INTEGER NOT NULL WITH DEFAULT 0,
NLEVELS SMALLINT NOT NULL WITH DEFAULT 0,
FIRSTKEYCARD BIGINT NOT NULL WITH DEFAULT 0,
FULLKEYCARD BIGINT NOT NULL WITH DEFAULT 0,
CLUSTERATIO SMALLINT NOT NULL WITH DEFAULT 0,
CLUSTERFACTOR DOUBLE NOT NULL WITH DEFAULT 0,
USERDEFINED SMALLINT NOT NULL WITH DEFAULT 0,
SYSTEM_REQUIRED SMALLINT NOT NULL WITH DEFAULT
0,
CREATE_TIME TIMESTAMP NOT NULL WITH DEFAULT
CURRENT_TIMESTAMP,
STATS_TIME TIMESTAMP WITH DEFAULT CURRENT
TIMESTAMP,
PAGE_FETCH_PAIRS VARCHAR(254) NOT NULL WITH
DEFAULT ",
REMARKS VARCHAR(254) WITH DEFAULT ",
DEFINER VARCHAR(128) NOT NULL WITH DEFAULT ",
CONVERTED CHAR(1) NOT NULL WITH DEFAULT ",
SEQUENTIAL_PAGES INTEGER NOT NULL WITH DEFAULT 0,
DENSITY INTEGER NOT NULL WITH DEFAULT 0,
FIRST2KEYCARD BIGINT NOT NULL WITH DEFAULT 0,
FIRST3KEYCARD BIGINT NOT NULL WITH DEFAULT 0,
FIRST4KEYCARD BIGINT NOT NULL WITH DEFAULT 0,
PCTFREE SMALLINT NOT NULL WITH DEFAULT -1,
UNIQUE_COLCOUNT SMALLINT NOT NULL WITH DEFAULT
-1,
MINPCTUSED SMALLINT NOT NULL WITH DEFAULT 0,
REVERSE_SCANS CHAR(1) NOT NULL WITH DEFAULT 'N',
USE_INDEX CHAR(1),
CREATION_TEXT CLOB(2M) NOT NULL NOT LOGGED WITH
DEFAULT ",
PACKED_DESC BLOB(1M) NOT LOGGED,
RUN_ID TIMESTAMP,
INDEXTYPE VARCHAR(4) NOT NULL WITH DEFAULT ",
EXISTS CHAR(1) NOT NULL WITH DEFAULT 'N',
RIDTOBLOCK CHAR(1) NOT NULL WITH DEFAULT 'N',
FOREIGN KEY (RUN_ID)
REFERENCES ADVISE_INSTANCE (START_TIME)
ON DELETE CASCADE)
IN USERSPACE1
INDEX IN USERSPACE1;
--
-- ADVISE_WORKLOAD
--
CREATE TABLE ADVISE_WORKLOAD (
WORKLOAD_NAME CHAR(128) NOT NULL WITH DEFAULT
'WK0',
STATEMENT_NO INTEGER NOT NULL WITH DEFAULT 1,
STATEMENT_TEXT CLOB(2M) NOT NULL NOT LOGGED,
STATEMENT_TAG VARCHAR(256) NOT NULL WITH DEFAULT "
",
FREQUENCY INTEGER NOT NULL WITH DEFAULT 1,
IMPORTANCE DOUBLE NOT NULL WITH DEFAULT 1,
WEIGHT DOUBLE NOT NULL WITH DEFAULT 1,
COST_BEFORE DOUBLE,
COST_AFTER DOUBLE,
COMPILABLE CHAR(17))
IN USERSPACE1
INDEX IN USERSPACE1;
--
-- ADVISE_MQT
--
CREATE TABLE ADVISE_MQT (
EXPLAIN_REQUESTER VARCHAR(128) NOT NULL WITH
DEFAULT ",
EXPLAIN_TIME TIMESTAMP NOT NULL WITH DEFAULT
CURRENT_TIMESTAMP,
SOURCE_NAME VARCHAR(128) NOT NULL WITH DEFAULT ",
SOURCE_SCHEMA VARCHAR(128) NOT NULL WITH DEFAULT
",
SOURCE_VERSION VARCHAR(64) NOT NULL WITH DEFAULT
",
EXPLAIN_LEVEL CHAR(1) NOT NULL WITH DEFAULT ",
STMTNO INTEGER NOT NULL WITH DEFAULT 0,
SECTNO INTEGER NOT NULL WITH DEFAULT 0,
NAME VARCHAR(128) NOT NULL,
CREATOR VARCHAR(128) NOT NULL WITH DEFAULT ",
IID SMALLINT NOT NULL WITH DEFAULT 0,
CREATE_TIME TIMESTAMP NOT NULL WITH DEFAULT
CURRENT_TIMESTAMP,
STATS_TIME TIMESTAMP WITH DEFAULT CURRENT
TIMESTAMP,
NUMROWS DOUBLE NOT NULL WITH DEFAULT 0,
NUMCOLS SMALLINT NOT NULL WITH DEFAULT 0,
ROWSIZE DOUBLE NOT NULL WITH DEFAULT 0,
BENEFIT FLOAT NOT NULL WITH DEFAULT 0.0,
USE_MQT CHAR(1),
MQT_SOURCE CHAR(1),
QUERY_TEXT CLOB(2M) NOT NULL NOT LOGGED WITH
DEFAULT ",
CREATION_TEXT CLOB(2M) NOT NULL NOT LOGGED WITH
DEFAULT ",
SAMPLE_TEXT CLOB(2M) NOT NULL NOT LOGGED WITH
DEFAULT ",
COLSTATS CLOB(2M) NOT NULL NOT LOGGED WITH
DEFAULT ",
EXTRA_INFO BLOB(2M) NOT NULL NOT LOGGED with
default BLOB(""),
TBSPACE VARCHAR(128) NOT NULL WITH DEFAULT ",
RUN_ID TIMESTAMP,
REFRESH_TYPE CHAR(1) NOT NULL WITH DEFAULT ",
EXISTS CHAR(1) NOT NULL WITH DEFAULT 'N',
FOREIGN KEY (RUN_ID)
REFERENCES ADVISE_INSTANCE (START_TIME)
ON DELETE CASCADE)
IN USERSPACE1
INDEX IN USERSPACE1;
--
-- ADVISE_PARTITION
--
CREATE TABLE ADVISE_PARTITION (
EXPLAIN_REQUESTER VARCHAR(128) NOT NULL WITH
DEFAULT ",
EXPLAIN_TIME TIMESTAMP NOT NULL WITH DEFAULT
CURRENT_TIMESTAMP,
SOURCE_NAME VARCHAR(128) NOT NULL WITH DEFAULT ",
SOURCE_SCHEMA VARCHAR(128) NOT NULL WITH DEFAULT
",

```

```

SOURCE_VERSION VARCHAR(64) NOT NULL WITH DEFAULT
",
EXPLAIN_LEVEL CHAR(1) NOT NULL WITH DEFAULT ",
STMTNO INTEGER NOT NULL WITH DEFAULT 0,
SECTNO INTEGER NOT NULL WITH DEFAULT 0,
QUERYNO INTEGER NOT NULL WITH DEFAULT 0,
QUERYTAG CHAR(20) NOT NULL WITH DEFAULT ",
TBNAME VARCHAR(128) NOT NULL,
TBCREATOR VARCHAR(128) NOT NULL WITH DEFAULT ",
PMID SMALLINT NOT NULL,
TBSPACE VARCHAR(128) NOT NULL WITH DEFAULT ",
COLNAMES CLOB(2M) NOT NULL NOT LOGGED WITH
DEFAULT ",
COLCOUNT SMALLINT NOT NULL WITH DEFAULT 0,
REPLICATE CHAR(1) NOT NULL WITH DEFAULT 'N',
COST DOUBLE NOT NULL,
USEIT CHAR(1),
RUN_ID TIMESTAMP,
FOREIGN KEY(RUN_ID)
REFERENCES ADVISE_INSTANCE (START_TIME)
ON DELETE CASCADE)
IN USERSPACE1
INDEX IN USERSPACE1;
--
-- ADVISE_TABLE
--
CREATE TABLE ADVISE_TABLE (
RUN_ID TIMESTAMP,
TABLE_NAME VARCHAR(128) NOT NULL,
TABLE_SCHEMA VARCHAR(128) NOT NULL WITH DEFAULT
",
TABLESPACE VARCHAR(128) NOT NULL WITH DEFAULT ",
SELECTION_FLAG VARCHAR(8) NOT NULL WITH DEFAULT ",
TABLE_EXISTS CHAR(1) NOT NULL WITH DEFAULT ",
USE_TABLE CHAR(1) NOT NULL WITH DEFAULT ",
GEN_COLUMNS CLOB(2M) NOT NULL NOT LOGGED
WITH DEFAULT ",
ORGANIZE_BY CLOB(2M) NOT NULL NOT LOGGED
WITH DEFAULT ",
CREATION_TEXT CLOB(2M) NOT NULL NOT LOGGED WITH
DEFAULT ",
ALTER_COMMAND CLOB(2M) NOT NULL NOT LOGGED
WITH DEFAULT ",
DISKUSE DOUBLE NOT NULL WITH DEFAULT 0,
FOREIGN KEY (RUN_ID)
REFERENCES ADVISE_INSTANCE (START_TIME)
ON DELETE CASCADE)
IN USERSPACE1
INDEX IN USERSPACE1;
--
-- Commit work
--
COMMIT WORK;
--
-- Optional Indexes: The following indexes are recommended for improved
performance
-- of explain-related utilities. These create index statements can be deleted, or
-- the indexes dropped if space is a problem.
--
CREATE INDEX STMT_I1 on
EXPLAIN_STATEMENT(EXPLAIN_TIME, EXPLAIN_LEVEL,
STMTNO, SECTNO);
CREATE INDEX ARG_I1 on
EXPLAIN_ARGUMENT(EXPLAIN_TIME, EXPLAIN_LEVEL, STMTNO,
SECTNO, OPERATOR_ID);
CREATE INDEX PRD_I1 on
EXPLAIN_PREDICATE(EXPLAIN_TIME, EXPLAIN_LEVEL, STMTNO,
SECTNO, OPERATOR_ID);
CREATE INDEX OPR_I1 on
EXPLAIN_OPERATOR(EXPLAIN_TIME, EXPLAIN_LEVEL, STMTNO,
SECTNO, OPERATOR_ID);
CREATE INDEX STM_I1 on
EXPLAIN_STREAM(EXPLAIN_TIME, EXPLAIN_LEVEL, STMTNO,
SECTNO);
CREATE INDEX OBJ_I1 on
EXPLAIN_OBJECT(EXPLAIN_TIME, EXPLAIN_LEVEL, STMTNO,
SECTNO);
CREATE INDEX IDX_I1 on
ADVISE_INDEX (EXPLAIN_TIME);
CREATE INDEX IDX_I2 on
ADVISE_INDEX (NAME, EXPLAIN_TIME);
CREATE INDEX MQT_I1 on
ADVISE_MQT (EXPLAIN_TIME);
CREATE INDEX MQT_I2 on
ADVISE_MQT (NAME, EXPLAIN_TIME);
CREATE INDEX PRT_I1 on
ADVISE_PARTITION (EXPLAIN_TIME);
--
-- Commit work
--
COMMIT WORK;
utils/UNEXPLAIN.ddl
-----
-- Licensed Materials - Property of IBM
--
-- Governed under the terms of the International
-- License Agreement for Non-Warranted Sample Code.
--
-- (C) COPYRIGHT International Business Machines Corp. 1996 - 2004
-- All Rights Reserved.
--
-- US Government Users Restricted Rights - Use, duplication or
-- disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
-----
DROP INDEX STMT_I1;
DROP INDEX ARG_I1;
DROP INDEX PRD_I1;
DROP INDEX OPR_I1;
DROP INDEX STM_I1;
DROP INDEX OBJ_I1;
DROP TABLE EXPLAIN_INSTANCE;
DROP TABLE EXPLAIN_STATEMENT;
DROP TABLE EXPLAIN_ARGUMENT;
DROP TABLE EXPLAIN_OBJECT;
DROP TABLE EXPLAIN_OPERATOR;
DROP TABLE EXPLAIN_PREDICATE;
DROP TABLE EXPLAIN_STREAM;
DROP TABLE ADVISE_INDEX;
DROP TABLE ADVISE_WORKLOAD;
tpccCom/comreg.h
// compreg.h : Declaration of the CCompReg
#pragma once
#include "resource.h" // main symbols
#include "tpccCom.h"
// CCompReg
class ATL_NO_VTABLE CCompReg :
public CComObjectRootEx<CComSingleThreadModel>,
public CComCoClass<CCompReg, &CLSID_CompReg>,

```

```

        public IDispatchImpl<IComponentRegistrar,
&IID_IComponentRegistrar, &LIBID_tpcComLib, /*wMajor =*/ 1, /*wMinor
= */ 0>
{
public:
    CCompReg()
    {
    }
    DECLARE_NO_REGISTRY()
    BEGIN_COM_MAP(CCompReg)
        COM_INTERFACE_ENTRY(IComponentRegistrar)
        COM_INTERFACE_ENTRY(IDispatch)
    END_COM_MAP()
    // IComponentRegistrar
public:
    STDMETHOD(Attach)(BSTR bstrPath)
    {
        return S_OK;
    }
    STDMETHOD(RegisterAll)()
    {
        return _AtlComModule.RegisterServer(TRUE);
    }
    STDMETHOD(UnregisterAll)()
    {
        _AtlComModule.UnregisterServer(TRUE);
        return S_OK;
    }
    STDMETHOD(GetComponents)(SAFEARRAY **ppCLSIDs,
SAFEARRAY **ppDescriptions)
    {
        if( ppCLSIDs == NULL || ppDescriptions == NULL )
            return E_POINTER;
        int nComponents = 0;
        for ( _ATL_OBJMAP_ENTRY** ppEntry =
_AtlComModule.m_ppAutoObjMapFirst; ppEntry <
_AtlComModule.m_ppAutoObjMapLast; ppEntry++)
        {
            if (*ppEntry != NULL)
            {
                _ATL_OBJMAP_ENTRY* pEntry
= *ppEntry;
                if (pEntry->pclsid != NULL)
                {
                    LPCTSTR
pszDescription = pEntry->pfnGetObjectDescription();
                    if (pszDescription)
                    {
                        LPOLESTR
pszCLSID;
                        StringFromCLSID(*pEntry->pclsid, &pszCLSID);
                        BSTR
pBSTR = OLE2BSTR(pszCLSID);
                        if (pBSTR
== NULL )
                        {
                            CoTaskMemFree(pszCLSID);
                            return E_OUTOFMEMORY;
                        }
                        HRESULT
hResult = SafeArrayPutElement(*ppCLSIDs, &i, pBSTR);
                        CoTaskMemFree(pszCLSID);
                        if(
FAILED(hResult) )
                        {
                            return hResult;
                        }
                        pBSTR =
T2BSTR_EX(pszDescription);
                        if (pBSTR
== NULL )
                        {
                            return E_OUTOFMEMORY;
                        }
                        hResult =
SafeArrayPutElement(*ppDescriptions, &i, pBSTR);
                        if(
FAILED(hResult) )
                        {
                            return hResult;
                        }
                        i++;
                    }
                }
            }
        }
        return S_OK;
    }
    STDMETHOD(RegisterComponent)(BSTR bstrCLSID)
    {
        CLSID clsid;
        CLSIDFromString(bstrCLSID, &clsid);
        _AtlComModule.RegisterServer(TRUE, &clsid);
        return S_OK;
    }
    STDMETHOD(UnregisterComponent)(BSTR bstrCLSID)
    {
        CLSID clsid;
        CLSIDFromString(bstrCLSID, &clsid);
        _AtlComModule.UnregisterServer(FALSE, &clsid);
        return S_OK;
    }
};
OBJECT_ENTRY_AUTO(CLSID_CompReg, CCompReg)

```

tpccCom/dlldatax.h

```
#pragma once
#ifdef _MERGE_PROXYSTUB
extern "C"
{
    BOOL WINAPI PrxDllMain(HINSTANCE hInstance, DWORD dwReason,
        LPVOID lpReserved);
    STDAPI PrxDllCanUnloadNow(void);
    STDAPI PrxDllGetClassObject(REFCLSID rclsid, REFIID riid, LPVOID*
        ppv);
    STDAPI PrxDllRegisterServer(void);
    STDAPI PrxDllUnregisterServer(void);
}
#endif
```

tpccCom/Resource.h

```
//{{NO_DEPENDENCIES}}
// Microsoft Visual C++ generated include file.
// Used by tpccCom.rc
//
#define IDS_PROJNAME            100
#define IDR_TPCCCOM            101
#define IDR_TPCC_COM          102
// Next default values for new objects
//
#ifdef APSTUDIO_INVOKED
#ifdef APSTUDIO_READONLY_SYMBOLS
#define _APS_NEXT_RESOURCE_VALUE 201
#define _APS_NEXT_COMMAND_VALUE 32768
#define _APS_NEXT_CONTROL_VALUE 201
#define _APS_NEXT_SYMED_VALUE 103
#endif
#endif
```

tpccCom/stdafx.h

```
// stdafx.h : include file for standard system include files,
// or project specific include files that are used frequently,
// but are changed infrequently
#pragma once
#ifdef STRICT
#define STRICT
#endif
// Modify the following defines if you have to target a platform prior to the ones
// specified below.
// Refer to MSDN for the latest info on corresponding values for different
// platforms.
#ifdef WINVER // Allow use of features
// specific to Windows 95 and Windows NT 4 or later.
#define WINVER 0x0400 // Change this to the appropriate
// value to target Windows 98 and Windows 2000 or later.
#endif
#ifdef _WIN32_WINNT // Allow use of features specific to
// Windows NT 4 or later.
#define _WIN32_WINNT 0x0400 // Change this to the appropriate
// value to target Windows 2000 or later.
#endif
#ifdef _WIN32_WINDOWS // Allow use of features specific to
// Windows 98 or later.
#define _WIN32_WINDOWS 0x0410 // Change this to the appropriate value to
// target Windows Me or later.
#endif
#ifdef _WIN32_IE // Allow use of features specific to
// IE 4.0 or later.
```

```
#define _WIN32_IE 0x0400 // Change this to the appropriate value to
// target IE 5.0 or later.
#endif
#define _ATL_APARTMENT_THREADED
#define _ATL_NO_AUTOMATIC_NAMESPACE
#define _ATL_CSTRING_EXPLICIT_CONSTRUCTORS // some
// CString constructors will be explicit
// turns off ATL's hiding of some common and often safely ignored warning
// messages
#define _ATL_ALL_WARNINGS

#include <comsvcs.h>
#include "resource.h"
#include <atlbase.h>
#include <atlcom.h>
using namespace ATL;
```

tpccCom/tpccCom.h

```
// tpcc_com.h : Declaration of the Ctpcc_com
#pragma once
#include "tpccCom.h"
#include "resource.h" // main symbols
#include <comsvcs.h>
#include "..\tpcc\api\tpcc.h"
#include <db2tpcc.h>
#include <tpcc.h>
#define NULL_DB "nullDB"
static HINSTANCE dbInstance = NULL;
static CRITICAL_SECTION debugMutex;
static CRITICAL_SECTION errorMutex;
static int comServerID = 0;
static ofstream debugStream;
static ofstream errorStream;
static int debugFileOpen = 0;
static int errorFileOpen = 0;
static int nullDB = 0;
static char dbType[32];
static char dbName[32];
typedef INT (*NORD_PTR)(nord_wrapper *nord, void *connectHandle);
typedef INT (*PYMT_PTR)(pymt_wrapper *pymt, void *connectHandle);
typedef INT (*ORDS_PTR)(ords_wrapper *ords, void *connectHandle);
typedef INT (*STOK_PTR)(stok_wrapper *stok, void *connectHandle);
typedef INT (*CONNECT_PTR)(char *dbName, void **connectHandle);
typedef INT (*DISCONNECT_PTR)(void *connectHandle);
NORD_PTR do_nord;
PYMT_PTR do_pymt;
ORDS_PTR do_ords;
STOK_PTR do_stok;
CONNECT_PTR do_connection;
DISCONNECT_PTR do_disconnect;

// Ctpcc_com
class ATL_NO_VTABLE Ctpcc_com :
public CComObjectRootEx<CComMultiThreadModel>,
public IObjectControl,
public CComCoClass<Ctpcc_com, &CLSID_tpcc_com>,
public Itpcc_com
{
public:
    Ctpcc_com()
    {
        int rc = ERR;
        connected = 0;
        connectHandleInUse = 0;
        if(debugFlag)
        {
            if(!debugFileOpen)
```



```

    {
InitializeCriticalSection(&debugMutex);
//open comLog
char comLogFile[128];

sprintf(comLogFile,"C:\\inetpub\\wwwroot\\tpcc\\comLog_debug.txt");
debugStream.rdbuf(
)->open(comLogFile,ios_base::in | ios_base::out | ios_base::app);
debugFileOpen = 1;
    }

//open error log file
if(!errorFileOpen)
{
InitializeCriticalSection(&errorMutex);
char errorLogFile[128];

sprintf(errorLogFile,"C:\\inetpub\\wwwroot\\tpcc\\comLog_err.txt");
errorStream.rdbuf(
)->open(errorLogFile,ios_base::in | ios_base::out | ios_base::app);
errorFileOpen=1;
}

//get registry values
if(rc = readRegistry() != OK)
{
ERRORMSG("Unable to open registry key "
<< REGISTRY_SUB_KEY << " rc:" << rc <<endl);
return;
}

DEBUGMSG("nullDB:" <<nullDB<<"
dbType:"<<dbType<<" dbName:"<<dbName<<endl);

//load library based on registry
if (rc = loadLibrary()) != OK)
{
ERRORMSG("load library failure rc:" << rc
<< endl);
return;
}

DEBUGMSG("dbtype:"<<dbType<<" instance:" <<
DEBUGADDRESS(dbInstance) << " loaded." << endl);

//connect to db
EnterCriticalSection(&errorMutex);
if(rc = connectDB() != OK)
{
ERRORMSG("unable to connect to db
"<<dbName<<" rc :"<<rc <<endl);
LeaveCriticalSection(&errorMutex);
return;
}
LeaveCriticalSection(&errorMutex);

DEBUGMSG("connected to db " <<dbName<<" rc:"<<
rc << " context:" <<DEBUGADDRESS(connectHandle) << endl);
DECLARE_PROTECT_FINAL_CONSTRUCT()
HRESULT FinalConstruct()
{
return S_OK;
}

void FinalRelease()
{
}

```

```

}
DECLARE_REGISTRY_RESOURCEID(IDR_TPCC_COM)
BEGIN_COM_MAP(Ctpcc_com)
COM_INTERFACE_ENTRY(Itpcc_com)
COM_INTERFACE_ENTRY(IObjectControl)
END_COM_MAP()
// IObjectControl
public:
STDMETHOD(Activate)();
STDMETHOD_(BOOL, CanBePooled)();
STDMETHOD_(void, Deactivate)();
CComPtr<IObjectContext> m_spObjectContext;

// Itpcc_com
public:
STDMETHOD(doStockLevel)(INT *size, UCHAR **buffer);
STDMETHOD(doNewOrder)(INT* size, UCHAR** buffer);
STDMETHOD(doPayment)(INT* size, UCHAR** buffer);
STDMETHOD(doOrderStatus)(INT* size, UCHAR** buffer);
STDMETHOD(doDBInfo)(void);
STDMETHOD(doSetComplete)(void);
int connected;
int connectHandleInUse;

private:
//db2 specific context
void *connectHandle;
int loadLibrary();
int readRegistry();
int connectDB();
};
OBJECT_ENTRY_AUTO(__uuidof(tpcc_com), Ctpcc_com)
tpccCom/tpcc_com.h

// tpcc_com.h : Declaration of the Ctpcc_com
#pragma once
#include "tpccCom.h"
#include "resource.h"// main symbols
#include <comsvcs.h>
#include "..\tpcc\api\tpcc.h"
#include <db2tpcc.h>
#include <tpcc.h>
#define NULL_DB "nullDB"
static HINSTANCE dbInstance = NULL;
static CRITICAL_SECTION debugMutex;
static CRITICAL_SECTION errorMutex;
static int comServerID = 0;
static ofstream debugStream;
static ofstream errorStream;
static int debugFileOpen = 0;
static int errorFileOpen = 0;
static int nullDB = 0;
static char dbType[32];
static char dbName[32];
typedef INT (*NORD_PTR)(nord_wrapper *nord,void *connectHandle);
typedef INT (*PYMT_PTR)(paym_wrapper *pymt,void *connectHandle);
typedef INT (*ORDS_PTR)(ords_wrapper *ords,void *connectHandle);
typedef INT (*STOK_PTR)(stok_wrapper *stok,void *connectHandle);
typedef INT (*CONNECT_PTR)(char *dbName,void **connectHandle);
typedef INT (*DISCONNECT_PTR)(void *connectHandle);
NORD_PTR do_nord;
PYMT_PTR do_pymt;
ORDS_PTR do_ords;
STOK_PTR do_stok;
CONNECT_PTR do_connection;
DISCONNECT_PTR do_disconnect;

```

```

// Ctpcc_com
class ATL_NO_VTABLE Ctpcc_com :
    public CComObjectRootEx<CComMultiThreadModel>,
    public IObjectControl,
    public CComCoClass<Ctpcc_com, &CLSID_tpcc_com>,
    public Itpcc_com
{
public:
    Ctpcc_com()
    {
        int rc = ERR;
        connected = 0;
        connectHandleInUse = 0;
        if(debugFlag)
        {
            if(!debugFileOpen)
            {
                InitializeCriticalSection(&debugMutex);

                //open comLog
                char comLogFile[128];

                sprintf(comLogFile, "C:\\inetpub\\wwwroot\\tpcc\\comLog_debug.txt");
                debugStream.rdbuf(
                debugFileOpen = 1;
                )->open(comLogFile, ios_base::in | ios_base::out | ios_base::app);
            }

            //open error log file
            if(!errorFileOpen)
            {
                InitializeCriticalSection(&errorMutex);
                char errorLogFile[128];

                sprintf(errorLogFile, "C:\\inetpub\\wwwroot\\tpcc\\comLog_err.txt");
                errorStream.rdbuf(
                errorFileOpen=1;
                )->open(errorLogFile, ios_base::in | ios_base::out | ios_base::app);
            }

            //get registry values
            if(rc = readRegistry() != OK)
            {
                ERRORMSG("Unable to open registry key "
                << REGISTRY_SUB_KEY << " rc:" << rc << endl);
                return;
            }

            DEBUGMSG("nullDB:" << nullDB << "
            dbType:" << dbType << " dbName:" << dbName << endl);

            //load library based on registry
            if( rc = loadLibrary() != OK)
            {
                ERRORMSG("load library failure rc:" << rc
                << endl);
                return;
            }

            DEBUGMSG("dbtype:" << dbType << " instance:" <<
            DEBUGADDRESS(dbInstance) << " loaded." << endl);

            //connect to db
            EnterCriticalSection(&errorMutex);
            if(rc = connectDB() != OK)
            {
                ERRORMSG("unable to connect to db
                " << dbName << " rc : " << rc << endl);
            }
        }
    }
};

```

```

        LeaveCriticalSection(&errorMutex);
        return;
    }
    LeaveCriticalSection(&errorMutex);

    DEBUGMSG("connected to db " << dbName << " rc:" <<
rc << " context:" << DEBUGADDRESS(connectHandle) << endl);
    DECLARE_PROTECT_FINAL_CONSTRUCT()
    HRESULT FinalConstruct()
    {
        return S_OK;
    }

    void FinalRelease()
    {
    }

    DECLARE_REGISTRY_RESOURCEID(IDR_TPCC_COM)
    BEGIN_COM_MAP(Ctpcc_com)
        COM_INTERFACE_ENTRY(Itpcc_com)
        COM_INTERFACE_ENTRY(IObjectControl)
    END_COM_MAP()
    // IObjectControl
public:
    STDMETHOD(Activate)();
    STDMETHOD_(BOOL, CanBePooled)();
    STDMETHOD_(void, Deactivate)();
    CComPtr<IObjectContext> m_spObjectContext;

    // Itpcc_com
public:
    STDMETHOD(doStockLevel)(INT *size, UCHAR **buffer);
    STDMETHOD(doNewOrder)(INT* size, UCHAR** buffer);
    STDMETHOD(doPayment)(INT* size, UCHAR** buffer);
    STDMETHOD(doOrderStatus)(INT* size, UCHAR** buffer);
    STDMETHOD(doDBInfo)(void);
    STDMETHOD(doSetComplete)(void);
    int connected;
    int connectHandleInUse;

private:
    //db2 specific context
    void *connectHandle;
    int loadLibrary();
    int readRegistry();
    int connectDB();
};
OBJECT_ENTRY_AUTO(__uuidof(tpcc_com), Ctpcc_com)
tpccCom/tpccCom.def

; tpccCom.def : Declares the module parameters.
LIBRARY "tpccCom.DLL"
EXPORTS
    DllCanUnloadNow PRIVATE
    DllGetClassObject PRIVATE
    DllRegisterServer PRIVATE
    DllUnregisterServer PRIVATE

tpccCom/tpccCom.idl

// tpccCom.idl : IDL source for tpccCom
//
// This file will be processed by the MIDL tool to
// produce the type library (tpccCom.tlb) and marshalling code.
import "oaidl.idl";
import "ocidl.idl";
//this is test.

```

```

[
    object,
    uuid(a817e7a2-43fa-11d0-9e44-00aa00b6770a),
    dual,
    helpstring("IComponentRegistrar Interface"),
    pointer_default(unique)
]
interface IComponentRegistrar : IDispatch
{
    [id(1)] HRESULT Attach([in] BSTR bstrPath);
    [id(2)] HRESULT RegisterAll();
    [id(3)] HRESULT UnregisterAll();
    [id(4)] HRESULT GetComponents([out]
SAFEARRAY(BSTR)* pBstrCLSIDs, [out] SAFEARRAY(BSTR)*
pBstrDescriptions);
    [id(5)] HRESULT RegisterComponent([in] BSTR bstrCLSID);
    [id(6)] HRESULT UnregisterComponent([in] BSTR bstrCLSID);
};
[
    object,
    uuid(5B4FA473-2E68-4D79-A626-F38B30B8196E),
    helpstring("Itpcc_com Interface"),
    pointer_default(unique)
]
interface Itpcc_com : IUnknown{
    [helpstring("method doStockLevel")] HRESULT doStockLevel([in]
INT *size, [in,out, size_is(*size)] UCHAR **buffer);
    [helpstring("method doNewOrder")] HRESULT doNewOrder([in]
INT* size, [in,out,size_is(*size)] UCHAR** buffer);
    [helpstring("method doPayment")] HRESULT doPayment([in] INT*
size, [in,out,size_is(*size)] UCHAR** buffer);
    [helpstring("method doOrderStatus")] HRESULT
doOrderStatus([in] INT* size, [in,out,size_is(*size)] UCHAR** buffer);
    [helpstring("method doDBInfo")] HRESULT doDBInfo(void);
    [helpstring("method doSetComplete")] HRESULT
doSetComplete(void);
};
[
    uuid(91F1B8B0-89E9-457B-A228-3E2D6CE3E752),
    version(1.0),
    helpstring("tpccCom 1.0 Type Library"),

custom(a817e7a1-43fa-11d0-9e44-00aa00b6770a,"{90EEDAFF-F8D3-4711-99
A9-8AC3C0FE5DB9}")
]
library tpccComLib
{
    importlib("stdole2.tlb");
    [
        uuid(90EEDAFF-F8D3-4711-99A9-8AC3C0FE5DB9),
        helpstring("ComponentRegistrar Class")
    ]
    coclass CompReg
    {
        [default] interface IComponentRegistrar;
    };
    [
        uuid(5F752BF2-F739-43D4-8492-44C19581C0A1),
        helpstring("tpcc_com Class")
    ]
    coclass tpcc_com
    {
        [default] interface Itpcc_com;
    };
};
tpccCom/tpcc_com.rgs

```

HKCR

```

{
    tpccCom.tpcc_com.1 = s 'tpcc_com Class'
    {
        CLSID = s
        '{5F752BF2-F739-43D4-8492-44C19581C0A1}'
    }
    tpccCom.tpcc_com = s 'tpcc_com Class'
    {
        CLSID = s
        '{5F752BF2-F739-43D4-8492-44C19581C0A1}'
        CurVer = s 'tpccCom.tpcc_com.1'
    }
    NoRemove CLSID
    {
        ForceRemove
        {5F752BF2-F739-43D4-8492-44C19581C0A1} = s 'tpcc_com Class'
    }
    ProgID = s 'tpccCom.tpcc_com.1'
    VersionIndependentProgID = s
    'tpccCom.tpcc_com'
    InprocServer32 = s '%MODULE%'
    {
        val ThreadingModel = s 'Both'
    }
    val AppID = s '%APPID%'
    'TypeLib' = s
    '{91F1B8B0-89E9-457B-A228-3E2D6CE3E752}'
    }
}
tpccCom/comreg.cpp
// compreg.cpp : Implementation of CCompReg
#include "stdafx.h"
#include "comreg.h"

// CCompReg
tpccCom/stdafx.cpp
// stdafx.cpp : source file that includes just the standard includes
// tpccCom.pch will be the pre-compiled header
// stdafx.obj will contain the pre-compiled type information
#include "stdafx.h"
tpccCom/tpccCom.cpp
// tpccCom.cpp : Implementation of DLL Exports.
//
// Note: COM+ 1.0 Information:
// Please remember to run Microsoft Transaction Explorer to install the
// component(s).
// Registration is not done by default.
#include "stdafx.h"
#include "resource.h"
#include "tpccCom.h"
#include "comreg.h"
#include "dldatax.h"
class CtpccComModule : public CAtdllModuleT< CtpccComModule >
{
public :
    DECLARE_LIBID(LIBID_tpccComLib)

    DECLARE_REGISTRY_APPID_RESOURCEID(IDR_TPCCCOM,
    "{11ED2355-1A27-42F1-ADFF-F201F5E82BCE}")
};
CtpccComModule _AtlModule;

```

```

// DLL Entry Point
extern "C" BOOL WINAPI DllMain(HINSTANCE hInstance, DWORD
dwReason, LPVOID lpReserved)
{
#ifdef _MERGE_PROXYSTUB
    if (!PrxDllMain(hInstance, dwReason, lpReserved))
        return FALSE;
#endif
    hInstance;
    return _AtlModule.DllMain(dwReason, lpReserved);
}

// Used to determine whether the DLL can be unloaded by OLE
STDAPI DllCanUnloadNow(void)
{
#ifdef _MERGE_PROXYSTUB
    HRESULT hr = PrxDllCanUnloadNow();
    if (FAILED(hr))
        return hr;
#endif
    return _AtlModule.DllCanUnloadNow();
}

// Returns a class factory to create an object of the requested type
STDAPI DllGetClassObject(REFCLSID rclsid, REFIID riid, LPVOID* ppv)
{
#ifdef _MERGE_PROXYSTUB
    if (PrxDllGetClassObject(rclsid, riid, ppv) == S_OK)
        return S_OK;
#endif
    return _AtlModule.DllGetClassObject(rclsid, riid, ppv);
}

// DllRegisterServer - Adds entries to the system registry
STDAPI DllRegisterServer(void)
{
    // registers object, typelib and all interfaces in typelib
    HRESULT hr = _AtlModule.DllRegisterServer();
#ifdef _MERGE_PROXYSTUB
    if (FAILED(hr))
        return hr;
    hr = PrxDllRegisterServer();
#endif
    return hr;
}

// DllUnregisterServer - Removes entries from the system registry
STDAPI DllUnregisterServer(void)
{
    HRESULT hr = _AtlModule.DllUnregisterServer();
#ifdef _MERGE_PROXYSTUB
    if (FAILED(hr))
        return hr;
    hr = PrxDllRegisterServer();
    if (FAILED(hr))
        return hr;
    hr = PrxDllUnregisterServer();
#endif
    return hr;
}

```

tpccCom/tpcc_com.cpp

```

// tpcc_com.cpp : Implementation of Ctpcc_com
#include "stdafx.h"
#include "tpcc_com.h"
#include ".\tpcc_com.h"

```

```

#include <db2tpcc.h>
// Ctpcc_com
HRESULT Ctpcc_com::Activate()
{
    HRESULT hr = GetObjectContext(&m_spObjectContext);
    if (SUCCEEDED(hr))
    {
        DEBUGMSG("Object assigned to thread."<<endl);
        return S_OK;
    }
    return hr;
}

BOOL Ctpcc_com::CanBePooled()
{
    DEBUGMSG("CanBePooled() returning true"<<endl);
    return TRUE;
}

void Ctpcc_com::Deactivate()
{
    DEBUGMSG("deactivated() releasing object back into
pool"<<endl);
    m_spObjectContext.Release();
}

/*
*****
** Name          :          doSetComplete
** Description   :          Release object back
into com pool
** Parameters    :
** Returns       :          int - return code
** Comments      :          Calls SetComplete on
the object that the com
**               :          pool manager returned
to the caller(isapi thread)
*****
*/
STDMETHODIMP Ctpcc_com::doSetComplete(void)
{
    // TODO: Add your implementation code here
    HRESULT hres = m_spObjectContext->SetComplete();
    if (SUCCEEDED(hres))
    {
        DEBUGMSG("SetComplete successful. object bit set to
release object into pool."<<endl);
    }
    else
    {
        DEBUGMSG("SetComplete failed. object bit set to
release object into pool."<<endl);
        ERRORMSG("SetComplete() failed,
code:<<HRESULT_CODE(hres)<<"
facility:<<HRESULT_FACILITY(hres)<<" hres:<<hex<<hres<<endl);
    }
    return S_OK;
}

/*
*****
** Name          :          doStockLevel
** Description   :          Call db2 dll entry point
to execute txn
** Parameters    :          int*
size of UCHAR buffer to pay attention to
*****
*/

```

```

**                                     UCHAR**
char buffer that holds txn wrapper struct
** Returns                             :
**                                     int - return code
** Comments                             :
**
*****
*/
STDMETHODIMP Ctpcc_com::doStockLevel(INT *size, UCHAR **buffer)
{
    stok_wrapper *stok;

    stok = (stok_wrapper *) *buffer;

    if(!connectHandleInUse)
    {
        DEBUGMSG("Setting Context handle in use to
true" << endl);
        connectHandleInUse = 1;
    }
    else
    {
        DEBUGMSG("Context handle in use." << endl);
        ERRORMSG("Context handle in use." << endl);
        return ERR_HANDLE_IN_USE;
    }
    DEBUGMSG("Calling do_stok call using
connectHandle:" << DEBUGADDRESS(connectHandle) << "
w_id:" << stok->in_stok.s_W_ID << " d_id:" << stok->in_stok.s_D_ID <<
"
s_transtatus:" << stok->out_stok.s_transtatus << endl);

    do_stok(stok,connectHandle);
    DEBUGMSG("Return from do_stok call using
connectHandle:" << DEBUGADDRESS(connectHandle) << "
w_id:" << stok->in_stok.s_W_ID << " d_id:" << stok->in_stok.s_D_ID <<
"
s_transtatus:" << stok->out_stok.s_transtatus << endl);
    DEBUGMSG("Connection handle set to free" << endl);
    connectHandleInUse = 0;
    return S_OK;
}
/*
*****
** Name                                     : doNewOrder
** Description                             :
**                                     Call db2 dll entry point
to execute txn
** Parameters                             :
**                                     int*
size of UCHAR buffer to pay attention to
**                                     UCHAR**
char buffer that holds txn wrapper struct
** Returns                             :
**                                     int - return code
** Comments                             :
**
*****
*/
STDMETHODIMP Ctpcc_com::doNewOrder(INT* size, UCHAR** buffer)
{
    nord_wrapper *nord;
    nord = (nord_wrapper *) *buffer;
    if(!connectHandleInUse)
    {
        DEBUGMSG("Setting Context handle in use to
true" << endl);
        connectHandleInUse = 1;
    }
    else
    {
        DEBUGMSG("Context handle in use." << endl);
        ERRORMSG("Context handle in use." << endl);
        return ERR_HANDLE_IN_USE;
    }
    DEBUGMSG("Calling do_nord call using
connectHandle:" << DEBUGADDRESS(connectHandle) << "
w_id:" << nord->in_nord.s_W_ID << " d_id:" << nord->in_nord.s_D_ID <<
"
s_transtatus:" << nord->out_nord.s_transtatus << endl);
    do_nord(nord,connectHandle);
    DEBUGMSG("Return from do_nord call using
connectHandle:" << DEBUGADDRESS(connectHandle) << "
w_id:" << nord->in_nord.s_W_ID << " d_id:" << nord->in_nord.s_D_ID <<
"
s_transtatus:" << nord->out_nord.s_transtatus << endl);
    DEBUGMSG("Connection handle set to free" << endl);
    connectHandleInUse = 0;
    return S_OK;
}
/*
*****
** Name                                     : doPayment
** Description                             :
**                                     Call db2 dll entry point
to execute txn
** Parameters                             :
**                                     int*
size of UCHAR buffer to pay attention to
**                                     UCHAR**
char buffer that holds txn wrapper struct
** Returns                             :
**                                     int - return code
** Comments                             :
**
*****
*/
STDMETHODIMP Ctpcc_com::doPayment(INT* size, UCHAR** buffer)
{
    paym_wrapper *pymt;
    pymt = (paym_wrapper *) *buffer;
    if(!connectHandleInUse)
    {
        DEBUGMSG("Setting Context handle in use to
true" << endl);
        connectHandleInUse = 1;
    }
    else
    {
        DEBUGMSG("Context handle in use." << endl);
        ERRORMSG("Context handle in use." << endl);
        return ERR_HANDLE_IN_USE;
    }
    DEBUGMSG("Calling do_pymt call using
connectHandle:" << DEBUGADDRESS(connectHandle) << "
w_id:" << pymt->in_paym.s_W_ID << " d_id:" << pymt->in_paym.s_D_ID <<
"
s_transtatus:" << pymt->out_paym.s_transtatus << endl);
    do_pymt(pymt,connectHandle);
    DEBUGMSG("Return from do_pymt call using
connectHandle:" << DEBUGADDRESS(connectHandle) << "
w_id:" << pymt->in_paym.s_W_ID << " d_id:" << pymt->in_paym.s_D_ID <<
"
s_transtatus:" << pymt->out_paym.s_transtatus << endl);
    DEBUGMSG("Connection handle set to free" << endl);
    connectHandleInUse = 0;
    return S_OK;
}

```

```

s_transtatus:"<<pymt->out_paym.s_transtatus<<endl);
    DEBUGMSG("Connection handle set to free" <<endl);
    connectHandleInUse = 0;

    return S_OK;
}
/*
*****
** Name          :          doOrderStatus
** Description    :
**               Call db2 dll entry point
to execute txn
** Parameters    :
**               int*
size of UCHAR buffer to pay attention to
** Returns       :
**               UCHAR**
char buffer that holds txn wrapper struct
** Returns       :
**               int - return code
** Comments      :
**
*****
*/
STDMETHODIMP Ctpcc_com::doOrderStatus(INT* size, UCHAR** buffer)
{
    ords_wrapper *ords;
    ords = (ords_wrapper *) *buffer;
    if(!connectHandleInUse)
    {
        DEBUGMSG("Setting Context handle in use to
true" <<endl);
        connectHandleInUse = 1;
    }
    else
    {
        DEBUGMSG("Context handle in use." <<endl);
        ERRORMSG("Context handle in use." <<endl);
        return ERR_HANDLE_IN_USE;
    }

    DEBUGMSG("Calling do_ords call using
connectHandle:" <<DEBUGADDRESS(connectHandle)<<"
w_id:" <<ords->in_ords.s_W_ID<<" d_id:" << ords->in_ords.s_D_ID<<"
"
s_transtatus:" <<ords->out_ords.s_transtatus<<endl);
    do_ords(ords,connectHandle);
    DEBUGMSG("Return from do_ords call using
connectHandle:" <<DEBUGADDRESS(connectHandle)<<"
w_id:" <<ords->in_ords.s_W_ID<<" d_id:" << ords->in_ords.s_D_ID<<"
"
s_transtatus:" <<ords->out_ords.s_transtatus<<endl);
    DEBUGMSG("Connection handle set to free" <<endl);
    connectHandleInUse = 0;

    return S_OK;
}
/*
*****
** Name          :          doDBInfo
** Description    :
**               Function to test com
interface
** Parameters    :
** Returns       :
**               int - return code
** Comments      :

```

```

**
*****
*/
STDMETHODIMP Ctpcc_com::doDBInfo(void)
{
    DEBUGMSG("Stub function to warm object pool" <<endl);
    return S_OK;
}
/*
*****
** Name          :          loadLibrary
** Description    :
**               Function loads
appropriate db library based on
**               registry setting
** Parameters    :
** Returns       :
**               int - return code
** Comments      :
**
*****
*/
Ctpcc_com::loadLibrary()
{
    DEBUGMSG("Entered loadLibrary function" <<endl);
    //check to see if dbInstance is already loaded
    if(!dbInstance)
    {
        DEBUGMSG("Database dll not loaded. Loading
dll." <<endl);
        if (nullDB)
        {
            DEBUGMSG("Loading "<<dbType <<"
nulldb dll." << endl);
            dbInstance =
LoadLibrary("c:\\inetpub\\wwwroot\\tpcc\\nullDB.dll");
            if(dbInstance == NULL)
            {
                DEBUGMSG("Unable to load null
db dll, rc:" <<GetLastError());
                ERRORMSG("Unable to load null
db dll, rc:" <<GetLastError());
                return
ERR_NULL_DLL_NOT_LOADED;
            }
            DEBUGMSG(dbType <<" nulldb dll
loaded" <<endl);
        }
        else if(strcmp(dbType,"DB2") == 0)
        {
            DEBUGMSG("Loading "<<dbType <<" dll."
<< endl);
            dbInstance =
LoadLibrary("c:\\inetpub\\wwwroot\\tpcc\\tpccDB2glue.dll");
            if(dbInstance == NULL)
            {
                DEBUGMSG("Unable to load
library." <<endl);
                ERRORMSG("Unable to load com
dll, rc:" << GetLastError() << endl);
                return
ERR_DB2_DLL_NOT_LOADED;
            }
            DEBUGMSG(dbType<<" dll
loaded" <<endl);

```

```

    }
    else if( strcmp(dbType,"ORACLE") == 0 )
    {
        DEBUGMSG("Unable to load orcale
dll"<<endl);
        ERRORMSG("Unable to load orcale dll,
rc:"<<GetLastError()<<endl);
        return
ERR_ORACLE_DLL_NOT_LOADED;
    }
    else
    {
        DEBUGMSG("Unknown database type
dll:"<<dbType<<endl);
        ERRORMSG("Unknown database type
dll:"<<dbType<<endl);
        return ERR_UNKNOWN_DB;
    }
    //retrieve function addresses from instance loaded.
    DEBUGMSG("Getting do_connection function address
from "<<dbType<<" dll"<<endl);
    if( do_connection =
(CONNECT_PTR)GetProcAddress(dbInstance,"connect_db") == NULL )
        return
ERR_CONNECT_ADDRESS_NOT_FOUND;
    DEBUGMSG("do_connection
address:"<<DEBUGADDRESS(do_connection)<<endl);
    DEBUGMSG("Getting do_disconnect function address
from "<<dbType<<" dll"<<endl);
    if( do_disconnect =
(DISCONNECT_PTR)GetProcAddress(dbInstance,"disconnect_db") ==
NULL )
        return
ERR_DISCONNECT_ADDRESS_NOT_FOUND;
    DEBUGMSG("do_disconnect
address:"<<DEBUGADDRESS(do_disconnect)<<endl);
    DEBUGMSG("Getting do_nord function address from
"<<dbType<<" dll"<<endl);
    if( do_nord = (NORD_PTR)
GetProcAddress(dbInstance,"do_nord") == NULL)
        return
ERR_NORD_ADDRESS_NOT_FOUND;
    DEBUGMSG("do_nord function
address:"<<DEBUGADDRESS(do_nord)<<endl);
    DEBUGMSG("Getting do_pymt function address from
"<<dbType<<" dll"<<endl);
    if( do_pymt = (PYMT_PTR)
GetProcAddress(dbInstance,"do_pymt") == NULL)
        return
ERR_PYMT_ADDRESS_NOT_FOUND;
    DEBUGMSG("do_pymt function
address:"<<DEBUGADDRESS(do_pymt)<<endl);
    DEBUGMSG("Getting do_ords function address from
"<<dbType<<" dll"<<endl);
    if( do_ords = (ORDS_PTR)
GetProcAddress(dbInstance,"do_ords") == NULL)
        return
ERR_ORDS_ADDRESS_NOT_FOUND;
    DEBUGMSG("do_ords function
address:"<<DEBUGADDRESS(do_ords)<<endl);
    DEBUGMSG("Getting do_stok function address from
"<<dbType<<"
dll"<<endl);
    if( do_stok = (STOK_PTR)
GetProcAddress(dbInstance,"do_stok") == NULL)
        return
ERR_STOK_ADDRESS_NOT_FOUND;

```

```

        DEBUGMSG("do_stok function
address:"<<DEBUGADDRESS(do_stok)<<endl);
        DEBUGMSG("All function addresses retrieved
successfully."<<endl);
    }
    return OK;
}
/*
*****
** Name          :          readRegistry()
** Description   :
**              :          Function reads registry
value
** Parameters   :
** Returns      :
**              :          int - return code
** Comments     :
**              :          Values retrieved from
registry
**              :          dbName, dbUserName,
and dbUserPassword
*****
*/
Ctpcc_com::readRegistry()
{
    //open registry key
    HKEY   registryKey;
    DWORD  regType;
    char   value[MAX_STRING_LEN];
    DWORD  regValue;
    DWORD  regValueSize = MAX_STRING_LEN;
    DEBUGMSG("Entered readRegistry(), opening key:"<<
REGISTRY_SUB_KEY <<endl);
    //open up registry key
    if(RegOpenKeyEx(HKEY_LOCAL_MACHINE,REGISTRY_SUB_KEY,0,K
EY_READ,&registryKey) == ERROR_SUCCESS)
    {
        DEBUGMSG(REGISTRY_SUB_KEY<<" open,
getting database type from key"<<endl);
        regValueSize = sizeof(value);
        if
(REGQueryValueEx(registryKey,DB_TYPE,0,&regType,(BYTE *)
&value,&regValueSize) == ERROR_SUCCESS )
            strcpy(dbType,value);
        DEBUGMSG("Database type:"<<dbType<<" from
registry key."<<endl);
        DEBUGMSG("Getting database name from registry
key."<<endl);
        regValueSize = sizeof(value);
        if
(REGQueryValueEx(registryKey,DB_NAME,0,&regType,(BYTE *)
&value,&regValueSize) == ERROR_SUCCESS )
            strcpy(dbName,value);
        DEBUGMSG("Database name:"<<dbName<<endl);
        DEBUGMSG("Getting null database flag from
key."<<endl);
        regValueSize = sizeof(regValue);
        if(RegQueryValueEx(registryKey,NULL_DB,0,&regType,(BYTE
*)&regValue,&regValueSize) == ERROR_SUCCESS)
            nullDB = regValue;
        DEBUGMSG("Null database flag:"<<nullDB<<endl);
        return OK;
    }
    DEBUGMSG("Error, unable to open registry key."<<endl);
    return ERR_UNABLE_TO_OPEN_REG;
}

```

```

/*
*****
** Name           :          connectDB
** Description    :
**               :          Function connects to
the db
** Parameters    :
** Returns       :
**               :          int - return code
** Comments      :
**
*****
*/
Ctpcc_com::connectDB()
{
    DEBUGMSG("Entered connectDB(), checking if object is
connected."<<endl);
    if(!connected)
    {
        DEBUGMSG("Object not connected, calling
do_connection with dbName:"<<dbName<<" connectHandle:"<<
DEBUGADDRESS(connectHandle)<<endl);
        if(!connectHandleInUse)
        {
            DEBUGMSG("Setting Context handle in use
to true"<<endl);
            connectHandleInUse = 1;
            connected =
do_connection(dbName,&connectHandle);
            if(connected != OK)
            {
                DEBUGMSG("Object do_connect
failed, rc:"<<connected<<endl);
                ERRORMSG("Object do_connect
failed, rc:"<<connected<<endl);
                return connected;
            }
            DEBUGMSG("Object connection complete,
connectHandle:"<<DEBUGADDRESS(connectHandle)<<endl);
            connectHandleInUse = 0;
            return OK;
        }
        else
        {
            DEBUGMSG("Object's connectHandle
already in use, connect failed"<<endl);
            ERRORMSG("Object's connectHandle
already in use, connect failed"<<endl);
            return ERR_HANDLE_IN_USE;
        }
    }
    DEBUGMSG("Object already has connection established."<<endl);
    return OK;
}

```

TpccCom/dlldata.c

```

/*****
DllData file -- generated by MIDL compiler
DO NOT ALTER THIS FILE
This file is regenerated by MIDL on every IDL file compile.
To completely reconstruct this file, delete it and rerun MIDL
on all the IDL files in this DLL, specifying this file for the
/dlldata command line option
*****/
#define PROXY_DELEGATION
#include <rpccom.h>
#ifdef __cplusplus

```

```

extern "C" {
#endif
EXTERN_PROXY_FILE( tpccCom )
PROXYFILE_LIST_START
/* Start of list */
REFERENCE_PROXY_FILE( tpccCom ),
/* End of list */
PROXYFILE_LIST_END
DLLDATA_ROUTINES( aProxyFileList, GET_DLL_CLSID )
#ifdef __cplusplus
} /*extern "C" */
#endif
/* end of generated dlldata file */

```

tpccCom/dlldata.c

```

// wrapper for dlldata.c
#ifdef _MERGE_PROXYSTUB // merge proxy stub DLL
#define REGISTER_PROXY_DLL //DllRegisterServer, etc.
#define _WIN32_WINNT 0x0500 //for Win2000, change it to 0x0400
for NT4 or Win95 with DCOM
#define USE_STUBLESS_PROXY //defined only with MIDL switch
/Oicf
#pragma comment(lib, "rpcns4.lib")
#pragma comment(lib, "rprct4.lib")
#define ENTRY_PREFIX Prx
#include "dlldata.c"
#include "tpccCom_p.c"
#endif // _MERGE_PROXYSTUB

```

tpccCom/tpccCom_i.c

```

* this ALWAYS GENERATED file contains the IIDs and CLSIDs */
/* link this file in with the server and any clients */
/* File created by MIDL compiler version 6.00.0361 */
/* at Wed Feb 11 08:32:46 2004
*/
/* Compiler settings for .\tpccCom.idl:
Oicf, W1, Zp8, env=Win32 (32b run)
protocol : dce , ms_ext, c_ext, robust
error checks: allocation ref bounds_check enum stub_data
VC __declspec() decoration level:
__declspec(uuid()), __declspec(selectany), __declspec(novtable)
DECLSPEC_UUID(), MIDL_INTERFACE()
*/
//@@@MIDL_FILE_HEADING( )
#ifdef _M_IA64 && !defined(_M_AMD64)
#pragma warning( disable: 4049 ) /* more than 64k source lines */

```

```

#ifdef __cplusplus
extern "C"{
#endif
#include <rpc.h>
#include <rpcndr.h>
#ifdef _MIDL_USE_GUIDDEF_
#ifndef INITGUID
#define INITGUID
#include <guiddef.h>
#undef INITGUID
#else

```



```

#include <guiddef.h>
#endif
#define MIDL_DEFINE_GUID(type,name,l,w1,w2,b1,b2,b3,b4,b5,b6,b7,b8) \
    DEFINE_GUID(name,l,w1,w2,b1,b2,b3,b4,b5,b6,b7,b8)
#else // !_MIDL_USE_GUIDDEF_
#ifndef __IID_DEFINED__
#define __IID_DEFINED__
typedef struct _IID
{
    unsigned long x;
    unsigned short s1;
    unsigned short s2;
    unsigned char c[8];
} IID;
#endif // __IID_DEFINED__
#ifndef CLSID_DEFINED
#define CLSID_DEFINED
typedef IID CLSID;
#endif // CLSID_DEFINED
#define MIDL_DEFINE_GUID(type,name,l,w1,w2,b1,b2,b3,b4,b5,b6,b7,b8) \
    const type name = {l,w1,w2,{b1,b2,b3,b4,b5,b6,b7,b8}}
#endif !_MIDL_USE_GUIDDEF_
MIDL_DEFINE_GUID(IID,
IID_IComponentRegistrar,0xa817e7a2,0x43fa,0x11d0,0x9e,0x44,0x00,0xaa,0x00,0xb6,0x77,0x0a);

MIDL_DEFINE_GUID(IID,
IID_Itpcc_com,0x5B4FA473,0x2E68,0x4D79,0xA6,0x26,0xF3,0x8B,0x30,0xB8,0x19,0x6E);

MIDL_DEFINE_GUID(IID,
LIBID_tpccComLib,0x91F1B8B0,0x89E9,0x457B,0xA2,0x28,0x3E,0x2D,0x06,0xC,0xE3,0xE7,0x52);

MIDL_DEFINE_GUID(CLSID,
CLSID_CompReg,0x90EEDAFF,0xF8D3,0x4711,0x99,0xA9,0x8A,0xC3,0xC0,0xFE,0x5D,0xB9);

MIDL_DEFINE_GUID(CLSID,
CLSID_tpcc_com,0x5F752BF2,0xF739,0x43D4,0x84,0x92,0x44,0xC1,0x95,0x81,0xC0,0xA1);
#undef MIDL_DEFINE_GUID
#ifdef __cplusplus
}
#endif

#endif /* !defined(_M_IA64) && !defined(_M_AMD64)*/

```

tpccCom/tpccCom_p.c

```

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

/* File created by MIDL compiler version 6.00.0361 */
/* at Wed Feb 11 08:32:46 2004
*/
/* Compiler settings for \tpccCom.idl:
Oicf, W1, Zp8, env=Win32 (32b run)
protocol : dce , ms_ext, c_ext, robust
error checks: allocation ref bounds_check enum stub_data
VC __declspec() decoration level:
__declspec(uuid()), __declspec(selectany), __declspec(novtable)
DECLSPEC_UUID(), MIDL_INTERFACE()
*/
//@@@MIDL_FILE_HEADING( )
#ifdef !defined(_M_IA64) && !defined(_M_AMD64)

#pragma warning( disable: 4049 ) /* more than 64k source lines */
#if _MSC_VER >= 1200

```

```

#pragma warning(push)
#endif
#pragma warning( disable: 4100 ) /* unreferenced arguments in x86 call */
#pragma warning( disable: 4211 ) /* redefine extent to static */
#pragma warning( disable: 4232 ) /* dllimport identity*/
#define USE_STUBLESS_PROXY

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

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

#include "tpccCom.h"
#define TYPE_FORMAT_STRING_SIZE 1089
#define PROC_FORMAT_STRING_SIZE 409
#define TRANSMIT_AS_TABLE_SIZE 0
#define WIRE_MARSHAL_TABLE_SIZE 2
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;

static RPC_SYNTAX_IDENTIFIER _RpcTransferSyntax =
{{0x8A885D04,0x1CEB,0x11C9,{0x9F,0xE8,0x08,0x00,0x2B,0x10,0x48,0x60}},{2,0}};

extern const MIDL_TYPE_FORMAT_STRING __MIDL_TypeFormatString;
extern const MIDL_PROC_FORMAT_STRING __MIDL_ProcFormatString;

extern const MIDL_STUB_DESC Object_StubDesc;

extern const MIDL_SERVER_INFO IComponentRegistrar_ServerInfo;
extern const MIDL_STUBLESS_PROXY_INFO IComponentRegistrar_ProxyInfo;

extern const MIDL_STUB_DESC Object_StubDesc;

extern const MIDL_SERVER_INFO Itpcc_com_ServerInfo;
extern const MIDL_STUBLESS_PROXY_INFO Itpcc_com_ProxyInfo;

extern const USER_MARSHAL_ROUTINE_QUADRUPLE
UserMarshalRoutines[ WIRE_MARSHAL_TABLE_SIZE ];
#ifdef !defined(__RPC_WIN32__)
#error Invalid build platform for this stub.
#endif
#ifdef !(TARGET_IS_NT50_OR_LATER)
#error You need a Windows 2000 or later to run this stub because it uses these
features:
#error /robust command line switch.
#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 Attach */
        0x33, /*
FC_AUTO_HANDLE */
        0x6c, /* Old Flags: object,
Oi2 */
        /* 2 */ NdrFcLong( 0x0 ), /* 0 */
        /* 6 */ NdrFcShort( 0x7 ), /* 7 */
        /* 8 */ NdrFcShort( 0xc ), /* x86 Stack size/offset = 12 */
        /* 10 */ NdrFcShort( 0x0 ), /* 0 */
        /* 12 */ NdrFcShort( 0x8 ), /* 8 */
        /* 14 */ 0x46, /* Oi2 Flags: clt must size, has return, has
ext, */
        0x2, /* 2 */
        /* 16 */ 0x8, /* 8 */
        0x5, /* Ext Flags: new corr
desc, srv corr check, */
        /* 18 */ NdrFcShort( 0x0 ), /* 0 */
        /* 20 */ NdrFcShort( 0x1 ), /* 1 */
        /* 22 */ NdrFcShort( 0x0 ), /* 0 */
        /* Parameter bstrPath */
        /* 24 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by val, */
        /* 26 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
        /* 28 */ NdrFcShort( 0x1c ), /* Type Offset=28 */
        /* Return value */
        /* 30 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
        /* 32 */ NdrFcShort( 0x8 ), /* x86 Stack size/offset = 8 */
        /* 34 */ 0x8, /* FC_LONG */
        0x0, /* 0 */
        /* Procedure doSetComplete */
        /* Procedure RegisterAll */
        /* 36 */ 0x33, /* FC_AUTO_HANDLE */
        0x6c, /* Old Flags: object,
Oi2 */
        /* 38 */ NdrFcLong( 0x0 ), /* 0 */
        /* 42 */ NdrFcShort( 0x8 ), /* 8 */
        /* 44 */ NdrFcShort( 0x8 ), /* x86 Stack size/offset = 8 */
        /* 46 */ NdrFcShort( 0x0 ), /* 0 */
        /* 48 */ NdrFcShort( 0x8 ), /* 8 */
        /* 50 */ 0x44, /* Oi2 Flags: has return, has ext, */
        0x1, /* 1 */
        /* 52 */ 0x8, /* 8 */
        0x1, /* Ext Flags: new corr
desc, */
        /* 54 */ NdrFcShort( 0x0 ), /* 0 */
        /* 56 */ NdrFcShort( 0x0 ), /* 0 */
        /* 58 */ NdrFcShort( 0x0 ), /* 0 */
        /* Return value */
        /* Return value */
        /* 60 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
        /* 62 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
        /* 64 */ 0x8, /* FC_LONG */
        0x0, /* 0 */
        /* Procedure UnregisterAll */
        /* 66 */ 0x33, /* FC_AUTO_HANDLE */
        0x6c, /* Old Flags: object,
Oi2 */
        /* 68 */ NdrFcLong( 0x0 ), /* 0 */
        /* 72 */ NdrFcShort( 0x9 ), /* 9 */
        /* 74 */ NdrFcShort( 0x8 ), /* x86 Stack size/offset = 8 */
        /* 76 */ NdrFcShort( 0x0 ), /* 0 */
        /* 78 */ NdrFcShort( 0x8 ), /* 8 */
        /* 80 */ 0x44, /* Oi2 Flags: has return, has ext, */
        0x1, /* 1 */
        /* 82 */ 0x8, /* 8 */
        0x1, /* Ext Flags: new corr
desc, */
        /* 84 */ NdrFcShort( 0x0 ), /* 0 */
        /* 86 */ NdrFcShort( 0x0 ), /* 0 */
        /* 88 */ NdrFcShort( 0x0 ), /* 0 */
        /* Return value */
        /* 90 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
        /* 92 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
        /* 94 */ 0x8, /* FC_LONG */
        0x0, /* 0 */
        /* Procedure GetComponent */
        /* 96 */ 0x33, /* FC_AUTO_HANDLE */
        0x6c, /* Old Flags: object,
Oi2 */
        /* 98 */ NdrFcLong( 0x0 ), /* 0 */
        /* 102 */ NdrFcShort( 0xa ), /* 10 */
        /* 104 */ NdrFcShort( 0x10 ), /* x86 Stack size/offset = 16 */
        /* 106 */ NdrFcShort( 0x0 ), /* 0 */
        /* 108 */ NdrFcShort( 0x8 ), /* 8 */
        /* 110 */ 0x45, /* Oi2 Flags: srv must size, has return, has
ext, */
        0x3, /* 3 */
        /* 112 */ 0x8, /* 8 */
        0x3, /* Ext Flags: new corr
desc, clt corr check, */
        /* 114 */ NdrFcShort( 0x24 ), /* 36 */
        /* 116 */ NdrFcShort( 0x0 ), /* 0 */
        /* 118 */ NdrFcShort( 0x0 ), /* 0 */
        /* Parameter pbstrCLSIDs */
        /* 120 */ NdrFcShort( 0x2113 ), /* Flags: must size, must free, out,
simple ref, srv alloc size=8 */
        /* 122 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
        /* 124 */ NdrFcShort( 0x41e ), /* Type Offset=1054 */
        /* Parameter pbstrDescriptions */
        /* 126 */ NdrFcShort( 0x2113 ), /* Flags: must size, must free, out,
simple ref, srv alloc size=8 */
        /* 128 */ NdrFcShort( 0x8 ), /* x86 Stack size/offset = 8 */
        /* 130 */ NdrFcShort( 0x41e ), /* Type Offset=1054 */
        /* Return value */
        /* 132 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
        /* 134 */ NdrFcShort( 0xc ), /* x86 Stack size/offset = 12 */
        /* 136 */ 0x8, /* FC_LONG */
        0x0, /* 0 */
        /* Procedure RegisterComponent */
        /* 138 */ 0x33, /* FC_AUTO_HANDLE */
        0x6c, /* Old Flags: object,
Oi2 */
        /* 140 */ NdrFcLong( 0x0 ), /* 0 */
        /* 144 */ NdrFcShort( 0xb ), /* 11 */
        /* 146 */ NdrFcShort( 0xc ), /* x86 Stack size/offset = 12 */
        /* 148 */ NdrFcShort( 0x0 ), /* 0 */
        /* 150 */ NdrFcShort( 0x8 ), /* 8 */
        /* 152 */ 0x46, /* Oi2 Flags: clt must size, has return, has
ext, */
        0x2, /* 2 */
        /* 154 */ 0x8, /* 8 */
        0x5, /* Ext Flags: new corr
desc, srv corr check, */
        /* 156 */ NdrFcShort( 0x0 ), /* 0 */
        /* 158 */ NdrFcShort( 0x1 ), /* 1 */
        /* 160 */ NdrFcShort( 0x0 ), /* 0 */
        /* Parameter bstrCLSID */
        /* 162 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by val, */
        /* 164 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
        /* 166 */ NdrFcShort( 0x1c ), /* Type Offset=28 */
        /* Return value */

```

```

/* 168 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
/* 170 */ NdrFcShort( 0x8 ), /* x86 Stack size/offset = 8 */
/* 172 */ 0x8, /* FC_LONG */
0x0, /* 0 */
/* Procedure UnregisterComponent */
/* 174 */ 0x33, /* FC_AUTO_HANDLE */
0x6c, /* Old Flags: object,
Oi2 */
/* 176 */ NdrFcLong( 0x0 ), /* 0 */
/* 180 */ NdrFcShort( 0xc ), /* 12 */
/* 182 */ NdrFcShort( 0xc ), /* x86 Stack size/offset = 12 */
/* 184 */ NdrFcShort( 0x0 ), /* 0 */
/* 186 */ NdrFcShort( 0x8 ), /* 8 */
/* 188 */ 0x46, /* Oi2 Flags: clt must size, has return, has
ext, */
0x2, /* 2 */
/* 190 */ 0x8, /* 8 */
0x5, /* Ext Flags: new corr
desc, srv corr check, */
/* 192 */ NdrFcShort( 0x0 ), /* 0 */
/* 194 */ NdrFcShort( 0x1 ), /* 1 */
/* 196 */ NdrFcShort( 0x0 ), /* 0 */
/* Parameter bstrCLSID */
/* 198 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by val, */
/* 200 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
/* 202 */ NdrFcShort( 0x1c ), /* Type Offset=28 */
/* Return value */
/* 204 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
/* 206 */ NdrFcShort( 0x8 ), /* x86 Stack size/offset = 8 */
/* 208 */ 0x8, /* FC_LONG */
0x0, /* 0 */
/* Procedure doStockLevel */
/* 210 */ 0x33, /* FC_AUTO_HANDLE */
0x6c, /* Old Flags: object,
Oi2 */
/* 212 */ NdrFcLong( 0x0 ), /* 0 */
/* 216 */ NdrFcShort( 0x3 ), /* 3 */
/* 218 */ NdrFcShort( 0x10 ), /* x86 Stack size/offset = 16 */
/* 220 */ NdrFcShort( 0x1c ), /* 28 */
/* 222 */ NdrFcShort( 0x8 ), /* 8 */
/* 224 */ 0x47, /* Oi2 Flags: srv must size, clt must size, has
return, has ext, */
0x3, /* 3 */
/* 226 */ 0x8, /* 8 */
0x7, /* Ext Flags: new corr
desc, clt corr check, srv corr check, */
/* 228 */ NdrFcShort( 0x1 ), /* 1 */
/* 230 */ NdrFcShort( 0x1 ), /* 1 */
/* 232 */ NdrFcShort( 0x0 ), /* 0 */
/* Parameter size */
/* 234 */ NdrFcShort( 0x148 ), /* Flags: in, base type, simple ref, */
/* 236 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
/* 238 */ 0x8, /* FC_LONG */
0x0, /* 0 */
/* Parameter buffer */
/* 240 */ NdrFcShort( 0x201b ), /* Flags: must size, must free, in,
out, srv alloc size=8 */
/* 242 */ NdrFcShort( 0x8 ), /* x86 Stack size/offset = 8 */
/* 244 */ NdrFcShort( 0x42c ), /* Type Offset=1068 */
/* Return value */
/* 246 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
/* 248 */ NdrFcShort( 0xc ), /* x86 Stack size/offset = 12 */
/* 250 */ 0x8, /* FC_LONG */
0x0, /* 0 */
/* Procedure doNewOrder */
/* 252 */ 0x33, /* FC_AUTO_HANDLE */
0x6c, /* Old Flags: object,
Oi2 */
/* 254 */ NdrFcLong( 0x0 ), /* 0 */
/* 258 */ NdrFcShort( 0x4 ), /* 4 */
/* 260 */ NdrFcShort( 0x10 ), /* x86 Stack size/offset = 16 */
/* 262 */ NdrFcShort( 0x1c ), /* 28 */
/* 264 */ NdrFcShort( 0x8 ), /* 8 */
/* 266 */ 0x47, /* Oi2 Flags: srv must size, clt must size, has
return, has ext, */
0x3, /* 3 */
/* 268 */ 0x8, /* 8 */
0x7, /* Ext Flags: new corr
desc, clt corr check, srv corr check, */
/* 270 */ NdrFcShort( 0x1 ), /* 1 */
/* 272 */ NdrFcShort( 0x1 ), /* 1 */
/* 274 */ NdrFcShort( 0x0 ), /* 0 */
/* Parameter size */
/* 276 */ NdrFcShort( 0x148 ), /* Flags: in, base type, simple ref, */
/* 278 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
/* 280 */ 0x8, /* FC_LONG */
0x0, /* 0 */
/* Parameter buffer */
/* 282 */ NdrFcShort( 0x201b ), /* Flags: must size, must free, in,
out, srv alloc size=8 */
/* 284 */ NdrFcShort( 0x8 ), /* x86 Stack size/offset = 8 */
/* 286 */ NdrFcShort( 0x42c ), /* Type Offset=1068 */
/* Return value */
/* 288 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
/* 290 */ NdrFcShort( 0xc ), /* x86 Stack size/offset = 12 */
/* 292 */ 0x8, /* FC_LONG */
0x0, /* 0 */
/* Procedure doPayment */
/* 294 */ 0x33, /* FC_AUTO_HANDLE */
0x6c, /* Old Flags: object,
Oi2 */
/* 296 */ NdrFcLong( 0x0 ), /* 0 */
/* 300 */ NdrFcShort( 0x5 ), /* 5 */
/* 302 */ NdrFcShort( 0x10 ), /* x86 Stack size/offset = 16 */
/* 304 */ NdrFcShort( 0x1c ), /* 28 */
/* 306 */ NdrFcShort( 0x8 ), /* 8 */
/* 308 */ 0x47, /* Oi2 Flags: srv must size, clt must size, has
return, has ext, */
0x3, /* 3 */
/* 310 */ 0x8, /* 8 */
0x7, /* Ext Flags: new corr
desc, clt corr check, srv corr check, */
/* 312 */ NdrFcShort( 0x1 ), /* 1 */
/* 314 */ NdrFcShort( 0x1 ), /* 1 */
/* 316 */ NdrFcShort( 0x0 ), /* 0 */
/* Parameter size */
/* 318 */ NdrFcShort( 0x148 ), /* Flags: in, base type, simple ref, */
/* 320 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
/* 322 */ 0x8, /* FC_LONG */
0x0, /* 0 */
/* Parameter buffer */
/* 324 */ NdrFcShort( 0x201b ), /* Flags: must size, must free, in,
out, srv alloc size=8 */
/* 326 */ NdrFcShort( 0x8 ), /* x86 Stack size/offset = 8 */
/* 328 */ NdrFcShort( 0x42c ), /* Type Offset=1068 */
/* Return value */
/* 330 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
/* 332 */ NdrFcShort( 0xc ), /* x86 Stack size/offset = 12 */
/* 334 */ 0x8, /* FC_LONG */
0x0, /* 0 */
/* Procedure doOrderStatus */
/* 336 */ 0x33, /* FC_AUTO_HANDLE */
0x6c, /* Old Flags: object,
Oi2 */
/* 338 */ NdrFcLong( 0x0 ), /* 0 */
/* 342 */ NdrFcShort( 0x6 ), /* 6 */

```

```

/* 344 */ NdrFcShort( 0x10 ), /* x86 Stack size/offset = 16 */
/* 346 */ NdrFcShort( 0x1c ), /* 28 */
/* 348 */ NdrFcShort( 0x8 ), /* 8 */
/* 350 */ 0x47, /* Oi2 Flags: srv must size, clt must size, has
return, has ext, */
                                0x3, /* 3 */
/* 352 */ 0x8, /* 8 */
                                0x7, /* Ext Flags: new corr
desc, clt corr check, srv corr check, */
/* 354 */ NdrFcShort( 0x1 ), /* 1 */
/* 356 */ NdrFcShort( 0x1 ), /* 1 */
/* 358 */ NdrFcShort( 0x0 ), /* 0 */
/* Parameter size */
/* 360 */ NdrFcShort( 0x148 ), /* Flags: in, base type, simple ref, */
/* 362 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
/* 364 */ 0x8, /* FC_LONG */
                                0x0, /* 0 */
/* Parameter buffer */
/* 366 */ NdrFcShort( 0x201b ), /* Flags: must size, must free, in,
out, srv alloc size=8 */
/* 368 */ NdrFcShort( 0x8 ), /* x86 Stack size/offset = 8 */
/* 370 */ NdrFcShort( 0x42c ), /* Type Offset=1068 */
/* Return value */
/* 372 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
/* 374 */ NdrFcShort( 0xc ), /* x86 Stack size/offset = 12 */
/* 376 */ 0x8, /* FC_LONG */
                                0x0, /* 0 */
/* Procedure doDBInfo */
/* 378 */ 0x33, /* FC_AUTO_HANDLE */
                                0x6c, /* Old Flags: object,
Oi2 */
/* 380 */ NdrFcLong( 0x0 ), /* 0 */
/* 384 */ NdrFcShort( 0x7 ), /* 7 */
/* 386 */ NdrFcShort( 0x8 ), /* x86 Stack size/offset = 8 */
/* 388 */ NdrFcShort( 0x0 ), /* 0 */
/* 390 */ NdrFcShort( 0x8 ), /* 8 */
/* 392 */ 0x44, /* Oi2 Flags: has return, has ext, */
                                0x1, /* 1 */
/* 394 */ 0x8, /* 8 */
                                0x1, /* Ext Flags: new corr
desc, */
/* 396 */ NdrFcShort( 0x0 ), /* 0 */
/* 398 */ NdrFcShort( 0x0 ), /* 0 */
/* 400 */ NdrFcShort( 0x0 ), /* 0 */
/* Return value */
/* 402 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
/* 404 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
/* 406 */ 0x8, /* FC_LONG */
                                0x0, /* 0 */
                                0x0
}
};
static const MIDL_TYPE_FORMAT_STRING __MIDL_TypeFormatString =
{
0,
{
/* 2 */ NdrFcShort( 0x0 ), /* 0 */
                                0x12, 0x0, /* FC_UP */
/* 4 */ NdrFcShort( 0xe ), /* Offset= 14 (18) */
/* 6 */
                                0x1b, /* FC_CARRAY */
                                0x1, /* 1 */
/* 8 */ NdrFcShort( 0x2 ), /* 2 */
/* 10 */ 0x9, /* Corr desc: FC_ULONG */
                                0x0, /* */
/* 12 */ NdrFcShort( 0xffff ), /* -4 */
/* 14 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
                                0x6, /* FC_SHORT */
                                0x5b, /* FC_END */
/* 18 */
                                0x17, /* FC_CSTRUCT */
                                0x3, /* 3 */
/* 20 */ NdrFcShort( 0x8 ), /* 8 */
/* 22 */ NdrFcShort( 0xffff0 ), /* Offset= -16 (6) */
/* 24 */ 0x8, /* FC_LONG */
                                0x8, /* FC_LONG */
/* 26 */ 0x5c, /* FC_PAD */
                                0x5b, /* FC_END */
/* 28 */ 0xb4, /* FC_USER_MARSHAL */
                                0x83, /* 131 */
/* 30 */ NdrFcShort( 0x0 ), /* 0 */
/* 32 */ NdrFcShort( 0x4 ), /* 4 */
/* 34 */ NdrFcShort( 0x0 ), /* 0 */
/* 36 */ NdrFcShort( 0xffde ), /* Offset= -34 (2) */
/* 38 */
                                0x11, 0x4, /* FC_RP [allocated_on_stack] */
/* 40 */ NdrFcShort( 0x3f6 ), /* Offset= 1014 (1054) */
/* 42 */
                                0x13, 0x10, /* FC_OP
[pointer_deref] */
/* 44 */ NdrFcShort( 0x2 ), /* Offset= 2 (46) */
/* 46 */
                                0x13, 0x0, /* FC_OP */
/* 48 */ NdrFcShort( 0x3dc ), /* Offset= 988 (1036) */
/* 50 */
                                0x2a, /*
FC_ENCAPSULATED_UNION */
                                0x49, /* 73 */
/* 52 */ NdrFcShort( 0x18 ), /* 24 */
/* 54 */ NdrFcShort( 0xa ), /* 10 */
/* 56 */ NdrFcLong( 0x8 ), /* 8 */
/* 60 */ NdrFcShort( 0x5a ), /* Offset= 90 (150) */
/* 62 */ NdrFcLong( 0xd ), /* 13 */
/* 66 */ NdrFcShort( 0x90 ), /* Offset= 144 (210) */
/* 68 */ NdrFcLong( 0x9 ), /* 9 */
/* 72 */ NdrFcShort( 0xc2 ), /* Offset= 194 (266) */
/* 74 */ NdrFcLong( 0xc ), /* 12 */
/* 78 */ NdrFcShort( 0x2c0 ), /* Offset= 704 (782) */
/* 80 */ NdrFcLong( 0x24 ), /* 36 */
/* 84 */ NdrFcShort( 0x2ea ), /* Offset= 746 (830) */
/* 86 */ NdrFcLong( 0x800d ), /* 32781 */
/* 90 */ NdrFcShort( 0x306 ), /* Offset= 774 (864) */
/* 92 */ NdrFcLong( 0x10 ), /* 16 */
/* 96 */ NdrFcShort( 0x320 ), /* Offset= 800 (896) */
/* 98 */ NdrFcLong( 0x2 ), /* 2 */
/* 102 */ NdrFcShort( 0x33a ), /* Offset= 826 (928) */
/* 104 */ NdrFcLong( 0x3 ), /* 3 */
/* 108 */ NdrFcShort( 0x354 ), /* Offset= 852 (960) */
/* 110 */ NdrFcLong( 0x14 ), /* 20 */
/* 114 */ NdrFcShort( 0x36e ), /* Offset= 878 (992) */
/* 116 */ NdrFcShort( 0xffff ), /* Offset= -1 (115) */
/* 118 */
                                0x1b, /* FC_CARRAY */
                                0x3, /* 3 */
/* 120 */ NdrFcShort( 0x4 ), /* 4 */
/* 122 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
                                0x0, /* */
/* 124 */ NdrFcShort( 0x0 ), /* 0 */
/* 126 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 128 */
                                0x4b, /* FC_PP */
                                0x5c, /* FC_PAD */
/* 130 */
                                0x48, /*
FC_VARIABLE_REPEAT */

```

```

0x49, /*
FC_FIXED_OFFSET */
/* 132 */ NdrFcShort( 0x4 ), /* 4 */
/* 134 */ NdrFcShort( 0x0 ), /* 0 */
/* 136 */ NdrFcShort( 0x1 ), /* 1 */
/* 138 */ NdrFcShort( 0x0 ), /* 0 */
/* 140 */ NdrFcShort( 0x0 ), /* 0 */
/* 142 */ 0x13, 0x0, /* FC_OP */
/* 144 */ NdrFcShort( 0xff82 ), /* Offset= -126 (18) */
/* 146 */
0x5b, /* FC_END */
0x8, /* FC_LONG */
/* 148 */ 0x5c, /* FC_PAD */
0x5b, /* FC_END */
/* 150 */
0x16, /* FC_PSTRUCT */
0x3, /* 3 */
/* 152 */ NdrFcShort( 0x8 ), /* 8 */
/* 154 */
0x4b, /* FC_PP */
0x5c, /* FC_PAD */
/* 156 */
0x46, /* FC_NO_REPEAT */
0x5c, /* FC_PAD */
/* 158 */ NdrFcShort( 0x4 ), /* 4 */
/* 160 */ NdrFcShort( 0x4 ), /* 4 */
/* 162 */ 0x11, 0x0, /* FC_RP */
/* 164 */ NdrFcShort( 0xffd2 ), /* Offset= -46 (118) */
/* 166 */
0x5b, /* FC_END */
0x8, /* FC_LONG */
/* 168 */ 0x8, /* FC_LONG */
0x5b, /* FC_END */
/* 170 */
0x2f, /* FC_IP */
0x5a, /*
FC_CONSTANT_IID */
/* 172 */ NdrFcLong( 0x0 ), /* 0 */
/* 176 */ NdrFcShort( 0x0 ), /* 0 */
/* 178 */ NdrFcShort( 0x0 ), /* 0 */
/* 180 */ 0xc0, /* 192 */
0x0, /* 0 */
/* 182 */ 0x0, /* 0 */
0x0, /* 0 */
/* 184 */ 0x0, /* 0 */
0x0, /* 0 */
/* 186 */ 0x0, /* 0 */
0x46, /* 70 */
/* 188 */
0x21, /*
FC_BOGUS_ARRAY */
0x3, /* 3 */
/* 190 */ NdrFcShort( 0x0 ), /* 0 */
/* 192 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
0x0, /*
/* 194 */ NdrFcShort( 0x0 ), /* 0 */
/* 196 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 198 */ NdrFcLong( 0xffffffff ), /* -1 */
/* 202 */ NdrFcShort( 0x0 ), /* Corr flags: */
/* 204 */ 0x4c, /* FC_EMBEDDED_COMPLEX */
0x0, /* 0 */
/* 206 */ NdrFcShort( 0xffdc ), /* Offset= -36 (170) */
/* 208 */ 0x5c, /* FC_PAD */
0x5b, /* FC_END */
/* 210 */
0x1a, /*
FC_BOGUS_STRUCT */
0x3, /* 3 */
/* 212 */ NdrFcShort( 0x8 ), /* 8 */
/* 214 */ NdrFcShort( 0x0 ), /* 0 */
/* 216 */ NdrFcShort( 0x6 ), /* Offset= 6 (222) */
/* 218 */ 0x8, /* FC_LONG */
0x36, /* FC_POINTER */
/* 220 */ 0x5c, /* FC_PAD */
0x5b, /* FC_END */
/* 222 */
0x11, 0x0, /* FC_RP */
/* 224 */ NdrFcShort( 0xffdc ), /* Offset= -36 (188) */
/* 226 */
0x2f, /* FC_IP */
0x5a, /*
FC_CONSTANT_IID */
/* 228 */ NdrFcLong( 0x20400 ), /* 132096 */
/* 232 */ NdrFcShort( 0x0 ), /* 0 */
/* 234 */ NdrFcShort( 0x0 ), /* 0 */
/* 236 */ 0xc0, /* 192 */
0x0, /* 0 */
/* 238 */ 0x0, /* 0 */
0x0, /* 0 */
/* 240 */ 0x0, /* 0 */
0x0, /* 0 */
/* 242 */ 0x0, /* 0 */
0x46, /* 70 */
/* 244 */
0x21, /*
FC_BOGUS_ARRAY */
0x3, /* 3 */
/* 246 */ NdrFcShort( 0x0 ), /* 0 */
/* 248 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
0x0, /*
/* 250 */ NdrFcShort( 0x0 ), /* 0 */
/* 252 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 254 */ NdrFcLong( 0xffffffff ), /* -1 */
/* 258 */ NdrFcShort( 0x0 ), /* Corr flags: */
/* 260 */ 0x4c, /* FC_EMBEDDED_COMPLEX */
0x0, /* 0 */
/* 262 */ NdrFcShort( 0xffdc ), /* Offset= -36 (226) */
/* 264 */ 0x5c, /* FC_PAD */
0x5b, /* FC_END */
/* 266 */
0x1a, /*
FC_BOGUS_STRUCT */
0x3, /* 3 */
/* 268 */ NdrFcShort( 0x8 ), /* 8 */
/* 270 */ NdrFcShort( 0x0 ), /* 0 */
/* 272 */ NdrFcShort( 0x6 ), /* Offset= 6 (278) */
/* 274 */ 0x8, /* FC_LONG */
0x36, /* FC_POINTER */
/* 276 */ 0x5c, /* FC_PAD */
0x5b, /* FC_END */
/* 278 */
0x11, 0x0, /* FC_RP */
/* 280 */ NdrFcShort( 0xffdc ), /* Offset= -36 (244) */
/* 282 */
0x2b, /*
FC_NON_ENCAPSULATED_UNION */
/* 284 */ 0x7, /* Corr desc: FC_USHORT */
0x0, /*
/* 286 */ NdrFcShort( 0xff8 ), /* -8 */
/* 288 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 290 */ NdrFcShort( 0x2 ), /* Offset= 2 (292) */
/* 292 */ NdrFcShort( 0x10 ), /* 16 */
/* 294 */ NdrFcShort( 0x2f ), /* 47 */
/* 296 */ NdrFcLong( 0x14 ), /* 20 */
/* 300 */ NdrFcShort( 0x800b ), /* Simple arm type: FC_HYPER */

```

```

/* 302 */ NdrFcLong( 0x3 ), /* 3 */
/* 306 */ NdrFcShort( 0x8008 ), /* Simple arm type: FC_LONG */
/* 308 */ NdrFcLong( 0x11 ), /* 17 */
/* 312 */ NdrFcShort( 0x8001 ), /* Simple arm type: FC_BYTE */
/* 314 */ NdrFcLong( 0x2 ), /* 2 */
/* 318 */ NdrFcShort( 0x8006 ), /* Simple arm type: FC_SHORT */
/* 320 */ NdrFcLong( 0x4 ), /* 4 */
/* 324 */ NdrFcShort( 0x800a ), /* Simple arm type: FC_FLOAT */
/* 326 */ NdrFcLong( 0x5 ), /* 5 */
/* 330 */ NdrFcShort( 0x800c ), /* Simple arm type: FC_DOUBLE
*/
/* 332 */ NdrFcLong( 0xb ), /* 11 */
/* 336 */ NdrFcShort( 0x8006 ), /* Simple arm type: FC_SHORT */
/* 338 */ NdrFcLong( 0xa ), /* 10 */
/* 342 */ NdrFcShort( 0x8008 ), /* Simple arm type: FC_LONG */
/* 344 */ NdrFcLong( 0x6 ), /* 6 */
/* 348 */ NdrFcShort( 0xe8 ), /* Offset= 232 (580) */
/* 350 */ NdrFcLong( 0x7 ), /* 7 */
/* 354 */ NdrFcShort( 0x800c ), /* Simple arm type: FC_DOUBLE
*/
/* 356 */ NdrFcLong( 0x8 ), /* 8 */
/* 360 */ NdrFcShort( 0xe2 ), /* Offset= 226 (586) */
/* 362 */ NdrFcLong( 0xd ), /* 13 */
/* 366 */ NdrFcShort( 0xff3c ), /* Offset= -196 (170) */
/* 368 */ NdrFcLong( 0x9 ), /* 9 */
/* 372 */ NdrFcShort( 0xff6e ), /* Offset= -146 (226) */
/* 374 */ NdrFcLong( 0x2000 ), /* 8192 */
/* 378 */ NdrFcShort( 0xd4 ), /* Offset= 212 (590) */
/* 380 */ NdrFcLong( 0x24 ), /* 36 */
/* 384 */ NdrFcShort( 0xd6 ), /* Offset= 214 (598) */
/* 386 */ NdrFcLong( 0x4024 ), /* 16420 */
/* 390 */ NdrFcShort( 0xd0 ), /* Offset= 208 (598) */
/* 392 */ NdrFcLong( 0x4011 ), /* 16401 */
/* 396 */ NdrFcShort( 0x100 ), /* Offset= 256 (652) */
/* 398 */ NdrFcLong( 0x4002 ), /* 16386 */
/* 402 */ NdrFcShort( 0xfe ), /* Offset= 254 (656) */
/* 404 */ NdrFcLong( 0x4003 ), /* 16387 */
/* 408 */ NdrFcShort( 0xfc ), /* Offset= 252 (660) */
/* 410 */ NdrFcLong( 0x4014 ), /* 16404 */
/* 414 */ NdrFcShort( 0xfa ), /* Offset= 250 (664) */
/* 416 */ NdrFcLong( 0x4004 ), /* 16388 */
/* 420 */ NdrFcShort( 0xf8 ), /* Offset= 248 (668) */
/* 422 */ NdrFcLong( 0x4005 ), /* 16389 */
/* 426 */ NdrFcShort( 0xf6 ), /* Offset= 246 (672) */
/* 428 */ NdrFcLong( 0x400b ), /* 16395 */
/* 432 */ NdrFcShort( 0xe0 ), /* Offset= 224 (656) */
/* 434 */ NdrFcLong( 0x400a ), /* 16394 */
/* 438 */ NdrFcShort( 0xde ), /* Offset= 222 (660) */
/* 440 */ NdrFcLong( 0x4006 ), /* 16390 */
/* 444 */ NdrFcShort( 0xe8 ), /* Offset= 232 (676) */
/* 446 */ NdrFcLong( 0x4007 ), /* 16391 */
/* 450 */ NdrFcShort( 0xde ), /* Offset= 222 (672) */
/* 452 */ NdrFcLong( 0x4008 ), /* 16392 */
/* 456 */ NdrFcShort( 0xe0 ), /* Offset= 224 (680) */
/* 458 */ NdrFcLong( 0x400d ), /* 16397 */
/* 462 */ NdrFcShort( 0xde ), /* Offset= 222 (684) */
/* 464 */ NdrFcLong( 0x4009 ), /* 16393 */
/* 468 */ NdrFcShort( 0xdc ), /* Offset= 220 (688) */
/* 470 */ NdrFcLong( 0x6000 ), /* 24576 */
/* 474 */ NdrFcShort( 0xda ), /* Offset= 218 (692) */
/* 476 */ NdrFcLong( 0x400c ), /* 16396 */
/* 480 */ NdrFcShort( 0xe0 ), /* Offset= 224 (704) */
/* 482 */ NdrFcLong( 0x10 ), /* 16 */
/* 486 */ NdrFcShort( 0x8002 ), /* Simple arm type: FC_CHAR */
/* 488 */ NdrFcLong( 0x12 ), /* 18 */
/* 492 */ NdrFcShort( 0x8006 ), /* Simple arm type: FC_SHORT */
/* 494 */ NdrFcLong( 0x13 ), /* 19 */
/* 498 */ NdrFcShort( 0x8008 ), /* Simple arm type: FC_LONG */

/* 500 */ NdrFcLong( 0x15 ), /* 21 */
/* 504 */ NdrFcShort( 0x800b ), /* Simple arm type: FC_HYPER */
/* 506 */ NdrFcLong( 0x16 ), /* 22 */
/* 510 */ NdrFcShort( 0x8008 ), /* Simple arm type: FC_LONG */
/* 512 */ NdrFcLong( 0x17 ), /* 23 */
/* 516 */ NdrFcShort( 0x8008 ), /* Simple arm type: FC_LONG */
/* 518 */ NdrFcLong( 0xe ), /* 14 */
/* 522 */ NdrFcShort( 0xbe ), /* Offset= 190 (712) */
/* 524 */ NdrFcLong( 0x400e ), /* 16398 */
/* 528 */ NdrFcShort( 0xc2 ), /* Offset= 194 (722) */
/* 530 */ NdrFcLong( 0x4010 ), /* 16400 */
/* 534 */ NdrFcShort( 0xc0 ), /* Offset= 192 (726) */
/* 536 */ NdrFcLong( 0x4012 ), /* 16402 */
/* 540 */ NdrFcShort( 0x74 ), /* Offset= 116 (656) */
/* 542 */ NdrFcLong( 0x4013 ), /* 16403 */
/* 546 */ NdrFcShort( 0x72 ), /* Offset= 114 (660) */
/* 548 */ NdrFcLong( 0x4015 ), /* 16405 */
/* 552 */ NdrFcShort( 0x70 ), /* Offset= 112 (664) */
/* 554 */ NdrFcLong( 0x4016 ), /* 16406 */
/* 558 */ NdrFcShort( 0x66 ), /* Offset= 102 (660) */
/* 560 */ NdrFcLong( 0x4017 ), /* 16407 */
/* 564 */ NdrFcShort( 0x60 ), /* Offset= 96 (660) */
/* 566 */ NdrFcLong( 0x0 ), /* 0 */
/* 570 */ NdrFcShort( 0x0 ), /* Offset= 0 (570) */
/* 572 */ NdrFcLong( 0x1 ), /* 1 */
/* 576 */ NdrFcShort( 0x0 ), /* Offset= 0 (576) */
/* 578 */ NdrFcShort( 0xffff ), /* Offset= -1 (577) */
/* 580 */
0x15, /* FC_STRUCT */
0x7, /* 7 */
/* 582 */ NdrFcShort( 0x8 ), /* 8 */
/* 584 */ 0xb, /* FC_HYPER */
0x5b, /* FC_END */

/* 586 */
0x13, 0x0, /* FC_OP */
/* 588 */ NdrFcShort( 0xfdc6 ), /* Offset= -570 (18) */
/* 590 */
0x13, 0x10, /* FC_OP
[pointer_deref] */
/* 592 */ NdrFcShort( 0x2 ), /* Offset= 2 (594) */
/* 594 */
0x13, 0x0, /* FC_OP */
/* 596 */ NdrFcShort( 0x1b8 ), /* Offset= 440 (1036) */
/* 598 */
0x13, 0x0, /* FC_OP */
/* 600 */ NdrFcShort( 0x20 ), /* Offset= 32 (632) */
/* 602 */
0x2f, /* FC_IP */
0x5a, /*
FC_CONSTANT_IID */
/* 604 */ NdrFcLong( 0x2f ), /* 47 */
/* 608 */ NdrFcShort( 0x0 ), /* 0 */
/* 610 */ NdrFcShort( 0x0 ), /* 0 */
/* 612 */ 0xc0, /* 192 */
0x0, /* 0 */
/* 614 */ 0x0, /* 0 */
0x0, /* 0 */
/* 616 */ 0x0, /* 0 */
0x0, /* 0 */
/* 618 */ 0x0, /* 0 */
0x46, /* 70 */
/* 620 */
0x1b, /* FC_CARRAY */
0x0, /* 0 */
/* 622 */ NdrFcShort( 0x1 ), /* 1 */
/* 624 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
0x0, /* */
/* 626 */ NdrFcShort( 0x4 ), /* 4 */

```

```

/* 628 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 630 */ 0x1, /* FC_BYTE */
/* 632 */ 0x5b, /* FC_END */
FC_BOGUS_STRUCT */
0x1a, /*
/* 634 */ NdrFcShort( 0x10 ), /* 16 */
/* 636 */ NdrFcShort( 0x0 ), /* 0 */
/* 638 */ NdrFcShort( 0xa ), /* Offset= 10 (648) */
/* 640 */ 0x8, /* FC_LONG */
/* 642 */ 0x4c, /* FC_LONG */
/* 644 */ NdrFcShort( 0xffd6 ), /* Offset= -42 (602) */
/* 646 */ 0x36, /* FC_POINTER */
/* 648 */ 0x5b, /* FC_END */
/* 650 */ NdrFcShort( 0xffe2 ), /* Offset= -30 (620) */
/* 652 */ 0x13, 0x0, /* FC_OP */
/* 654 */ 0x1, /* FC_BYTE [simple_pointer] */
/* 656 */ 0x5c, /* FC_PAD */
/* 658 */ 0x6, /* FC_SHORT [simple_pointer] */
/* 660 */ 0x5c, /* FC_PAD */
/* 662 */ 0x8, /* FC_LONG [simple_pointer] */
/* 664 */ 0x5c, /* FC_PAD */
/* 666 */ 0xb, /* FC_HYPER [simple_pointer] */
/* 668 */ 0x5c, /* FC_PAD */
/* 670 */ 0xa, /* FC_FLOAT [simple_pointer] */
/* 672 */ 0x5c, /* FC_PAD */
/* 674 */ 0xc, /* FC_DOUBLE [simple_pointer] */
/* 676 */ 0x5c, /* FC_PAD */
/* 678 */ NdrFcShort( 0xff9e ), /* Offset= -98 (580) */
/* 680 */ 0x13, 0x10, /* FC_OP
[pointer_deref] */
/* 682 */ NdrFcShort( 0xffa0 ), /* Offset= -96 (586) */
/* 684 */ 0x13, 0x10, /* FC_OP
[pointer_deref] */
/* 686 */ NdrFcShort( 0xffdc ), /* Offset= -516 (170) */
/* 688 */ 0x13, 0x10, /* FC_OP
[pointer_deref] */
/* 690 */ NdrFcShort( 0xfe30 ), /* Offset= -464 (226) */
/* 692 */ 0x13, 0x10, /* FC_OP
[pointer_deref] */
/* 694 */ NdrFcShort( 0x2 ), /* Offset= 2 (696) */
/* 696 */ 0x13, 0x10, /* FC_OP
[pointer_deref] */
/* 698 */ NdrFcShort( 0x2 ), /* Offset= 2 (700) */
/* 700 */ 0x13, 0x0, /* FC_OP */
/* 702 */ NdrFcShort( 0x14e ), /* Offset= 334 (1036) */
/* 704 */ 0x13, 0x10, /* FC_OP
[pointer_deref] */
/* 706 */ NdrFcShort( 0x2 ), /* Offset= 2 (708) */
/* 708 */ 0x13, 0x0, /* FC_OP */
/* 710 */ NdrFcShort( 0x14 ), /* Offset= 20 (730) */
/* 712 */ 0x15, /* FC_STRUCT */
/* 714 */ NdrFcShort( 0x10 ), /* 16 */
/* 716 */ 0x6, /* FC_SHORT */
/* 718 */ 0x1, /* FC_BYTE */
/* 720 */ 0xb, /* FC_HYPER */
/* 722 */ 0x5b, /* FC_END */
/* 724 */ NdrFcShort( 0xfff4 ), /* Offset= -12 (712) */
/* 726 */ 0x13, 0x8, /* FC_OP [simple_pointer] */
/* 728 */ 0x2, /* FC_CHAR */
/* 730 */ 0x5c, /* FC_PAD */
FC_BOGUS_STRUCT */
0x7, /* 7 */
/* 732 */ NdrFcShort( 0x20 ), /* 32 */
/* 734 */ NdrFcShort( 0x0 ), /* 0 */
/* 736 */ NdrFcShort( 0x0 ), /* Offset= 0 (736) */
/* 738 */ 0x8, /* FC_LONG */
/* 740 */ 0x6, /* FC_SHORT */
/* 742 */ 0x6, /* FC_SHORT */
/* 744 */ 0x4c, /* FC_EMBEDDED_COMPLEX */
/* 746 */ NdrFcShort( 0xfe30 ), /* Offset= -464 (282) */
/* 748 */ 0x5c, /* FC_PAD */
/* 750 */ 0x5b, /* FC_END */
/* 752 */ NdrFcShort( 0x4 ), /* 4 */
/* 754 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
/* 756 */ NdrFcShort( 0x0 ), /* 0 */
/* 758 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 760 */ 0x4b, /* FC_PP */
/* 762 */ 0x5c, /* FC_PAD */
FC_VARIABLE_REPEAT */
0x48, /*
FC_FIXED_OFFSET */
/* 764 */ NdrFcShort( 0x4 ), /* 4 */
/* 766 */ NdrFcShort( 0x0 ), /* 0 */
/* 768 */ NdrFcShort( 0x1 ), /* 1 */
/* 770 */ NdrFcShort( 0x0 ), /* 0 */
/* 772 */ NdrFcShort( 0x0 ), /* 0 */
/* 774 */ 0x13, 0x0, /* FC_OP */
/* 776 */ NdrFcShort( 0xffd2 ), /* Offset= -46 (730) */
/* 778 */

```

	0x5b,	/* FC_END */			0x3,	/* 3 */
/* 780 */	0x8,	/* FC_LONG */	/* 854 */	NdrFcShort(0x10),	/* 16 */	
	/* FC_PAD */		/* 856 */	0x8,	/* FC_LONG */	
/* 782 */	0x5b,	/* FC_END */	/* 858 */	0x6,	/* FC_SHORT */	/* FC_SHORT */
	0x1a,	/*		0x4c,	/*	
FC_BOGUS_STRUCT */			FC_EMBEDDED_COMPLEX */			
/* 784 */	0x3,	/* 3 */	/* 860 */	0x0,	/* 0 */	
/* 786 */	NdrFcShort(0x8),	/* 8 */		NdrFcShort(0xffff1),	/* Offset= -15 (846) */	
/* 788 */	NdrFcShort(0x0),	/* 0 */	/* 864 */	0x5b,	/* FC_END */	
/* 790 */	NdrFcShort(0x6),	/* Offset= 6 (794) */		0x1a,	/*	
/* 792 */	0x8,	/* FC_LONG */	FC_BOGUS_STRUCT */	0x3,	/* 3 */	
	0x36,	/* FC_POINTER */	/* 866 */	NdrFcShort(0x18),	/* 24 */	
/* 794 */	0x5b,	/* FC_END */	/* 868 */	NdrFcShort(0x0),	/* 0 */	
	0x11, 0x0,	/* FC_RP */	/* 870 */	NdrFcShort(0xa),	/* Offset= 10 (880) */	
/* 796 */	NdrFcShort(0xffd2),	/* Offset= -46 (750) */	/* 872 */	0x8,	/* FC_LONG */	
/* 798 */			/* 874 */	0x4c,	/* FC_POINTER */	/* FC_POINTER */
	0x1b,	/* FC_CARRAY */		0x0,	/* FC_EMBEDDED_COMPLEX */	
/* 800 */	NdrFcShort(0x4),	/* 4 */	/* 876 */	NdrFcShort(0xffe8),	/* Offset= -24 (852) */	
/* 802 */	0x19,	/* Corr desc: field pointer, FC_ULONG */	/* 878 */	0x5c,	/* FC_PAD */	
	0x0,	/* */	/* 880 */	0x5b,	/* FC_END */	
/* 804 */	NdrFcShort(0x0),	/* 0 */		0x11, 0x0,	/* FC_RP */	
/* 806 */	NdrFcShort(0x1),	/* Corr flags: early, */	/* 882 */	NdrFcShort(0xfd4a),	/* Offset= -694 (188) */	
/* 808 */			/* 884 */			
	0x4b,	/* FC_PP */		0x1b,	/* FC_CARRAY */	/* FC_CARRAY */
/* 810 */	0x5c,	/* FC_PAD */		0x0,	/* 0 */	/* 0 */
	0x48,	/*	/* 886 */	NdrFcShort(0x1),	/* 1 */	
FC_VARIABLE_REPEAT */			/* 888 */	0x19,	/* Corr desc: field pointer, FC_ULONG */	/* FC_ULONG */
	0x49,	/*	/* 890 */	NdrFcShort(0x0),	/* 0 */	
FC_FIXED_OFFSET */			/* 892 */	NdrFcShort(0x1),	/* Corr flags: early, */	
/* 812 */	NdrFcShort(0x4),	/* 4 */	/* 894 */	0x1,	/* FC_BYTE */	
/* 814 */	NdrFcShort(0x0),	/* 0 */	/* 896 */	0x5b,	/* FC_END */	/* FC_END */
/* 816 */	NdrFcShort(0x1),	/* 1 */		0x16,	/* FC_PSTRUCT */	/* FC_PSTRUCT */
/* 818 */	NdrFcShort(0x0),	/* 0 */	/* 898 */	NdrFcShort(0x8),	/* 8 */	/* 3 */
/* 820 */	NdrFcShort(0x0),	/* 0 */	/* 900 */			
/* 822 */	0x13, 0x0,	/* FC_OP */	/* 902 */	0x4b,	/* FC_PP */	/* FC_PP */
/* 824 */	NdrFcShort(0xff40),	/* Offset= -192 (632) */		0x5c,	/* FC_PAD */	/* FC_PAD */
/* 826 */			/* 904 */	NdrFcShort(0x4),	/* 4 */	
	0x5b,	/* FC_END */	/* 906 */	NdrFcShort(0x4),	/* 4 */	
/* 828 */	0x8,	/* FC_LONG */	/* 908 */	0x13, 0x0,	/* FC_OP */	
	/* FC_PAD */		/* 910 */	NdrFcShort(0xffe6),	/* Offset= -26 (884) */	
/* 830 */	0x5b,	/* FC_END */	/* 912 */			
	0x1a,	/*		0x5b,	/* FC_END */	/* FC_END */
FC_BOGUS_STRUCT */			/* 914 */	0x8,	/* FC_LONG */	/* FC_LONG */
/* 832 */	NdrFcShort(0x8),	/* 8 */	/* 916 */	0x5b,	/* FC_END */	/* FC_END */
/* 834 */	NdrFcShort(0x0),	/* 0 */		0x1b,	/* FC_CARRAY */	/* FC_CARRAY */
/* 836 */	NdrFcShort(0x6),	/* Offset= 6 (842) */	/* 918 */	0x1,	/* 1 */	
/* 838 */	0x8,	/* FC_LONG */	/* 920 */	0x19,	/* Corr desc: field pointer, FC_ULONG */	/* FC_ULONG */
/* 840 */	0x5c,	/* FC_POINTER */	/* 922 */	NdrFcShort(0x0),	/* 0 */	
	0x5b,	/* FC_END */	/* 924 */	NdrFcShort(0x1),	/* Corr flags: early, */	
/* 842 */			/* 926 */	0x6,	/* FC_SHORT */	/* FC_SHORT */
	0x11, 0x0,	/* FC_RP */	/* 928 */	0x5b,	/* FC_END */	/* FC_END */
/* 844 */	NdrFcShort(0xffd2),	/* Offset= -46 (798) */		0x16,	/* FC_PSTRUCT */	/* FC_PSTRUCT */
/* 846 */						
	0x1d,	/* FC_SMFARRAY */				
/* 848 */	NdrFcShort(0x8),	/* 8 */				
/* 850 */	0x1,	/* FC_BYTE */				
/* 852 */	0x5b,	/* FC_END */				
	0x15,	/* FC_STRUCT */				

/* 930 */ NdrFcShort(0x8),	0x3,	/* 3 */	/* 1004 */ 0x13, 0x0, /* FC_OP */
/* 932 */	/* 8 */		/* 1006 */ NdrFcShort(0xffe6), /* Offset= -26 (980) */
	0x4b,	/* FC_PP */	/* 1008 */
/* 934 */	0x5c,	/* FC_PAD */	0x5b,
			/* FC_END */
	0x46,	/* FC_NO_REPEAT */	0x8,
/* 936 */ NdrFcShort(0x4),	0x5c,	/* FC_PAD */	/* 1010 */ 0x8,
/* 938 */ NdrFcShort(0x4),	/* 4 */		0x5b,
/* 940 */ 0x13, 0x0, /* FC_OP */	/* 4 */		/* FC_END */
/* 942 */ NdrFcShort(0xffe6), /* Offset= -26 (916) */			0x15,
/* 944 */			/* FC_STRUCT */
	0x5b,	/* FC_END */	0x3,
/* 946 */ 0x8,	0x8,	/* FC_LONG */	/* 3 */
/* 948 */	0x5b,	/* FC_END */	/* 1014 */ NdrFcShort(0x8),
			/* 8 */
/* 950 */ NdrFcShort(0x4),	0x1b,	/* FC_CARRAY */	/* 1016 */ 0x8,
/* 952 */ 0x19,	0x3,	/* 3 */	/* FC_LONG */
	/* Corr desc: field pointer, FC_ULONG */		0x8,
/* 954 */ NdrFcShort(0x0),	0x0,	/* FC_END */	/* FC_LONG */
/* 956 */ NdrFcShort(0x1),	/* 0 */		0x5b,
/* 958 */ 0x8,	/* Corr flags: early, */		/* FC_END */
/* 960 */	0x5b,	/* FC_END */	0x1b,
			/* FC_CARRAY */
/* 962 */ NdrFcShort(0x8),	0x16,	/* FC_PSTRUCT */	0x3,
/* 964 */	0x3,	/* 3 */	/* 3 */
	0x4b,	/* FC_PP */	FC_BOGUS_STRUCT */
/* 966 */	0x5c,	/* FC_PAD */	0x3,
	0x46,	/* FC_NO_REPEAT */	/* 3 */
/* 968 */ NdrFcShort(0x4),	0x5c,	/* FC_PAD */	/* 1038 */ NdrFcShort(0x28),
/* 970 */ NdrFcShort(0x4),	/* 4 */		/* 40 */
/* 972 */ 0x13, 0x0, /* FC_OP */	/* 4 */		/* 1040 */ NdrFcShort(0xffec), /* Offset= -20 (1020) */
/* 974 */ NdrFcShort(0xffe6), /* Offset= -26 (948) */			/* 1042 */ NdrFcShort(0x0),
/* 976 */			/* Offset= 0 (1042) */
	0x5b,	/* FC_END */	/* 1044 */ 0x6,
/* 978 */ 0x8,	0x8,	/* FC_LONG */	0x6,
/* 980 */	0x8,	/* FC_LONG */	/* FC_SHORT */
	0x5b,	/* FC_END */	/* FC_LONG */
/* 982 */ NdrFcShort(0x8),	0x1b,	/* FC_CARRAY */	/* FC_EMBEDDED_COMPLEX */
/* 984 */ 0x19,	0x7,	/* 7 */	0x0,
	/* 8 */		/* 0 */
/* 986 */ NdrFcShort(0x0),	/* Corr desc: field pointer, FC_ULONG */		/* 1050 */ NdrFcShort(0xfc18), /* Offset= -1000 (50) */
/* 988 */ NdrFcShort(0x1),	0x0,	/* FC_END */	/* 1052 */ 0x5c,
/* 990 */ 0xb,	/* FC_HYPER */		/* FC_PAD */
/* 992 */	0x5b,	/* FC_END */	0x5b,
			/* FC_USER_MARSHAL */
/* 994 */ NdrFcShort(0x8),	0x16,	/* FC_PSTRUCT */	0x83,
/* 996 */	0x3,	/* 3 */	/* 131 */
	0x4b,	/* FC_PP */	/* 1056 */ NdrFcShort(0x1),
/* 998 */	0x5c,	/* FC_PAD */	/* 1058 */ NdrFcShort(0x4),
	0x46,	/* FC_NO_REPEAT */	/* 4 */
/* 1000 */ NdrFcShort(0x4),	0x5c,	/* FC_PAD */	/* 1060 */ NdrFcShort(0x0),
/* 1002 */ NdrFcShort(0x4),	/* 4 */		/* 0 */
			/* 1062 */ NdrFcShort(0xfc04), /* Offset= -1020 (42) */
			/* 1064 */
			0x11, 0x8, /* FC_RP [simple_pointer] */
			/* FC_LONG */
			/* 1066 */ 0x8,
			/* FC_PAD */
			/* 1068 */
			0x11, 0x14,
			/* FC_RP
			[allocated_on_stack] [pointer_deref] */
			/* 1070 */ NdrFcShort(0x2),
			/* Offset= 2 (1072) */
			/* 1072 */
			0x13, 0x0, /* FC_OP */
			/* 1074 */ NdrFcShort(0x2),
			/* Offset= 2 (1076) */
			/* 1076 */
			0x1b,
			/* FC_CARRAY */
			0x0,
			/* 0 */
			/* 1078 */ NdrFcShort(0x1),
			/* 1 */
			/* 1080 */ 0x28,
			/* Corr desc: parameter, FC_LONG */
			0x54,
			/* FC_DEREFERENCE */

```

/* 1082 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
/* 1084 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 1086 */ 0x2, /* FC_CHAR */
                0x5b, /* FC_END */
                0x0
        }
};
static const USER_MARSHAL_ROUTINE_QUADRUPLE
UserMarshalRoutines[ WIRE_MARSHAL_TABLE_SIZE ] =
{
    {
        BSTR_UserSize
        ,BSTR_UserMarshal
        ,BSTR_UserUnmarshal
        ,BSTR_UserFree
    },
    {
        LPSAFEARRAY_UserSize
        ,LPSAFEARRAY_UserMarshal
        ,LPSAFEARRAY_UserUnmarshal
        ,LPSAFEARRAY_UserFree
    }
};

/* Object interface: IUnknown, ver. 0.0,

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

/* Object interface: IDispatch, ver. 0.0,

GUID={0x00020400,0x0000,0x0000,{0xC0,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
0x46}} */

/* Object interface: IComponentRegistrar, ver. 0.0,

GUID={0xa817e7a2,0x43fa,0x11d0,{0x9e,0x44,0x00,0xaa,0x00,0xb6,0x77,0x0a}} */
#pragma code_seg(".orpc")
static const unsigned short IComponentRegistrar_FormatStringOffsetTable[] =
{
    (unsigned short) -1,
    (unsigned short) -1,
    (unsigned short) -1,
    (unsigned short) -1,
    0,
    36,
    66,
    96,
    138,
    174
};
static const MIDL_STUBLESS_PROXY_INFO
IComponentRegistrar_ProxyInfo =
{
    &Object_StubDesc,
    __MIDL_ProcFormatString.Format,
    &IComponentRegistrar_FormatStringOffsetTable[-3],
    0,
    0,
    0
};

static const MIDL_SERVER_INFO IComponentRegistrar_ServerInfo =
{
    &Object_StubDesc,
    0,

```

```

    __MIDL_ProcFormatString.Format,
    &IComponentRegistrar_FormatStringOffsetTable[-3],
    0,
    0,
    0,
    0;
CINTERFACE_PROXY_VTABLE(13) _IComponentRegistrarProxyVtbl =
{
    &IComponentRegistrar_ProxyInfo,
    &IID_IComponentRegistrar,
    IUnknown_QueryInterface_Proxy,
    IUnknown_AddRef_Proxy,
    IUnknown_Release_Proxy,
    0 /* (void *) (INT_PTR) -1 /* IDispatch::GetTypeInfoCount */ ,
    0 /* (void *) (INT_PTR) -1 /* IDispatch::TypeInfo */ ,
    0 /* (void *) (INT_PTR) -1 /* IDispatch::GetIDsOfNames */ ,
    0 /* IDispatch_Invoke_Proxy */ ,
    (void *) (INT_PTR) -1 /* IComponentRegistrar::Attach */ ,
    (void *) (INT_PTR) -1 /* IComponentRegistrar::RegisterAll */ ,
    (void *) (INT_PTR) -1 /* IComponentRegistrar::UnregisterAll */ ,
    (void *) (INT_PTR) -1 /* IComponentRegistrar::GetComponents */ ,
    (void *) (INT_PTR) -1 /* IComponentRegistrar::RegisterComponent */ ,
    (void *) (INT_PTR) -1 /* IComponentRegistrar::UnregisterComponent */
};

static const PRPC_STUB_FUNCTION IComponentRegistrar_table[] =
{
    STUB_FORWARDING_FUNCTION,
    STUB_FORWARDING_FUNCTION,
    STUB_FORWARDING_FUNCTION,
    STUB_FORWARDING_FUNCTION,
    NdrStubCall2,
    NdrStubCall2,
    NdrStubCall2,
    NdrStubCall2,
    NdrStubCall2,
    NdrStubCall2,
    NdrStubCall2
};
CInterfaceStubVtbl _IComponentRegistrarStubVtbl =
{
    &IID_IComponentRegistrar,
    &IComponentRegistrar_ServerInfo,
    13,
    &IComponentRegistrar_table[-3],
    CStdStubBuffer_DELEGATING_METHODS
};

/* Object interface: Itpcc_com, ver. 0.0,

GUID={0x5B4FA473,0x2E68,0x4D79,{0xA6,0x26,0xF3,0x8B,0x30,0xB8,0x19,0x6E}} */
#pragma code_seg(".orpc")
static const unsigned short Itpcc_com_FormatStringOffsetTable[] =
{
    210,
    252,
    294,
    336,
    378,
    36
};
static const MIDL_STUBLESS_PROXY_INFO Itpcc_com_ProxyInfo =
{
    &Object_StubDesc,
    __MIDL_ProcFormatString.Format,
    &Itpcc_com_FormatStringOffsetTable[-3],
    0,
    0,
    0,

```

```

0
};

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

CINTERFACE_PROXY_VTABLE(9) _Itpcc_comProxyVtbl =
{
    &Itpcc_com_ProxyInfo,
    &IID_Itpcc_com,
    IUnknown_QueryInterface_Proxy,
    IUnknown_AddRef_Proxy,

    IUnknown_Release_Proxy,
    (void *) (INT_PTR) -1 /* Itpcc_com::doStockLevel */,
    (void *) (INT_PTR) -1 /* Itpcc_com::doNewOrder */,
    (void *) (INT_PTR) -1 /* Itpcc_com::doPayment */,
    (void *) (INT_PTR) -1 /* Itpcc_com::doOrderStatus */,
    (void *) (INT_PTR) -1 /* Itpcc_com::doDBInfo */,
    (void *) (INT_PTR) -1 /* Itpcc_com::doSetComplete */
};

const CInterfaceStubVtbl _Itpcc_comStubVtbl =
{
    &IID_Itpcc_com,
    &Itpcc_com_ServerInfo,
    9,
    0, /* pure interpreted */
    CStdStubBuffer_METHODS
};

static const MIDL_STUB_DESC Object_StubDesc =
{
    0,
    NdrOleAllocate,
    NdrOleFree,
    0,
    0,
    0,
    0,
    0,
    0,
    __MIDL_TypeFormatString.Format,
    1, /* -error bounds_check flag */
    0x50002, /* Ndr library version */
    0,
    0x6000169, /* MIDL Version 6.0.361 */
    0,
    UserMarshalRoutines,
    0, /* notify & notify_flag routine table */
    0x1, /* MIDL flag */
    0, /* cs routines */
    0, /* proxy/server info */
    0 /* Reserved5 */
};

const CInterfaceProxyVtbl * _tpccCom_ProxyVtblList[] =
{
    ( CInterfaceProxyVtbl *) &_Itpcc_comProxyVtbl,
    ( CInterfaceProxyVtbl *) &_IComponentRegistrarProxyVtbl,
    0
};

const CInterfaceStubVtbl * _tpccCom_StubVtblList[] =
{
    ( CInterfaceStubVtbl *) &_Itpcc_comStubVtbl,

```

```

    ( CInterfaceStubVtbl *) &_IComponentRegistrarStubVtbl,
    0
};

PCInterfaceName const _tpccCom_InterfaceNamesList[] =
{
    "Itpcc_com",
    "IComponentRegistrar",
    0
};

const IID * _tpccCom_BaseIIDList[] =
{
    0,
    &IID_IDispatch,
    0
};

#define _tpccCom_CHECK_IID(n)      IID_GENERIC_CHECK_IID(
    _tpccCom, pIID, n)
int __stdcall _tpccCom_IID_Lookup( const IID * pIID, int * pIndex )
{
    IID_BS_LOOKUP_SETUP
    IID_BS_LOOKUP_INITIAL_TEST( _tpccCom, 2, 1 )
    IID_BS_LOOKUP_RETURN_RESULT( _tpccCom, 2, *pIndex )
}

const ExtendedProxyFileInfo tpccCom_ProxyFileInfo =
{
    (PCInterfaceProxyVtblList *) &_tpccCom_ProxyVtblList,

    (PCInterfaceStubVtblList *) &_tpccCom_StubVtblList,
    (const PCInterfaceName *) &_tpccCom_InterfaceNamesList,
    (const IID **) &_tpccCom_BaseIIDList,
    &_tpccCom_IID_Lookup,
    2,
    2,
    0, /* table of [async_uuid] interfaces */
    0, /* Filler1 */
    0, /* Filler2 */
    0 /* Filler3 */
};

#ifdef _MSC_VER >= 1200
#pragma warning(pop)
#endif

#endif /* !defined(_M_IA64) && !defined(_M_AMD64) */

```

TpccDB2Glue/stdafx.h

```

// stdafx.h : include file for standard system include files,
// or project specific include files that are used frequently, but
// are changed infrequently
//
#pragma once

#define WIN32_LEAN_AND_MEAN // Exclude rarely-used
stuff from Windows headers
// Windows Header Files:
#include <windows.h>
// TODO: reference additional headers your program requires here

tpccDB2Glue/tpccDB2glue.h

// The following ifdef block is the standard way of creating macros which make
exporting
// from a DLL simpler. All files within this DLL are compiled with the
TPCCDB2GLUE_EXPORTS

```

```

// symbol defined on the command line. this symbol should not be defined on
// any project
// that uses this DLL. This way any other project whose source files include this
// file see
// TPCCDB2GLUE_API functions as being imported from a DLL, whereas this
// DLL sees symbols
// defined with this macro as being exported.
#ifdef TPCCDB2GLUE_EXPORTS
#define TPCCDB2GLUE_API __declspec(dllexport)
#else
#define TPCCDB2GLUE_API __declspec(dllimport)
#endif
#ifndef SPGENERAL
#define SPGENERAL
#endif
#include <db2tpcc.h>
#include <tpcc.h>
// Error/Debug log file defines
ofstream debugStream;
ofstream errorStream;
CRITICAL_SECTION debugMutex;
CRITICAL_SECTION errorMutex;
// #define TIMING 1
FILE *respTimes;
struct txn
{
    short txnType;
    struct _timeb startTime;
    struct _timeb endTime;
    short padding;
};
// Registry Values
#define DB_USER_NAME
"dbUserName"
#define DB_USER_PASSWORD
"dbPassword"
#define DB_NAME
"dbName"
char userName[16] = {NULL};
char userPassword[16] = {NULL};
HKEY registryKey;
DWORD regType;
char value[MAX_STRING_LEN];
DWORD regValueSize = MAX_STRING_LEN;
// DB2 Glue Function Prototypes
extern "C" TPCCDB2GLUE_API int connect_db(char *dbName, void **ctx);
extern "C" TPCCDB2GLUE_API int getContext(void **ctx);
extern "C" TPCCDB2GLUE_API int detachContext(void *ctx);
extern "C" TPCCDB2GLUE_API int attachContext(void *ctx);
extern "C" TPCCDB2GLUE_API int disconnect_db(void *ctx);
extern "C" TPCCDB2GLUE_API int do_nord(nord_wrapper *nord, void *ctx);
extern "C" TPCCDB2GLUE_API int do_pymt(paym_wrapper *pymt, void
*ctx);
extern "C" TPCCDB2GLUE_API int do_ords(ords_wrapper *ords, void *ctx);
extern "C" TPCCDB2GLUE_API int do_dlv(dlv_wrapper *dlvy, void *ctx);
extern "C" TPCCDB2GLUE_API int do_stok(stok_wrapper *stok, void *ctx);

```

tpccDB2Glue/stdafx.cpp

```

// stdafx.cpp : source file that includes just the standard includes
// tpccDB2glue.pch will be the pre-compiled header
// stdafx.obj will contain the pre-compiled type information

```

```

#include "stdafx.h"
// TODO: reference any additional headers you need in STDAFX.H
// and not in this file

```

tpccDB2Glue/tpccDB2glue.cpp

```

// tpccDB2glue.cpp : Defines the entry point for the DLL application.
//
#include "stdafx.h"
#include "tpccDB2glue.h"
BOOL APIENTRY DllMain( HANDLE hModule,
    DWORD ul_reason_for_call,
    LPVOID lpReserved
    )
{
    switch (ul_reason_for_call)
    {
        case DLL_PROCESS_ATTACH:
            if (debugFlag)
            {
                InitializeCriticalSection(&debugMutex);

                debugStream.rdbuf(
)->open("C:\\inetpub\\wwwroot\\tpcc\\debug_gluecode.txt", ios_base::in |
ios_base::out | ios_base::app);
                if (!debugStream.rdbuf() ->is_open())
                    return FALSE;
            }

            DEBUGMSG("Entered dllMain of tpccDB2glue.dll" <<
endl);

            InitializeCriticalSection(&errorMutex);
            errorStream.rdbuf(
)->open("C:\\inetpub\\wwwroot\\tpcc\\error_gluecode.txt", ios_base::in |
ios_base::out | ios_base::app);
            if (!errorStream.rdbuf() ->is_open())
                return FALSE;

#ifdef TIMING
respTimes=fopen("c:\\inetpub\\wwwroot\\tpcc\\respTimes", "wb");
if (!respTimes)
{
    ERRORMSG("Unable to open response time
file c:\\inetpub\\wwwroot\\tpcc\\respTimes" <<endl);
    return FALSE;
}
ERRORMSG("Response time file created:" <<endl);
#endif

            DEBUGMSG("Opening registry sub key "<<
REGISTRY_SUB_KEY << endl);
            //open up registry key

if (RegOpenKeyEx(HKEY_LOCAL_MACHINE, REGISTRY_SUB_KEY, 0, K
EY_READ, &registryKey) == ERROR_SUCCESS)
{
            DEBUGMSG("Registry key open" <<endl);
            //get the null db user name
            regValueSize = sizeof(value);
            if
(RegQueryValueEx(registryKey, DB_USER_NAME, 0, &regType, (BYTE *)
&value, &regValueSize) == ERROR_SUCCESS )
                strcpy(userName, value);
            else
                return
ERR_INVALID_USERNAME;

```

```

                DEBUGMSG("DB user name:"<< userName
<< endl);
                regValueSize = sizeof(value);
                if
                    (RegQueryValueEx(registryKey,DB_USER_PASSWORD,0,&regType,(BYTE
                    E *) &value,&regValueSize)== ERROR_SUCCESS )
                        strcpy(userPassword,value);
                    else
                        return
ERR_INVALID_PASSWORD;
                DEBUGMSG("DB user
password:"<<userPassword << endl);
                }
                else
                {
                    return ERR_INVALID_REGISTRY_KEY;
                    DEBUGMSG("Unable to open registry
key"<< REGISTRY_SUB_KEY << endl);
                }
                break;
            case DLL_THREAD_ATTACH:
                break;
            case DLL_THREAD_DETACH:
                break;
            case DLL_PROCESS_DETACH:
                #ifdef TIMING
                    ERRORMSG("dll_process_detach called,
closing timing file"<<endl);
                    fclose(respTimes);
                #endif
                break;
        }
        return TRUE;
    }
}
/*
*****
** Name          :          attachContext
** Description   :
**              Function calls db2 api
to attach thread to
**              a specific context per
thread basis.
** Parameters   :
**              void*
stored context
** Returns      :
**              int - return code
** Comments     :
**
*****
*/
extern "C" int attachContext(void *ctx)
{
    int rc;
    if ( (rc = attach_context(ctx)) != OK)
        return ERR_ATTACHING_CONTEXT;

    return OK;
}
/*
*****
** Name          :          detachContext
** Description   :
**              Function calls db2 api
to detach thread from context
** Parameters   :

```

```

        DEBUGMSG("Object calling detach_context() w/
ctx:"<<DEBUGADDRESS(*ctx)<<endl);
        if( rc = detach_context(*ctx) != OK)
        {
            DEBUGMSG("Object failed detach_context w/
ctx:"<<DEBUGADDRESS(*ctx)<<" rc:" << rc << endl);
            ERRORMSG("Object failed detach_context w/
ctx:"<<DEBUGADDRESS(*ctx)<<" rc:" << rc << endl);

            return ERR_DETACHING_CONTEXT;
        }
        DEBUGMSG("Object detach_context successful,
context:"<<DEBUGADDRESS(*ctx)<<" , connection complete"<<endl);
        return OK;
    }
/*
*****
** Name          :          disconnect_db
** Description   :
**              :          Function calls db2 api
to disconnect from db
** Parameters    :
**              :          void*
stored context
** Returns       :
**              :          int - return code
** Comments     :
**              :          To disconnect from db,
first must attach to
**              :          thread's context. Next,
disconnect from db
*****
*/
extern "C" TPCCDB2GLUE_API int disconnect_db(void *ctx)
{
    DEBUGMSG("Entered do_disconnect, attaching to context:" <<
DEBUGADDRESS(ctx) << endl);
    int rc = attachContext(ctx);
    if(rc != OK)
    {
        ERRORMSG("failed attach_context w/
ctx:"<<DEBUGADDRESS(ctx)<<" rc:" << rc << endl);
        DEBUGMSG("failed attach_context w/
ctx:"<<DEBUGADDRESS(ctx)<<" rc:" << rc << endl);
        return ERR_ATTACHING_CONTEXT;
    }
    DEBUGMSG("context established. preparing to call db2" << endl);
    rc = disconnect_from_TM();
    if(rc != OK)
    {
        DEBUGMSG("disconnect failed, rc:"<<rc<<endl);
        ERRORMSG("disconnect failed, rc:"<<rc<<endl);
        return rc;
    }
    return OK;
}
/*
*****
** Name          :          do_nord
** Description   :
**              :          Function calls db2 api
to execute nord txn
** Parameters    :
**              :          nord_wrapper*
new order txn structs wrapper
**              :          void*
stored context

```

```

** Returns      :
**              :          int - return code
** Comments     :
**              :          Attach to thread's
context, call nord sql function
**              :          then detach from
context.
*****
*/
extern "C" TPCCDB2GLUE_API int do_nord(nord_wrapper *nord,void *ctx)
{
    DEBUGMSG("Entered do_nord, attaching to context:" <<
DEBUGADDRESS(ctx) << endl);
    int rc = attachContext(ctx);
    if(rc != OK)
    {
        ERRORMSG("nord failed attach_context w/
ctx:"<<DEBUGADDRESS(ctx)<<" rc:" << rc << endl);
        DEBUGMSG("nord failed attach_context w/
ctx:"<<DEBUGADDRESS(ctx)<<" rc:" << rc << endl);
        return ERR_ATTACHING_CONTEXT;
    }
    DEBUGMSG("attached to context:" <<
DEBUGADDRESS(ctx)<<" , preparing to call db2" << endl);
#ifdef TIMING
    struct txn timeSample;
    _ftime(&timeSample.startTime);
#endif
    //call new order txn
    neword_sql(&nord->in_nord,&nord->out_nord);
#ifdef TIMING
    _ftime(&timeSample.endTime);
    timeSample.txnType=1;
    EnterCriticalSection(&errorMutex);
    rc = fwrite(&timeSample,sizeof(struct txn),1,respTimes);
    LeaveCriticalSection(&errorMutex);
#endif
    DEBUGMSG("return from neword_sql(), s_transtatus:" <<
nord->out_nord.s_transtatus << endl);
    rc = detachContext(ctx);
    if(rc != OK)
    {
        ERRORMSG("nord failed detach_context w/
ctx:"<<DEBUGADDRESS(ctx)<<" rc:" << rc << endl);
        DEBUGMSG("nord failed detach_context w/
ctx:"<<DEBUGADDRESS(ctx)<<" rc:" << rc << endl);
        return ERR_DETACHING_CONTEXT;
    }
    return OK;
}
/*
*****
** Name          :          do_pymt
** Description   :
**              :          Function calls db2 api
to execute pymt txn
** Parameters    :
**              :          paym_wrapper*
**              :          void*
stored context
** Returns       :
**              :          int - return code
** Comments     :
**              :          Attach to thread's
context, call nord sql function

```

```

**                                     then detach from
context.

*****

*/
extern "C" TPCCDB2GLUE_API int do_pymt(payment_wrapper *pymt,void *ctx)
{
    DEBUGMSG("Entered do_pymt, attaching to context:" <<
DEBUGADDRESS(ctx) << endl);
    int rc = attachContext(ctx);
    if(rc != OK)
    {
        ERRORMSG("pymt failed attach_context w/
ctx:"<<DEBUGADDRESS(ctx)<<" rc:" << rc << endl);
        DEBUGMSG("pymt failed attach_context w/
ctx:"<<DEBUGADDRESS(ctx)<<" rc:" << rc << endl);
        return ERR_ATTACHING_CONTEXT;
    }
    DEBUGMSG("attached to context:"<< DEBUGADDRESS(ctx) <<"
preparing to call db2" << endl);
#ifdef TIMING
    struct txn timeSample;
    _ftime(&timeSample.startTime);
#endif
    //call pymt txn
    payment_sql(&pymt->in_paym,&pymt->out_paym);
#ifdef TIMING
    _ftime(&timeSample.endTime);
    timeSample.txnType=2;
    EnterCriticalSection(&errorMutex);
    if( (fwrite(&timeSample,sizeof(struct txn),1,respTimes)) != 1 )
    {
        ERRORMSG("Unable to write to binary file,
pymt"<<endl);
    }
    LeaveCriticalSection(&errorMutex);
#endif
    DEBUGMSG("return from payment_sql(), s_transtatus:" <<
pymt->out_paym.s_transtatus << endl);

    rc = detachContext(ctx);
    if(rc != OK)
    {
        ERRORMSG("pymt failed detach_context w/
ctx:"<<DEBUGADDRESS(ctx)<<endl);
        DEBUGMSG("pymt failed detach_context w/
ctx:"<<DEBUGADDRESS(ctx)<<" rc:" << rc << endl);

        return ERR_DETACHING_CONTEXT;
    }
    DEBUGMSG("pymt detach_context successful. pymt txn
complete."<<endl);
    return OK;
}
/*
*****
** Name           :           do_ords
** Description    :
**               :           Function calls db2 api
to execute ords txn
** Parameters    :
**               :           ords_wrapper*
order status txn structs wrapper
**               :           void*
stored context

```

```

** Returns       :
**               :           int - return code
** Comments     :
**               :           Attach to thread's
context, call nord sql function
**               :           then detach from
context.
*****
*/
extern "C" TPCCDB2GLUE_API int do_ords(ords_wrapper *ords,void *ctx)
{
    DEBUGMSG("Entered do_ords, attaching to context:" <<
DEBUGADDRESS(ctx) << endl);
    int rc = attachContext(ctx);
    if(rc != OK)
    {
        ERRORMSG("ords failed attach_context w/
ctx:"<<DEBUGADDRESS(ctx)<<" rc:" << rc << endl);
        DEBUGMSG("ords failed attach_context w/
ctx:"<<DEBUGADDRESS(ctx)<<" rc:" << rc << endl);
        return ERR_ATTACHING_CONTEXT;
    }
    DEBUGMSG("attached to context:"<<DEBUGADDRESS(ctx)<<"
preparing to call db2" << endl);
    DEBUGMSG("calling ordstat_sql()" <<endl);
#ifdef TIMING
    struct txn timeSample;
    _ftime(&timeSample.startTime);
#endif
    ordstat_sql(&ords->in_ords,&ords->out_ords);
#ifdef TIMING
    _ftime(&timeSample.endTime);
    timeSample.txnType=3;
    EnterCriticalSection(&errorMutex);
    if( (fwrite(&timeSample,sizeof(struct txn),1,respTimes)) != 1 )
    {
        ERRORMSG("Unable to write to binary file,
ords"<<endl);
    }
    LeaveCriticalSection(&errorMutex);
#endif
    DEBUGMSG("return from ordstat_sql(), s_transtatus:" <<
ords->out_ords.s_transtatus << endl);
    rc = detachContext(ctx);
    if(rc != OK)
    {
        ERRORMSG("ords failed attach_context w/
ctx:"<<DEBUGADDRESS(ctx)<<" rc:" << rc << endl);
        DEBUGMSG("ords failed attach_context w/
ctx:"<<DEBUGADDRESS(ctx)<<" rc:" << rc << endl);
        return ERR_DETACHING_CONTEXT;
    }
    DEBUGMSG("ords detach_context successful. pymt txn
complete."<<endl);
    return OK;
}
/*
*****
** Name           :           do_dlv
** Description    :
**               :           Function calls db2 api
to execute ords txn
** Parameters    :
**               :           dlv_wrapper*
dlv txn structs wrapper
**               :           void*
stored context
** Returns       :

```

```

**                                     int - return code
** Comments                           :
**                                     Attach to thread's
context, call nord sql function
**                                     then detach from
context.
*****
extern "C" TPCCDB2GLUE_API int do_dlvvy(dlvvy_wrapper *dlvy,void *ctx)
{
    DEBUGMSG("Entered do_dlvvy, attaching to context:" <<
DEBUGADDRESS(ctx) << endl);
    int rc = attachContext(ctx);
    if(rc != OK)
    {
        ERRORMSG("dlvy failed attach_context w/
ctx:"<<DEBUGADDRESS(ctx)<<" rc:" << rc << endl);
        DEBUGMSG("dlvy failed attach_context w/
ctx:"<<DEBUGADDRESS(ctx)<<" rc:" << rc << endl);
        return ERR_ATTACHING_CONTEXT;
    }
    DEBUGMSG("attached to context:"<<DEBUGADDRESS(ctx)<<"
preparing to call db2" << endl);
    DEBUGMSG("calling delivery_sql" << endl);
#ifdef TIMING
    struct txn timeSample;
    _ftime(&timeSample.startTime);
#endif
    //call dlvvy txn
    delivery_sql(&dlvy->in_dlvvy,&dlvy->out_dlvvy);
#ifdef TIMING
    _ftime(&timeSample.endTime);
    timeSample.txnType=3;
    EnterCriticalSection(&errorMutex);
    if( (fwrite(&timeSample,sizeof(struct txn),1,respTimes)) != 1 )
    {
        ERRORMSG("Unable to write to binary file,
dlvy"<<endl);
    }
    LeaveCriticalSection(&errorMutex);
#endif
    DEBUGMSG("return from delivery_sql(), s_transtatus:" <<
dlvy->out_dlvvy.s_transtatus << endl);
    rc = detachContext(ctx);
    if(rc != OK)
    {
        ERRORMSG("dlvy failed detach_context w/
ctx:"<<DEBUGADDRESS(ctx)<<" rc:" << rc << endl);
        DEBUGMSG("dlvy failed detach_context w/
ctx:"<<DEBUGADDRESS(ctx)<<" rc:" << rc << endl);
        return ERR_DETACHING_CONTEXT;
    }
    DEBUGMSG("dlvy detach_context successful. dlvvy txn
complete."<<endl);
    return OK;
}
/*
*****
** Name                               : do_stok
** Description                         :
**                                     Function calls db2 api
to execute stok txn
** Parameters                          :
**                                     stok_wrapper*
stock txn structs wrapper
**                                     void*
stored context
** Returns                             :

```

```

**                                     int - return code
** Comments                           :
**                                     Attach to thread's
context, call nord sql function
**                                     then detach from
context.
*****
extern "C" TPCCDB2GLUE_API int do_stok(stok_wrapper *stok,void *ctx)
{
    DEBUGMSG("Entered do_stok, attaching to context:" <<
DEBUGADDRESS(ctx) << endl);
    int rc = attachContext(ctx);
    if(rc != OK)
    {
        ERRORMSG("stok failed attach_context w/
ctx:"<<DEBUGADDRESS(ctx)<<" rc:" << rc << endl);
        DEBUGMSG("stok failed attach_context w/
ctx:"<<DEBUGADDRESS(ctx)<<" rc:" << rc << endl);
        return ERR_ATTACHING_CONTEXT;
    }
    DEBUGMSG("attaching to context:"<<DEBUGADDRESS(ctx)<<"
preparing to call db2" << endl);
    DEBUGMSG("calling stocklev_sql()" <<endl);
#ifdef TIMING
    struct txn timeSample;
    _ftime(&timeSample.startTime);
#endif
    //call stock level txn
    stocklev_sql(&stok->in_stok, &stok->out_stok);
#ifdef TIMING
    _ftime(&timeSample.endTime);
    timeSample.txnType=5;
    EnterCriticalSection(&errorMutex);
    if( (fwrite(&timeSample,sizeof(struct txn),1,respTimes)) != 1 )
    {
        ERRORMSG("Unable to write to binary file,
stok"<<endl);
    }
    LeaveCriticalSection(&errorMutex);
#endif
    DEBUGMSG("return from stocklev_sql(), s_transtatus:" <<
stok->out_stok.s_transtatus << endl);
    DEBUGMSG("calling detach_context"<<endl);
    rc = detachContext(ctx);
    if(rc != OK)
    {
        ERRORMSG("stok failed attach_context w/
ctx:"<<DEBUGADDRESS(ctx)<<" rc:" << rc << endl);
        DEBUGMSG("stok failed attach_context w/
ctx:"<<DEBUGADDRESS(ctx)<<" rc:" << rc << endl);
        return ERR_DETACHING_CONTEXT;
    }
    DEBUGMSG("detach_context successful. stok txn
complete."<<endl);
    return OK;
}
??
??
??
??
TPC Benchmark™ C Full Disclosure Report - IBM eServer p5 595 Model
9119-595 Page 121 of 704

```


NullDB.cpp

```
// nullDB.cpp : Defines the entry point for the DLL application.
//

#include "stdafx.h"
#include "nullDB.h"
#include "..\tpcc\api\tpcc.h"

BOOL APIENTRY DllMain( HANDLE hModule,
                      DWORD ul_reason_for_call,
                      LPVOID lpReserved
                      )
{
    switch (ul_reason_for_call)
    {
        case DLL_PROCESS_ATTACH:
        case DLL_THREAD_ATTACH:
        case DLL_THREAD_DETACH:
        case DLL_PROCESS_DETACH:
            break;
    }
    return TRUE;
}

// This is an example of an exported variable
NULLDB_API int dataSet = 0;

extern "C" NULLDB_API int connect_db(char *dbName, void **ctx)
{
    return OK;
}

extern "C" NULLDB_API int disconnect_db(void *ctx)
{
    return OK;
}

extern "C" NULLDB_API int do_nord(struct nord_wrapper *nord, void *ctx)
{
    nord->out_nord.s_transtatus = 0;

    if (dataSet == 0)
    {
        strcpy(nord->out_nord.s_C_LAST, "NOYOLA");
        strcpy(nord->out_nord.s_C_CREDIT, "GC");
        nord->out_nord.s_W_TAX = 1694;
        nord->out_nord.s_D_TAX = 967;
        nord->out_nord.s_C_DISCOUNT = 1024;
        nord->out_nord.s_O_ID = 3013;
        nord->out_nord.s_O_OL_CNT = 4;
        nord->out_nord.s_total_amount = 32345;
        nord->out_nord.s_O_ENTRY_D_time = 1234567890;

        strcpy(nord->out_nord.item[0].s_I_NAME, "98 Toyota
Supra Turbo");

        nord->in_nord.in_item[0].s_OL_I_ID = 1;
        nord->in_nord.in_item[0].s_OL_QUANTITY = 1;
        nord->in_nord.in_item[0].s_OL_SUPPLY_W_ID = 1;
        nord->out_nord.item[0].s_I_PRICE = 42000;
        nord->out_nord.item[0].s_OL_AMOUNT = 554000;
        nord->out_nord.item[0].s_S_QUANTITY = 31;
        nord->out_nord.item[0].s_brand_generic = 'G';

        strcpy(nord->out_nord.item[1].s_I_NAME, "HKS Turbo
Timer");
```

Exhaust");

DUAL-SOLENOID");

```
}
else
{
```

Supra NA");

Stereo");

Exhaust Header");

```
nord->in_nord.in_item[1].s_OL_I_ID = 1;
nord->in_nord.in_item[1].s_OL_QUANTITY = 1;
nord->in_nord.in_item[1].s_OL_SUPPLY_W_ID = 1;
nord->out_nord.item[1].s_I_PRICE = 4500;
nord->out_nord.item[1].s_OL_AMOUNT = 438100;
nord->out_nord.item[1].s_S_QUANTITY = 57;
nord->out_nord.item[1].s_brand_generic = 'G';
```

```
strcpy(nord->out_nord.item[2].s_I_NAME, "TRD GEN2
```

```
nord->in_nord.in_item[2].s_OL_I_ID = 1;
nord->in_nord.in_item[2].s_OL_QUANTITY = 1;
nord->in_nord.in_item[2].s_OL_SUPPLY_W_ID = 1;
nord->out_nord.item[2].s_I_PRICE = 6734;
nord->out_nord.item[2].s_OL_AMOUNT = 47173;
nord->out_nord.item[2].s_S_QUANTITY = 42;
nord->out_nord.item[2].s_brand_generic = 'G';
```

```
strcpy(nord->out_nord.item[3].s_I_NAME, "BLITZ
DUAL-SOLENOID");
```

```
nord->in_nord.in_item[3].s_OL_I_ID = 1;
nord->in_nord.in_item[3].s_OL_QUANTITY = 1;
nord->in_nord.in_item[3].s_OL_SUPPLY_W_ID = 1;
nord->out_nord.item[3].s_I_PRICE = 35000;
nord->out_nord.item[3].s_OL_AMOUNT = 12096;
nord->out_nord.item[3].s_S_QUANTITY = 84;
nord->out_nord.item[3].s_brand_generic = 'G';
```

```
dataSet = 1;
```

```
strcpy(nord->out_nord.s_C_LAST, "SIMPSON");
strcpy(nord->out_nord.s_C_CREDIT, "GC");
nord->out_nord.s_W_TAX = 913;
nord->out_nord.s_D_TAX = 1519;
nord->out_nord.s_C_DISCOUNT = 958;
nord->out_nord.s_O_ID = 1410;
nord->out_nord.s_O_OL_CNT = 9;
nord->out_nord.s_total_amount = 12345;
nord->out_nord.s_O_ENTRY_D_time = 1234567890;
```

```
strcpy(nord->out_nord.item[0].s_I_NAME, "97 Toyota
```

```
nord->in_nord.in_item[0].s_OL_I_ID = 1;
nord->in_nord.in_item[0].s_OL_QUANTITY = 1;
nord->in_nord.in_item[0].s_OL_SUPPLY_W_ID = 1;
nord->out_nord.item[0].s_I_PRICE = 30000;
nord->out_nord.item[0].s_OL_AMOUNT = 769600;
nord->out_nord.item[0].s_S_QUANTITY = 97;
nord->out_nord.item[0].s_brand_generic = 'G';
```

```
strcpy(nord->out_nord.item[1].s_I_NAME, "98 Turbo
```

```
nord->in_nord.in_item[1].s_OL_I_ID = 1;
nord->in_nord.in_item[1].s_OL_QUANTITY = 1;
nord->in_nord.in_item[1].s_OL_SUPPLY_W_ID = 1;
nord->out_nord.item[1].s_I_PRICE = 10001;
nord->out_nord.item[1].s_OL_AMOUNT = 192999;
nord->out_nord.item[1].s_S_QUANTITY = 51;
nord->out_nord.item[1].s_brand_generic = 'G';
```

```
strcpy(nord->out_nord.item[2].s_I_NAME, "XERD
```

```
nord->in_nord.in_item[2].s_OL_I_ID = 1;
nord->in_nord.in_item[2].s_OL_QUANTITY = 1;
nord->in_nord.in_item[2].s_OL_SUPPLY_W_ID = 1;
nord->out_nord.item[2].s_I_PRICE = 4000;
```

```

nord->out_nord.item[2].s_OL_AMOUNT = 41670;
nord->out_nord.item[2].s_S_QUANTITY = 14;
nord->out_nord.item[2].s_brand_generic = 'G';
}
extern "C" NULLDB_API int do_pymt(struct paym_wrapper *pymt,void *ctx)
{
    Conditioner");
    nord->out_nord.item[3].s_I_NAME,"LEXOL
    nord->in_nord.in_item[3].s_OL_I_ID = 1;
    nord->in_nord.in_item[3].s_OL_QUANTITY = 1;
    nord->in_nord.in_item[3].s_OL_SUPPLY_W_ID = 1;
    nord->out_nord.item[3].s_I_PRICE = 1400;
    nord->out_nord.item[3].s_OL_AMOUNT = 17213;
    nord->out_nord.item[3].s_S_QUANTITY = 90;
    nord->out_nord.item[3].s_brand_generic = 'G';
    1");
    nord->out_nord.item[4].s_I_NAME,"TRD Sticker
    nord->in_nord.in_item[4].s_OL_I_ID = 1;
    nord->in_nord.in_item[4].s_OL_QUANTITY = 1;
    nord->in_nord.in_item[4].s_OL_SUPPLY_W_ID = 1;
    nord->out_nord.item[4].s_I_PRICE = 1400;
    nord->out_nord.item[4].s_OL_AMOUNT = 27232;
    nord->out_nord.item[4].s_S_QUANTITY = 75;
    nord->out_nord.item[4].s_brand_generic = 'G';
    Horse");
    nord->out_nord.item[5].s_I_NAME,"TRD Sticker
    nord->in_nord.in_item[5].s_OL_I_ID = 1;
    nord->in_nord.in_item[5].s_OL_QUANTITY = 1;
    nord->in_nord.in_item[5].s_OL_SUPPLY_W_ID = 1;
    nord->out_nord.item[5].s_I_PRICE = 4400;
    nord->out_nord.item[5].s_OL_AMOUNT = 35808;
    nord->out_nord.item[5].s_S_QUANTITY = 22;
    nord->out_nord.item[5].s_brand_generic = 'G';
    2");
    nord->out_nord.item[6].s_I_NAME,"TRD Sticker
    nord->in_nord.in_item[6].s_OL_I_ID = 1;
    nord->in_nord.in_item[6].s_OL_QUANTITY = 1;
    nord->in_nord.in_item[6].s_OL_SUPPLY_W_ID = 1;
    nord->out_nord.item[6].s_I_PRICE = 5500;
    nord->out_nord.item[6].s_OL_AMOUNT = 44392;
    nord->out_nord.item[6].s_S_QUANTITY = 21;
    nord->out_nord.item[6].s_brand_generic = 'G';
    3");
    nord->out_nord.item[7].s_I_NAME,"TRD Sticker
    nord->in_nord.in_item[7].s_OL_I_ID = 1;
    nord->in_nord.in_item[7].s_OL_QUANTITY = 1;
    nord->in_nord.in_item[7].s_OL_SUPPLY_W_ID = 1;
    nord->out_nord.item[7].s_I_PRICE = 8300;
    nord->out_nord.item[7].s_OL_AMOUNT = 83410;
    nord->out_nord.item[7].s_S_QUANTITY = 35;
    nord->out_nord.item[7].s_brand_generic = 'G';
    4");
    nord->out_nord.item[8].s_I_NAME,"98 Toyota
    nord->in_nord.in_item[8].s_OL_I_ID = 1;
    nord->in_nord.in_item[8].s_OL_QUANTITY = 1;
    nord->in_nord.in_item[8].s_OL_SUPPLY_W_ID = 1;
    nord->out_nord.item[8].s_I_PRICE = 10000;
    nord->out_nord.item[8].s_OL_AMOUNT = 43160;
    nord->out_nord.item[8].s_S_QUANTITY = 73;
    nord->out_nord.item[8].s_brand_generic = 'G';
    OEM Bra");
    dataSet = 0;
}
return OK;
}
    pymt->out_paym.s_transtatus = 0;
    if (dataSet == 0)
    {
        pymt->out_paym.s_C_CREDIT_LIM = 5000000;
        pymt->out_paym.s_C_DISCOUNT = 1024;
        pymt->out_paym.s_C_BALANCE = 17815;
        pymt->out_paym.s_C_ID = 89;
        pymt->out_paym.s_H_DATE_time = 1234567890;
        strcpy(pymt->out_paym.s_W_STREET_1,"11501 Burnet
        Rd");
        strcpy(pymt->out_paym.s_W_STREET_2,"BLD 905");
        strcpy(pymt->out_paym.s_W_CITY,"Austin");
        strcpy(pymt->out_paym.s_W_STATE,"TX");
        strcpy(pymt->out_paym.s_W_ZIP,"78758");
        strcpy(pymt->out_paym.s_D_STREET_1,"11900 Hobby
        Horse");
        strcpy(pymt->out_paym.s_D_STREET_2,"Apt. 525");
        strcpy(pymt->out_paym.s_D_CITY,"Valley");
        strcpy(pymt->out_paym.s_D_STATE,"TX");
        strcpy(pymt->out_paym.s_D_ZIP,"78559");
        strcpy(pymt->out_paym.s_C_FIRST,"Jim");
        strcpy(pymt->out_paym.s_C_MIDDLE,"F");
        strcpy(pymt->out_paym.s_C_LAST,"Truck");
        strcpy(pymt->out_paym.s_C_STREET_1,"100 N Solis");
        strcpy(pymt->out_paym.s_C_STREET_2,"Flat 343");
        strcpy(pymt->out_paym.s_C_CITY,"Cambridge");
        strcpy(pymt->out_paym.s_C_STATE,"NY");
        strcpy(pymt->out_paym.s_C_ZIP,"785585432");
        strcpy(pymt->out_paym.s_C_PHONE,"1234567890123456");
        pymt->out_paym.s_C_SINCE_time = 0;
        strcpy(pymt->out_paym.s_C_CREDIT,"BC");
        strcpy(pymt->out_paym.s_C_DATA,"XXXXXXXXXXXXXXXXXXXXXXXXXXXX
        XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
        XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
        XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
        XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
        X");
        dataSet = 1;
    }
    else
    {
        pymt->out_paym.s_C_CREDIT_LIM = 4000000;
        pymt->out_paym.s_C_DISCOUNT = 52400;
        pymt->out_paym.s_C_BALANCE = 14080;
        pymt->out_paym.s_C_ID = 3180;
        pymt->out_paym.s_H_DATE_time = 1234567890;
        strcpy(pymt->out_paym.s_W_STREET_1,"1201 Park
        Ave.");
        strcpy(pymt->out_paym.s_W_STREET_2,"Suite 432");
        strcpy(pymt->out_paym.s_W_CITY,"Denver");
        strcpy(pymt->out_paym.s_W_STATE,"CO");
        strcpy(pymt->out_paym.s_W_ZIP,"787562356");
        strcpy(pymt->out_paym.s_D_STREET_1,"3404 Garth
        Rd");
        strcpy(pymt->out_paym.s_D_STREET_2,"Suite 320");
        strcpy(pymt->out_paym.s_D_CITY,"Austin");
        strcpy(pymt->out_paym.s_D_STATE,"TX");
        strcpy(pymt->out_paym.s_D_ZIP,"785598767");
    }
}

```

```

    strcpy(pymt->out_paym.s_C_FIRST,"John");
    strcpy(pymt->out_paym.s_C_MIDDLE,"P");
    strcpy(pymt->out_paym.s_C_LAST,"Williams");
    strcpy(pymt->out_paym.s_C_STREET_1,"North Rab
Road");
    strcpy(pymt->out_paym.s_C_STREET_2,"Apt 343");
    strcpy(pymt->out_paym.s_C_CITY,"La Fiera");
    strcpy(pymt->out_paym.s_C_STATE,"TX");
    strcpy(pymt->out_paym.s_C_ZIP,"785585432");

strcpy(pymt->out_paym.s_C_PHONE,"1234567890123456");
pymt->out_paym.s_C_SINCE_time = 0;
strcpy(pymt->out_paym.s_C_CREDIT,"GC");
strcpy(pymt->out_paym.s_C_DATA,"Great Ebaye");

    dataSet = 0;
}
return OK;
}

extern "C" NULLDB_API int do_ords(struct ords_wrapper *ords,void *ctx)
{
    ords->out_ords.s_transtatus = 0;

    if (dataSet == 0)
    {
        ords->out_ords.s_C_BALANCE = 100000;
        ords->out_ords.s_C_ID = 3;
        ords->out_ords.s_O_ID = 1696;
        ords->out_ords.s_O_CARRIER_ID = 9;
        ords->out_ords.s_ol_cnt = 6;
        ords->out_ords.s_O_ENTRY_D_time = 1234567890;

        strcpy(ords->out_ords.s_C_FIRST,"Homer");
        strcpy(ords->out_ords.s_C_MIDDLE,"J");
        strcpy(ords->out_ords.s_C_LAST,"Simpson");

        ords->out_ords.item[0].s_OL_AMOUNT = 30000;
        ords->out_ords.item[0].s_OL_I_ID = 23492;
        ords->out_ords.item[0].s_OL_SUPPLY_W_ID = 9;
        ords->out_ords.item[0].s_OL_QUANTITY = 5;
        ords->out_ords.item[0].s_OL_DELIVERY_D_time =
1234567890;

        ords->out_ords.item[1].s_OL_AMOUNT = 12300;
        ords->out_ords.item[1].s_OL_I_ID = 18860;
        ords->out_ords.item[1].s_OL_SUPPLY_W_ID = 9;
        ords->out_ords.item[1].s_OL_QUANTITY = 5;
        ords->out_ords.item[1].s_OL_DELIVERY_D_time =
1234567890;

        ords->out_ords.item[2].s_OL_AMOUNT = 15000;
        ords->out_ords.item[2].s_OL_I_ID = 90488;
        ords->out_ords.item[2].s_OL_SUPPLY_W_ID = 9;
        ords->out_ords.item[2].s_OL_QUANTITY = 5;
        ords->out_ords.item[2].s_OL_DELIVERY_D_time =
1234567890;

        ords->out_ords.item[3].s_OL_AMOUNT = 25000;
        ords->out_ords.item[3].s_OL_I_ID = 22741;
        ords->out_ords.item[3].s_OL_SUPPLY_W_ID = 9;
        ords->out_ords.item[3].s_OL_QUANTITY = 5;
        ords->out_ords.item[3].s_OL_DELIVERY_D_time =
1234567890;

        ords->out_ords.item[4].s_OL_AMOUNT = 20000;
        ords->out_ords.item[4].s_OL_I_ID = 92952;
        ords->out_ords.item[4].s_OL_SUPPLY_W_ID = 9;

        ords->out_ords.item[4].s_OL_QUANTITY = 5;
        ords->out_ords.item[4].s_OL_DELIVERY_D_time =
1234567890;

        ords->out_ords.item[5].s_OL_AMOUNT = 2345;
        ords->out_ords.item[5].s_OL_I_ID = 29956;
        ords->out_ords.item[5].s_OL_SUPPLY_W_ID = 9;
        ords->out_ords.item[5].s_OL_QUANTITY = 5;
        ords->out_ords.item[5].s_OL_DELIVERY_D_time =
1234567890;

        dataSet = 1;
    }
    else
    {
        ords->out_ords.s_C_BALANCE = 123000;
        ords->out_ords.s_C_ID = 856;
        ords->out_ords.s_O_ID = 418;
        ords->out_ords.s_O_CARRIER_ID = 10;
        ords->out_ords.s_ol_cnt = 5;
        strcpy(ords->out_ords.s_C_FIRST,"Erick");
        strcpy(ords->out_ords.s_C_MIDDLE,"J");
        strcpy(ords->out_ords.s_C_LAST,"Forman");
        ords->out_ords.s_O_ENTRY_D_time = 1234567890;

        ords->out_ords.item[0].s_OL_AMOUNT = 12000;
        ords->out_ords.item[0].s_OL_I_ID = 54602;
        ords->out_ords.item[0].s_OL_SUPPLY_W_ID = 10;
        ords->out_ords.item[0].s_OL_QUANTITY = 5;
        ords->out_ords.item[0].s_OL_DELIVERY_D_time =
1234567890;

        ords->out_ords.item[1].s_OL_AMOUNT = 2300;
        ords->out_ords.item[1].s_OL_I_ID = 18860;
        ords->out_ords.item[1].s_OL_SUPPLY_W_ID = 10;
        ords->out_ords.item[1].s_OL_QUANTITY = 5;
        ords->out_ords.item[1].s_OL_DELIVERY_D_time =
1234567890;

        ords->out_ords.item[2].s_OL_AMOUNT = 56009;
        ords->out_ords.item[2].s_OL_I_ID = 90488;
        ords->out_ords.item[2].s_OL_SUPPLY_W_ID = 10;
        ords->out_ords.item[2].s_OL_QUANTITY = 5;
        ords->out_ords.item[2].s_OL_DELIVERY_D_time =
1234567890;

        ords->out_ords.item[3].s_OL_AMOUNT = 98000;
        ords->out_ords.item[3].s_OL_I_ID = 22741;
        ords->out_ords.item[3].s_OL_SUPPLY_W_ID = 10;
        ords->out_ords.item[3].s_OL_QUANTITY = 5;
        ords->out_ords.item[3].s_OL_DELIVERY_D_time =
1234567890;

        ords->out_ords.item[4].s_OL_AMOUNT = 25000;
        ords->out_ords.item[4].s_OL_I_ID = 92952;
        ords->out_ords.item[4].s_OL_SUPPLY_W_ID = 10;
        ords->out_ords.item[4].s_OL_QUANTITY = 5;
        ords->out_ords.item[4].s_OL_DELIVERY_D_time =
1234567890;

        dataSet = 0;
    }
    return OK;
}

extern "C" NULLDB_API int do_dlvy(struct dlvy_wrapper *dlvy,void *ctx)
{

```

```

    dlvy->out_dlvy.s_transtatus = 0;

    if (dataSet == 0)
    {
        dataSet = 1;

        for(int districtIndex=0;districtIndex <
DISTRICTS_PER_WAREHOUSE;districtIndex++)
            dlvy->out_dlvy.s_O_ID[districtIndex]= 2055;
    }
    else
    {
        for(int districtIndex=0;districtIndex <
DISTRICTS_PER_WAREHOUSE;districtIndex++)
            dlvy->out_dlvy.s_O_ID[districtIndex]= 2056;

        dataSet = 0;
    }
    return OK;
}

```

```

extern "C" NULLDB_API int do_stok(struct stok_wrapper *stok,void *ctx)
{
    stok->out_stok.s_transtatus = 0;

    if (dataSet == 0)
    {
        stok->out_stok.s_low_stock = 100;

        dataSet = 1;
    }
    else
    {
        stok->out_stok.s_low_stock = 40;

        dataSet = 0;
    }
    return OK;
}

```

NullDB.h

```

// The following ifdef block is the standard way of creating macros which make
exporting
// from a DLL simpler. All files within this DLL are compiled with the
NULLDB_EXPORTS
// symbol defined on the command line. this symbol should not be defined on
any project
// that uses this DLL. This way any other project whose source files include this
file see
// NULLDB_API functions as being imported from a DLL, whereas this DLL
sees symbols
// defined with this macro as being exported.
#ifdef NULLDB_EXPORTS
#define NULLDB_API __declspec(dllexport)
#else
#define NULLDB_API __declspec(dllimport)
#endif

```

```
extern NULLDB_API int dataSet;
```

```

extern "C" NULLDB_API int do_nord(struct nord_wrapper *nord,void *ctx);
extern "C" NULLDB_API int do_pymt(struct pymt_wrapper *pymt,void *ctx);
extern "C" NULLDB_API int do_ords(struct ords_wrapper *ords,void *ctx);
extern "C" NULLDB_API int do_dlvy(struct dlvy_wrapper *dlvy,void *ctx);

```

```
extern "C" NULLDB_API int do_stok(struct stok_wrapper *stok,void *ctx);
```

```

extern "C" NULLDB_API int connect_db(char *dbName,void **ctx);
extern "C" NULLDB_API int disconnect_db(void *ctx);

```

Stdafx.cpp

```

// stdafx.cpp : source file that includes just the standard includes
// tpccsapi.pch will be the pre-compiled header
//          stdafx.obj will contain the pre-compiled type information

```

```
#include "stdafx.h"
```

```

// TODO: reference any additional headers you need in STDAFX.H
// and not in this file

```

Stdafx.h

```

// stdafx.h : include file for standard system include files,
// or project specific include files that are used frequently, but
// are changed infrequently
//

```

```
#pragma once
```

```
#define WIN32_LEAN_AND_MEAN           // Exclude rarely-used
stuff from Windows headers
```

```
#define _ATL_CSTRING_EXPLICIT_CONSTRUCTORS // some
CString constructors will be explicit
```

```

// turns off ATL's hiding of some common and often safely ignored warning
messages

```

```
#define _ATL_ALL_WARNINGS
```

```

// critical error descriptions will only be shown to the user
// in debug builds. they will always be logged to the event log
#ifdef _DEBUG

```

```
#define ATL_CRITICAL_ISAPI_ERROR_LOGONLY
#endif
```

```

#ifdef _WIN32_WINNT
#define _WIN32_WINNT 0x0403
#endif

```

```

// TODO: this disables support for registering COM objects
// exported by this project since the project contains no
// COM objects or typelib. If you wish to export COM objects
// from this project, add a typelib and remove this line
#define _ATL_NO_COM_SUPPORT

```

```

#include "resource.h"
#include <atlsrvres.h>
#include <atlisapi.h>
#include <atlstencil.h>

```

```
// TODO: reference additional headers your program requires here
```

Stdafx.cpp

```

// stdafx.cpp : source file that includes just the standard includes
// tpccComClient.pch will be the pre-compiled header

```

```
// stdafx.obj will contain the pre-compiled type information

#include "stdafx.h"

// TODO: reference any additional headers you need in STDAFX.H
// and not in this file
```

Stdafx.h

```
// stdafx.h : include file for standard system include files,
// or project specific include files that are used frequently, but
// are changed infrequently
//
```

```
#pragma once
#include <iostream>
#include <tchar.h>
```

```
// TODO: reference additional headers your program requires here
```

TpccComClient.cpp

```
// tpccComClient.cpp : Defines the entry point for the console application.
//
```

```
#include "stdafx.h"
```

```
#include "..\tpccCom\tpccCom.h"
#include "..\tpccCom\tpccCom_i.c"
#include <tpcc.h>
```

```
struct txn_buffer
{
    char    *dataBuffer;
    int     size;
};
```

```
int _tmain(int argc, _TCHAR* argv[])
{
```

```
    HRESULT hres;
    Itpcc_com * pTxn;

    hres = CoInitialize(NULL);
    if (FAILED(hres))
    {
        printf("Error : CoInitialize() failed
rc:%d\n",GetLastError());
        fflush(stdout);
        return 0;
    }
```

```
    hres =
CoCreateInstance(CLSID_tpcc_com,NULL,CLSCTX_SERVER,IID_Itpcc_co
m,(void **)&pTxn);
    if (FAILED(hres))
    {
        printf("Error : CoCreateInstance() failed rc:%d
hres:%X\n",GetLastError(),hres);
        fflush(stdout);
        return 0;
    }
```

```
//int size = sizeof(in_stocklev_struct);
//int size2 = sizeof(out_stocklev_struct);

//define txn buffer to store txn structure in
```

```
    struct txn_buffer    comBuffer;
    comBuffer.dataBuffer = (char *)
CoTaskMemAlloc(sizeof(STOCKLEVELDATA));
    if (!(comBuffer.dataBuffer))
    {
        printf(comBuffer.dataBuffer,"CoTaskMemAlloc failed,
rc:%d\n",GetLastError());
        return(TRUE);
    };
    comBuffer.size =
sizeof(STOCKLEVELDATA);

    struct STOCKLEVELDATA *pStock;
    pStock = (STOCKLEVELDATA*)comBuffer.dataBuffer;
    ZeroMemory(pStock,comBuffer.size);

    //initialize fields
    pStock->in_s_W_ID = 10; pStock->in_s_D_ID = 1;
    pStock->in_s_threshold = 2; pStock->out_s_transtatus = -1;

    int dataLen = comBuffer.size;
    try{
        hres = pTxn->doStockLevel(&dataLen,(unsigned
char**)&comBuffer.dataBuffer);
    }
    catch(...)
    {
        printf("Error : StockLevel() com caused exeception failed
rc:%d\n",GetLastError());
        fflush(stdout);
        return 0;
    }
    if (FAILED(hres))
    {
        printf("Error : StockLevel() com call failed
rc:%d\n",GetLastError());
        return 0;
    }

    pStock = (STOCKLEVELDATA *)comBuffer.dataBuffer;

    printf("Stock Level txn complete.
s_transtatus:%d\n",pStock->out_s_transtatus);

    return 0;
    return 0;
}
```

HtmlPhraser.cpp

```
////////////////////////////////////
// htmlPhraser.cpp
////////////////////////////////////
// Class implmentation of htmlPhraser.
// This class will take a query string and break it into a series
// of consituant parts
////////////////////////////////////
#include "htmlPhraser.h"

////////////////////////////////////
// htmlPhraser:htmlPhraser
////////////////////////////////////
// Title : Constructor
// Parameters : char * query string
// Return Value : None
```

```

// Comments      :
///////////////////////////////////////////////////////////////////

htmlPhraser::htmlPhraser(char *queryString)
{
    // initialize query values
    iCustomerIdFlag = iCarrierNumFlag = iStockThresholdFlag = false;

    // this initializes the query list to NULL's. This means that
    // characters being added are overwriting null characters and
    // therefore the string will be null terminated implicitly.

    memset(iQueryValues, NULL, (MAX_FIELD_NUM *
MAX_FIELD_LEN));

    // controls
    char          queryChar          = NULL;

    int          queryIndex          = -1;
    int          valueIndex          = -1;

    // process each character of query string
    while(*queryString)
    {
        // check for special case characters
        if(queryChar)
        {
            // a percentage sign would indicate a token
            if(*queryString != '%')
            {
                // a plus sign represents a space
                if(*queryString == '+')
                {
                    queryChar = ' ';
                    *queryString++;
                }
                else queryChar = *queryString++;
            }
            else queryChar =
convertQueryToken(&queryString);
        }
        else queryChar = '&';

        // handle query reference (&)
        if(queryChar == '&')
        {
            // reset value index
            valueIndex = -1;

            // do we have a numeric query reference
            if(*queryString >= '0' && *queryString <=
'9')
            {
                // numeric query id
                queryIndex =
10) + (*(queryString + 1) - '0');

                // walk past the two command
                queryChar += 2;

                // validate query value
                if(queryIndex >
MAX_QUERY_ID)
                {
                    queryIndex = -1;
                }
                else queryIndex = -1;
            }
        }
    }

    // finished processing for query reference
    continue;
}

// we have a query reference but need to wait until we see
// before accepting value

if(valueIndex == -1)
{
    // we are waiting for '='
    if(queryChar == '=')
    {
        valueIndex = 0;

        // set query string flags
        switch(queryIndex)
        {
            case C_ID:
                iCustomerIdFlag = true;
            case CARRIER_NUM:
                iCarrierNumFlag =
true; break;
            case STK_THRESHOLD:
                iStockThresholdFlag =
true; break;
            default: break;
        }
    }

    // finishes looking for '='
    continue;
}

// add each character to the query value
if(queryIndex > -1 && valueIndex > -1)
{
    // we are processing a query value
    if(valueIndex < MAX_FIELD_LEN)
    {
        // we have not exceeded max line
        len
iQueryValues[queryIndex][valueIndex++] = queryChar;
        }
        else
        {
            continue;
        }
    }
}

return;
}

/////////////////////////////////////////////////////////////////
// htmlPhraser::getCommandId
/////////////////////////////////////////////////////////////////
// Title      : Returns the page command
// Parameters  : None
// Return Value : int - page command
// Comments    :
/////////////////////////////////////////////////////////////////

int htmlPhraser::getCommandId()
{
    // return command numeric code
    switch(*iQueryValues[COMMAND_ID])
    {

```

```

    case NEW_ORDER_CODE:
        if(iCustomerIdFlag)
            return
        COMMAND_NEW_ORDER_RESULTS;
    else return COMMAND_NEW_ORDER;
    case PAYMENT_CODE:
        if(iCustomerIdFlag)
            return COMMAND_PAYMENT_RESULTS;
        else return COMMAND_PAYMENT;
    case ORDER_STATUS_CODE:
        if(iCustomerIdFlag)
            return
        COMMAND_ORDER_STATUS_RESULTS;
    else return COMMAND_ORDER_STATUS;
    case DELIVERY_CODE:
        if(iCarrierNumFlag)
            return COMMAND_DELIVERY_RESULTS;
        else return COMMAND_DELIVERY;
    case STOCK_CODE:
        if(iStockThresholdFlag)
            return COMMAND_STOCK_RESULTS;
        else return COMMAND_STOCK;
    case MENU_CODE:
        return COMMAND_LOGIN_RESULTS;
    case EXIT_CODE:
        return COMMAND_EXIT;
    default:
        return COMMAND_LOGIN;
};

```

```

// should not get here
return COMMAND_LOGIN;
}

```

```

////////////////////////////////////
// htmlPhraser::validate
////////////////////////////////////
// Title   : validate url parameter list for all txn types
// Parameters   : int - txn type
// Return Value   : int - error code
// Comments     :
////////////////////////////////////

```

```

int validate(int txnType)
{
    return 0;
}

```

```

////////////////////////////////////
// htmlPhraser::convertQueryToken
////////////////////////////////////
// Title   : Returns the page command
// Parameters   : None
// Return Value   : int - page command
// Comments     :
////////////////////////////////////

```

```

char htmlPhraser::convertQueryToken(char **queryString)
{
    char queryChar = NULL;

    // skip over %
    (*queryString)++;

    // look at first character
    switch(**queryString)

```

```

{
    case '2':
        {
            // what follows?
            (*queryString)++;

            switch(**queryString)
            {
                case '1':
                    queryChar = '!';
                    break;
                case '3':
                    queryChar = '#';
                    break;
                case '4':
                    queryChar = '$';
                    break;
                case '5':
                    queryChar = '%';
                    break;
                case '6':
                    queryChar = '&';
                    break;
                case '8':
                    queryChar = '(';
                    break;
                case '9':
                    queryChar = ')';
                    break;
                case 'B':
                    queryChar = '+';
                    break;
                case 'C':
                    queryChar = ',';
                    break;
                case 'F':
                    queryChar = '/';
                    break;
                case ' ':
                    queryChar = ' ';
                    break;
            }
        }
}

```

```

break;
case '3':
    {
        // what follows?
        (*queryString)++;

        switch(**queryString)
        {
            case 'A':
                queryChar = '!';
                break;
            case 'B':
                queryChar = '!';
                break;
            case 'D':
                queryChar = '=';
                break;
            case 'F':
                queryChar = '?';
                break;
            case ' ':
                queryChar = ' ';
                break;
        }
    }
}

```

```

    }
break;
case '4':
{
    // what follows?
    (*queryString)++;

    switch(**queryString)
    {
    case '0':
        queryChar = '@';
        break;
    case ':':
        queryChar = ':';
        break;
    }
}
break;
case '5':
{
    // what follows?
    (*queryString)++;

    switch(**queryString)
    {
    case 'B':
        queryChar = '[';
        break;
    case 'D':
        queryChar = ']';
        break;
    case 'E':
        queryChar = '^';
        break;
    case ':':
        queryChar = ':';
        break;
    }
}
break;
case '7':
{
    // what follows?
    (*queryString)++;

    switch(**queryString)
    {
    case 'B':
        queryChar = '{';
        break;
    case 'C':
        queryChar = '|';
        break;
    case 'D':
        queryChar = '}';
        break;
    case 'E':
        queryChar = '~';
        break;
    case ':':
        queryChar = ':';
        break;
    }
}

```

```

        break;
    case '+':
        queryChar = '+';
        break;
    }
    // advance pointer and return
    (*queryString)++; return queryChar;
}

```

HtmlPhraser.h

```

/////////////////////////////////////////////////////////////////
// htmlPhraser.h
/////////////////////////////////////////////////////////////////
// Class to decode a html query string
/////////////////////////////////////////////////////////////////

#pragma once
#include <memory.h>

/////////////////////////////////////////////////////////////////
// Definitions
/////////////////////////////////////////////////////////////////

#define NULL 0
#define COMMAND_ID 0
#define TERM_ID 1
#define W_ID 2
#define D_ID 3
#define C_ID 4
#define C_NAME 5

#define C_W_ID 6
#define C_D_ID 7
#define AMT_PAID 8

#define STK_THRESHOLD 9
#define CARRIER_NUM 10

#define ITEM_LIST_START 11
#define ITEM_LIST_FINISH 55

#define MAX_QUERY_ID 55
#define MAX_FIELD_LEN 256
#define MAX_FIELD_NUM 56

/////////////////////////////////////////////////////////////////
// Command Codes
/////////////////////////////////////////////////////////////////

#define NEW_ORDER_CODE 'n'
#define PAYMENT_CODE 'p'
#define ORDER_STATUS_CODE 'o'
#define DELIVERY_CODE 'd'
#define STOCK_CODE 's'

```



```

#define EXIT_CODE
'e'
#define MENU_CODE
'm'

#define COMMAND_LOGIN
#define COMMAND_NEW_ORDER
#define COMMAND_PAYMENT
2
#define COMMAND_ORDER_STATUS
#define COMMAND_DELIVERY
#define COMMAND_STOCK
5
#define COMMAND_EXIT
6

#define COMMAND_LOGIN_RESULTS
7
#define COMMAND_NEW_ORDER_RESULTS
#define COMMAND_PAYMENT_RESULTS
9
#define COMMAND_ORDER_STATUS_RESULTS
10
#define COMMAND_DELIVERY_RESULTS
11
#define COMMAND_STOCK_RESULTS
12

////////////////////////////////////
// Class htmlPhraser
////////////////////////////////////

class htmlPhraser
{
    // Constructors / Destructor
    public:
        htmlPhraser(char *queryString);
        ~htmlPhraser()

{return;}

    // getters
    public:
        int getCommandId();
        int validate(int txnType);

        char * get_TERM_ID()
{return iQueryValues[TERM_ID];}
        char * get_W_ID()
{return iQueryValues[W_ID];}
        char * get_D_ID()
{return iQueryValues[D_ID];}
        char * get_C_ID()
{return iQueryValues[C_ID];}
        char * get_C_NAME()
{return iQueryValues[C_NAME];}
        char * get_C_W_ID()
{return iQueryValues[C_W_ID];}
        char * get_C_D_ID()
{return iQueryValues[C_D_ID];}
        char * get_AMT_PAID()
{return iQueryValues[AMT_PAID];}
        char * get_STK_THRESHOLD()
{return iQueryValues[STK_THRESHOLD];}
        char * get_CARRIER_NUM()
{return iQueryValues[CARRIER_NUM];}

        char * get_ITEM_SUPP_W(int item) {return
iQueryValues[(ITEM_LIST_START + 0) + (item * 3)];}

```

```

        char * get_ITEM_ITEM_NUM(int item)
{return iQueryValues[(ITEM_LIST_START + 1) + (item * 3)];}
        char * get_ITEM_QTY(int item)
{return iQueryValues[(ITEM_LIST_START + 2) + (item * 3)];}

    // Class Functions
    private:
        char convertQueryToken(char **queryString);

    // Class Attributes
    private:
        int iCustomerIdFlag;
        int iCarrierNumFlag;
        int iStockThresholdFlag;

        char
iQueryValues[MAX_FIELD_NUM][MAX_FIELD_LEN];
};

```

Resource.h

```

//{{NO_DEPENDENCIES}}
// Microsoft Visual C++ generated include file.
// Used by tpccsapi.rc
//
#define IDS_PROJNAME 100

// Next default values for new objects
//
#ifdef APSTUDIO_INVOKED
#ifndef APSTUDIO_READONLY_SYMBOLS
#define _APS_NEXT_RESOURCE_VALUE 201
#define _APS_NEXT_COMMAND_VALUE 32768
#define _APS_NEXT_CONTROL_VALUE 201
#define _APS_NEXT_SYMED_VALUE 101
#endif
#endif

```

StdAfx.cpp

```

// stdafx.cpp : source file that includes just the standard includes
// tpccsapi.pch will be the pre-compiled header
// stdafx.obj will contain the pre-compiled type information

```

```
#include "stdafx.h"
```

```

// TODO: reference any additional headers you need in STDAFX.H
// and not in this file

```

StdAfx.h

```

// stdafx.h : include file for standard system include files,
// or project specific include files that are used frequently, but
// are changed infrequently
//

```

```
#pragma once
```

```

#define WIN32_LEAN_AND_MEAN // Exclude rarely-used
stuff from Windows headers

```

```

#define _ATL_CSTRING_EXPLICIT_CONSTRUCTORS // some
CString constructors will be explicit

// turns off ATL's hiding of some common and often safely ignored warning
messages
#define _ATL_ALL_WARNINGS

// critical error descriptions will only be shown to the user
// in debug builds. they will always be logged to the event log
#ifndef _DEBUG
#define ATL_CRITICAL_ISAPI_ERROR_LOGONLY
#endif

#ifndef _WIN32_WINNT
#define _WIN32_WINNT 0x0403
#endif

// TODO: this disables support for registering COM objects
// exported by this project since the project contains no
// COM objects or typelib. If you wish to export COM objects
// from this project, add a typelib and remove this line
#define _ATL_NO_COM_SUPPORT

#include "resource.h"
#include <atlsrvres.h>
#include <atlisapi.h>
#include <atlstencil.h>

// TODO: reference additional headers your program requires here

Tpcc.h

// Common defines and structures use internally by client code
// Not to be confused with structures actually passed in transactions
//

// standard includes

#ifndef _COMMON_TPCC
#define _COMMON_TPCC
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <sys/timeb.h>
#include <time.h>

#include <db2tpcc.h>
#include <iostream>
#include <fstream>
#include <process.h>
#include <ios>

////////////////////////////////////
// Defines
////////////////////////////////////

#define OK
0
#define INVALID_STATUS -1
#define ERR
-1
#define INVALID_COM_STATUS -2

#define TXN_MAX_COMMANDS 55

#define MAX_TRANSACTIONS 14
#define MAX_CMD_LENGTH 100
#define INPUT_ITEMS 3
#define MAX_INT_BUFFER 15
#define NORD_ITEMS 15
#define ITEM_START 11
#define ITEM_END 55
#define MAX_ITEMS 15

#define MAX_STRING_LEN 256
#define MAX_HTML_PAGE_LEN 4096
#define MAX_HTML_HEADER_LEN 512

#define DELIVERY_THREADS_NUM 100

#define DISTRICTS_PER_WAREHOUSE 10
////////////////////////////////////
// Transaction Codes
////////////////////////////////////

#define TXN_LOGIN 0
#define TXN_NEW_ORDER 1
#define TXN_PAYMENT 2
#define TXN_ORDER_STATUS 3
#define TXN_DELIVERY 4
#define TXN_STOCK 5
#define TXN_EXIT 6
#define TXN_LOGIN_RESULTS 7
#define TXN_NEW_ORDER_RESULTS 8
#define TXN_PAYMENT_RESULTS 9
#define TXN_ORDER_STATUS_RESULTS 10
#define TXN_DELIVERY_RESULTS 11
#define TXN_STOCK_RESULTS 12

#define CMD_NORD "nord"
#define CMD_PYMT "pymt"
#define CMD_ORDS "ords"
#define CMD_DLVY "dlvy"
#define CMD_STOK "stok"
#define CMD_EXIT "exit"
#define CMD_MENU "menu"

#define APP_NAME "tpcc.html"
#define HEADER "Content-Type:text/html\r\nContent-Length: %d\r\nConnection: Keep-Alive\r\n\r\n"

////////////////////////////////////
// URL Commands
////////////////////////////////////

```

#define	CMD_TXN_ID		#define	INITIALS_LEN	2
"00"			#define	CREDIT_LEN	2
#define	CMD_TERM_ID		#define	STREET_LEN	20
"01"			#define	CITY_LEN	20
#define	CMD_W_ID		#define	STATE_LEN	2
"02"			#define	ZIP_LEN	9
#define	CMD_D_ID		#define	PHONE_LEN	16
"03"			#define	DATA_LEN	200
#define	CMD_C_ID		#define	ITEM_LIST	15
"04"			#define	ORDER_LIST	10
#define	CMD_C_NAME		////////////////////////////////////		
"05"			// Type definitions		
#define	CMD_C_W_ID		////////////////////////////////////		
"06"			typedef __int8	INT8b;	
#define	CMD_C_D_ID		typedef __int16	INT16b;	
"07"			typedef __int32	INT32b;	
#define	CMD_AMT_PAID	"08"	typedef __int64	INT64b;	
#define	CMD_STK_THRESHOLD	"09"	typedef unsigned __int8	UINT8b;	
#define	CMD_CARRIER_NUM	"10"	typedef unsigned __int16	UINT16b;	
#define	ITEM01_SUPP_W	"11"	typedef unsigned __int32	UINT32b;	
#define	ITEM01_ITEM_NUM	"12"	typedef unsigned __int64	UINT64b;	
#define	ITEM01_OTY	"13"	typedef INT16b	sqlint16;	
#define	CHAR_FILL	' '	typedef INT32b	sqlint32;	
#define	NUMERIC_FILL	'.'	typedef INT64b	sqlint64;	
#define	NEGITIVE_SYMBOL	'-'	typedef INT16b	int16_t;	
#define	MONEY_SYMBOL	'\$'	typedef INT32b	int32_t;	
#define	DECIMAL_SYMBOL	'.'	typedef INT64b	int64_t;	
#define	ZERO_SYMBOL	'0'	typedef char	BYTE8b;	
#define	ZIP_DELIMITER	'.'	typedef double	DOUBLE;	
#define	PHONE_DELIMITER	'.'	typedef unsigned long	NATURAL;	
#define	DATE_DELIMITER	'.'	////////////////////////////////////		
#define	TIME_DELIMITER	'.'	// Date and time values		
#define	DEFAULT_MONEY64_LEN	15	////////////////////////////////////		
#define	DEFAULT_MONEY32_LEN	9	#define SECONDS_IN_DAY	86400	
#define	DEFAULT_MONEY16_LEN	9	#define SECONDS_IN_HOUR	3600	
#define	DEFAULT_NUMERIC64_LEN	15	#define SECONDS_IN_MINUTE	60	
#define	DEFAULT_NUMERIC32_LEN	9	#define GMT_OFFSET	5	
#define	DEFAULT_NUMERIC16_LEN	9	#define DAYS_IN_YEAR	365	
#define	DEFAULT_DECIMAL64_LEN	5	#define YEARS_IN_LEAP	4	
#define	DEFAULT_DECIMAL32_LEN	5	#define START_YEAR	1970	
#define	DEFAULT_DECIMAL16_LEN	5	#define MONTHS_IN_YEAR	12	
#define	DEFAULT_DATETIME_LEN	19	////////////////////////////////////		
#define	DEFAULT_DATE_LEN	11	// Error codes		
#define	DEFAULT_TIME_LEN	8	////////////////////////////////////		
#define	DEFAULT_STRING_LEN	25	#define ERR_INVALID_TXN_TYPE	-1	
#define	DEFAULT_ZIP_LEN	17	#define ERR_MISSING_W_ID	-2	
#define	DEFAULT_PHONE_LEN	18	#define ERR_NON_NUMERIC_W_ID	-3	
////////////////////////////////////			#define ERR_MISSING_D_ID	-4	
// String Field Lengths			#define ERR_NON_NUMERIC_D_ID	-5	
////////////////////////////////////			#define ERR_MISSING_C_ID	-6	
#define NAME_LEN	24		#define ERR_NON_NUMERIC_C_ID	-7	
#define LAST_NAME_LEN	16				
#define FIRST_NAME_LEN	16				

```

#define ERR_MISSING_SUPP_W -8
#define ERR_NON_NUMERIC_SUPP_W -9
#define ERR_MISSING_ITEM_NUM -10
#define ERR_NON_NUMERIC_ITEM_NUM -11
#define ERR_MISSING_ITEM_OTY -12
#define ERR_NON_NUMERIC_ITEM_QTY -13

#define ERR_MISSING_CLAST_NAME -14
#define ERR_NON_NUMERIC_CUST_W_ID -15
#define ERR_NON_NUMERIC_CUST_D_ID -16
#define ERR_MISSING_AMOUNT_PAID -17
#define ERR_NON_NUMERIC_AMOUNT_PAID -18

#define ERR_INVALID_D_ID "ERROR: Invalid District ID. Try Again."
#define ERR_INVALID_W_ID "ERROR: Invalid Warehouse ID. Try Again."
#define ERR_INVALID_C_ID "ERROR: Invalid Customer ID. Try Again."
#define ERR_INVALID_SUPPLY_W_ID "ERROR: Invalid Item Supply Warehouse. Try Again."
#define ERR_INVALID_ITEM_NUM "ERROR: Invalid Item Number. Try Again."
#define ERR_INVALID_ITEM_OTY "ERROR: Invalid Item Qty. Try Again."
#define ERR_MISSING_C_ID_OR_CLAST "ERROR: Must Enter Customer Id or Customer Last Name. Try Again."
#define ERR_INVALID_PAYMENT_AMOUNT "ERROR: Invalid Payment Amount. Try Again."
#define ERR_INVALID_CARRIER "ERROR: Invalid Carrier Number. Try Again."
#define ERR_INVALID_THRESHOLD "ERROR: Invalid Threshold. Try Again."
#define ERR_INVALID_C_D_ID "ERROR: Invalid Customer District Id. Try Again."
#define ERR_INVALID_C_W_ID "ERROR: Invalid Customer Warehouse Id. Try Again."
#define ERR_TERMINAL_FULL "ERROR: Terminal can not support user. Terminal full."
#define ERR_C_ID_OR_CLAST_ONLY "ERROR: Either customer id or customer last name can be specified."

#define ERR_UNABLE_TO_OPEN_REG -50
#define ERR_DLVY_THREAD_FAILED -51
#define ERR_DLVY_SEMAPHORE_INIT_FAILED -52
#define ERR_DLVY_EVENT_INIT_FAILED -53
#define ERR_DLVY_QUEUE_EATING_TAIL -54

#define ERR_INVALID_USERNAME -70
#define ERR_INVALID_PASSWORD -71
#define ERR_INVALID_DB_NAME -72
#define ERR_INVALID_REGISTRY_KEY -73
#define ERR_DB2_DLL_NOT_LOADED -74
#define ERR_ORACLE_DLL_NOT_LOADED -75
#define ERR_CONNECT_ADDRESS_NOT_FOUND -76
#define ERR_NORD_ADDRESS_NOT_FOUND -77
#define ERR_PYMT_ADDRESS_NOT_FOUND -78
#define ERR_ORDS_ADDRESS_NOT_FOUND -79
#define ERR_DLVY_ADDRESS_NOT_FOUND -80
#define ERR_STOK_ADDRESS_NOT_FOUND -81
#define ERR_NULL_DLL_NOT_LOADED -82
#define ERR_UNKNOWN_DB -83

#define ERR_DISCONNECT_ADDRESS_NOT_FOUND -84
#define ERR_SAVING_CONTEXT -90
#define ERR_DETACHING_CONTEXT -91
#define ERR_ATTACHING_CONTEXT -92
#define ERR_HANDLE_IN_USE -93
#define ERR_CONNECT_TO_TM_FAILED -99
#define ERR_DLVY_LOG_OPEN_FAILED -100
#define ERR_DLVY_QUEUE_FULL -101

// Registry Definitions
#define REGISTRY_SUB_KEY "SOFTWARE\TPCC"
#define DELIVERY_THREADS "dlvyThreads"
#define DELIVERY_QUEUE_LEN "dlvyQueueLen"
#define DELIVERY_LOG_PATH "dlvyLogPath"
#define ERROR_LOG_FILE "errorLogFile"
#define HTML_TRACE_LOG_FILE "htmlTraceLogFile"
#define DB_NAME "dbName"
#define NULL_DB "nullDB"
#define COM_NULL_DB "comnullDB"
#define CLIENT_NULL_DB "clientNullDB"
#define NUM_USERS "numUsers"
#define DB_TYPE "dbType"
#define TXN_MONITOR "txn_server"
#define COMM_POOL "comm_pool"
#define HTML_TRACE "htmlTrace"
#define ISAPI_TRACE "isapi_trace"
#define DEFAULT_DLVY_THREADS 1
#define DEFAULT_DLVY_QUEUE_LEN 10
#define DEFAULT_DLVY_LOG_PATH "c:\inetpub\wwwroot\tpcc\dlvy"
#define DEFAULT_ERROR_LOG_FILE "c:\inetpub\wwwroot\tpcc\errorLog.txt"
#define DEFAULT_HTML_TRACE_LOG_FILE "c:\inetpub\wwwroot\tpcc\htmlTrace.txt"
#define DEFAULT_NUM_USERS 10000
#define DEFAULT_DB_NAME "tpcc"

```

```

// Structure defines
///////////////////////////////////////////////////////////////////

struct nord_wrapper {
    struct in_neword_struct in_nord;
    struct out_neword_struct out_nord;
};

struct paym_wrapper {
    struct in_payment_struct in_paym;
    struct out_payment_struct out_paym;
};

struct ords_wrapper {
    struct in_ordstat_struct in_ords;
    struct out_ordstat_struct out_ords;
};

struct dlvy_wrapper {
    struct in_delivery_struct in_dlvy;
    struct out_delivery_struct out_dlvy;
};

struct stok_wrapper {
    struct in_stocklev_struct in_stok;
    struct out_stocklev_struct out_stok;
};

typedef struct
{
    int year;
    int month;
    int day;

    int hour;
    int minute;
    int second;
} datetime;

struct NEWORDERDATA
{
    struct in_items_struct {
        int s_OL_I_ID;
        int s_OL_SUPPLY_W_ID;
        short s_OL_QUANTITY;
    } in_item[15];

    long long in_s_O_ENTRY_D_time; /* init by SUT */
    int in_s_C_ID;
    int in_s_W_ID;
    short in_s_D_ID;
    short in_s_O_OL_CNT; /* init by SUT */
    short in_s_all_local;
    short in_duplicate_items;

    struct out_items_struct {
        double s_I_PRICE;
        double s_OL_AMOUNT;
        short s_S_QUANTITY;
        char s_I_NAME[25];
        char s_brand_generic;
    } out_item[15];

    long long out_s_O_ENTRY_D_time;
    double out_s_W_TAX;
    double out_s_D_TAX;
    double out_s_C_DISCOUNT;
    double out_s_total_amount;

```

```

    int out_s_O_ID;
    short out_s_O_OL_CNT;
    short out_s_transtatus;
    short out_deadlocks;
    char out_s_C_LAST[17];
    char out_s_C_CREDIT[3];
};

struct PAYMENTDATA
{
    long long in_s_H_DATE_time;
    double in_s_H_AMOUNT;
    int in_s_W_ID;
    int in_s_C_W_ID;
    int in_s_C_ID;
    short in_s_C_D_ID;
    short in_s_D_ID;
    char in_s_C_LAST[17];

    long long out_s_H_DATE_time;
    long long out_s_C_SINCE_time;
    double out_s_C_CREDIT_LIM;
    double out_s_C_BALANCE;
    double out_s_C_DISCOUNT;
    int out_s_C_ID;
    short out_s_transtatus;
    short out_deadlocks;
    char out_s_W_STREET_1[21];
    char out_s_W_STREET_2[21];
    char out_s_W_CITY[21];
    char out_s_W_STATE[3];
    char out_s_W_ZIP[10];
    char out_s_D_STREET_1[21];
    char out_s_D_STREET_2[21];
    char out_s_D_CITY[21];
    char out_s_D_STATE[3];
    char out_s_D_ZIP[10];
    char out_s_C_FIRST[17];
    char out_s_C_MIDDLE[3];
    char out_s_C_LAST[17];
    char out_s_C_STREET_1[21];
    char out_s_C_STREET_2[21];
    char out_s_C_CITY[21];
    char out_s_C_STATE[3];
    char out_s_C_ZIP[10];
    char out_s_C_PHONE[17];
    char out_s_C_CREDIT[3];
    char out_s_C_DATA[201];
};

struct ORDERSTATUSDATA
{
    int in_s_C_ID;
    int in_s_W_ID;
    short in_s_D_ID;
    char in_s_C_LAST[17];

    double out_s_C_BALANCE;
    long long out_s_O_ENTRY_D_time;
    int out_s_C_ID;
    int out_s_O_ID;
    short out_s_O_CARRIER_ID;
    short out_s_ol_cnt;
    struct out_oitems_struct {
        long long s_OL_DELIVERY_D_time;
        double s_OL_AMOUNT;
        int s_OL_I_ID;

```

```

        int s_OL_SUPPLY_W_ID;
        short s_OL_QUANTITY;
    } out_item[15];
    short out_s_transtatus;
    short out_deadlocks;
    char out_s_C_FIRST[17];
    char out_s_C_MIDDLE[3];
    char out_s_C_LAST[17];
};

struct DELIVERYDATA
{
    long long in_s_O_DELIVERY_D_time;
    int in_s_W_ID;
    short in_s_O_CARRIER_ID;
    int out_s_O_ID[10];
    short out_s_transtatus;
    short outdeadlocks;
};

struct STOCKLEVELDATA
{
    int in_s_threshold;
    int in_s_W_ID;
    short in_s_D_ID;

    int out_s_low_stock;
    short out_s_transtatus;
    short out_deadlocks;
};

// MISCELLANEOUS HELPER FUNCTIONS
inline void appendText(char **string,char *text);
inline void appendText(char **string,char *text,int length,int justify);
inline void appendChar(char **string,char byte);
inline void DEBUGMSG(FILE * debugFile, char * message);
inline void appendSpaces(char **string,int spaces);

inline void calcOutDateTime(const INT64b value,datetime *timestamp);
inline int copyOutPhone(char *buffer,char *value,int len);
inline bool copyInMoney64(const char * value,INT64 *number);
inline int copyInMoney(const char *value);
inline void copyOutMoney64(char *buffer,INT64b value,unsigned int len);
inline int copyOutDateTime(char *buffer,INT64b value);
inline int copyOutDate(char *buffer,INT64b value);
inline int copyOutTime(char *buffer,INT64b value);
inline int copyOutDecimal64(char *buffer,INT64b value,unsigned int len);

inline UINT16b changeOrder16(UINT16b value);
inline UINT32b changeOrder32(UINT32b value);
inline UINT64b changeOrder64(UINT64b value);

inline INT16b changeOrder16(INT16b value);
inline INT32b changeOrder32(INT32b value);
inline INT64b changeOrder64(INT64b value);

//
// Name      : appendText
// Description :
//           Append text to string
// Parameters :
//           char ** - string point to append to
//           char * - text to append
// Returns   :

```

```

// None
// Comments  :
//
inline void appendText(char **string,char *text)
{
    while(*text)
    {
        *(*string)++ = *text++;
    }

    **string='\0';
    return;
}

//
// Name      : appendText
// Description :
//           Append text to string
// Parameters :
//           char ** - string point to append to
//           char * - text to append
//           int - total field length including
//           blank spaces
//           int - justify flag
// Returns   :
//           None
// Comments  :
//           right justify
//           left justify

inline void appendText(char **string,char *text,int length,int justify)
{
    int byteCount = 0;

    if(justify)
    {
        while(*text)
        {
            *(*string)++ = *text++;
            byteCount++;
        }

        //append blank spaces if text is less than length at end
        for(byteCount;byteCount < length;byteCount++)
            *(*string)++ = ' ';
    }
    else
    {
        long long textLen = strlen(text);
        for(textLen;textLen < length;textLen++)
            *(*string)++ = ' ';

        while(*text)
            *(*string)++ = *text++;
    }

    **string='\0';
}

// Name      : appendChar
// Description :
//           Append text to string
// Parameters :
//           char ** - string point to append to
//           char * - text to append

```



```

        timestamp->month += 1;
        days = days -
// copyOutPhone
//
// Title           : Copy phone data out of class array
// Parameters      : char * - buffer to copy phone string into
//
// Return Value    : int - Length of copy
// Comments       :
//
// increment year and
        timestamp->year += 1;
// are we now on a leap
        leap =
!(timestamp->year % YEARS_IN_LEAP);
    }
    else
    {
        // set day of month to remaining
        timestamp->day = days; days = 0;
    }
    // set time values to remaining seconds
    timestamp->hour = hms / SECONDS_IN_HOUR;
    hms = hms % SECONDS_IN_HOUR;
    timestamp->minute = hms / SECONDS_IN_MINUTE;
    timestamp->second = hms % SECONDS_IN_MINUTE;
    return;
}

//
// copyOutZip
//
// Title           : Copy zip data out of class array
// Parameters      : char * - buffer to copy zip string into
//
// Return Value    : int - Length of copy
// Comments       :
//
inline int copyOutZip(char *buffer,char *value,int len = DEFAULT_ZIP_LEN)
{
    int index      = 0;
    int bufferPos  = 0;

    // add each digit of zip number to buffer inserting delimiter at 5
    while(value[index] && bufferPos < len)
    {
        if(index == 5)
            buffer[bufferPos++] = ZIP_DELIMITER;

        buffer[bufferPos++] = value[index++];
    }

    // space fill to the required length
    while(bufferPos < len)
        buffer[bufferPos++] = CHAR_FILL;

    buffer[bufferPos] = NULL;
    return len;
}

timestamp->month += 1;
days = days -
// copyOutPhone
//
// Title           : Copy phone data out of class array
// Parameters      : char * - buffer to copy phone string into
//
// Return Value    : int - Length of copy
// Comments       :
//
// increment year and
        timestamp->year += 1;
// are we now on a leap
        leap =
!(timestamp->year % YEARS_IN_LEAP);
    }
    else
    {
        // set day of month to remaining
        timestamp->day = days; days = 0;
    }
    // set time values to remaining seconds
    timestamp->hour = hms / SECONDS_IN_HOUR;
    hms = hms % SECONDS_IN_HOUR;
    timestamp->minute = hms / SECONDS_IN_MINUTE;
    timestamp->second = hms % SECONDS_IN_MINUTE;
    return;
}

//
// copyOutZip
//
// Title           : Copy zip data out of class array
// Parameters      : char * - buffer to copy zip string into
//
// Return Value    : int - Length of copy
// Comments       :
//
inline int copyOutZip(char *buffer,char *value,int len = DEFAULT_ZIP_LEN)
{
    int index      = 0;
    int bufferPos  = 0;

    // add each digit of zip number to buffer inserting delimiter at 5
    while(value[index] && bufferPos < len)
    {
        if(index == 5)
            buffer[bufferPos++] = ZIP_DELIMITER;

        buffer[bufferPos++] = value[index++];
    }

    // space fill to the required length
    while(bufferPos < len)
        buffer[bufferPos++] = CHAR_FILL;

    buffer[bufferPos] = NULL;
    return len;
}

//
// copyInMoney64
//
// Title           : Copy money data into class array
// Parameters      : const char * - value string
// Return Value    : INT64b integer value
// Comments       :
//
inline bool copyInMoney64(const char * value,INT64b *number)
{
    //INT64b  number           = 0;
    int      index           = 0;
    int      decimal         = 0;
    int      decimals        = 0;
    int      digitsAfterDec  = 0;

    bool     negativeFlag    = false;

    // convert each digit to a numeric portion
    while(value[index])
    {
        // handle $ . - All the rest assumed numeric
        switch(value[index])
        {
            case MONEY_SYMBOL:

```



```

                // ignore $ sign
                break;
case NEGATIVE_SYMBOL:
    // set negative flag
    negativeFlag = true;
    break;

case DECIMAL_SYMBOL:
    // set decimal
    decimal=1;
    decimals++;
    if(decimals>1)
        //more than 1 decimal point found
        return false;
    break;

default:
    // adjust decimal places
    decimal = decimal * 10;

    // add digit to running total
    if(value[index] >= '0' && value[index] <= '9')
    {
        if(decimal)
            if(++digitsAfterDec >
2)
                return false;

        *number = (*number * 10) +
(value[index] - '0');
    }
    else
    {
        //non-numeric field inserted
        return false;
    }
    index++;
}

// apply decimal where decimal not found
if(decimal < 100)
{
    if(decimal)
    {
        *number *= (100 / decimal);
    }
    else
    {
        *number *= 100;
    }
}

// make negative
if(negativeFlag)
    *number = *number * (-1);

return true;
}

//
// copyInMoney
//
// Title           : Convert char string money field to double
// Parameters      : const char * - value string
// Return Value    : double integer value
// Comments       :

```

```

//
inline int copyInMoney(const char *value)
{
    char buf[20];
    int i,j,decimalFound,digitsAfterDecimal=0;

    int decimal=0;

    //walk past $ if present in char string
    if(*value == '$')
        *value++;

    int len=(int)strlen(value);
    for (i=0;i<len;i++)
    {
        if(value[i] == '.')
        {
            decimalFound++;
            if(decimalFound > 1)
                return -1;
        }
        if(value[i] == '-')

        if (value[i] != '.')
        {
            if(decimal)
            {
                if(digitsAfterDecimal<2)
                    digitsAfterDecimal++;
                else
                    return -1;
            }
            buf[j++] = value[i];
        }
    }
    int amount = atoi(buf);

    return amount;
}

//
// copyOutMoney64
//
// Title           : Copy money data out of class array
// Parameters      : char * - buffer to copy string 64 bit money into
//                  INT64b - value
//                  unsigned len - max number of
bytes to copy
// Return Value    : int - Length of copy
// Comments       :
//
inline void copyOutMoney64(char *buffer,INT64b value,unsigned int len =
DEFAULT_MONEY64_LEN)
{
    unsigned int    index                = len;

    int              places

= 0;

    bool             negativeFlag        = false;
    bool             moneyFlag           = true;

    // NULL terminate string
    buffer[index] = NULL;

```

```

// check length > 0
if(!index) return len;

// handle negative value
if(value < 0)
{
    negativeFlag = true;
    value = value * (-1);
}

// break off each digit from value, fill if needed
do
{
    if(value)
    {
        // get next digit and add to buffer
        buffer[--index] = (char) (value % 10 + '0');
        value /= 10; places++;

        if(places == 2 && index)
        {
            places++;
            buffer[--index] =
DECIMAL_SYMBOL;
        }
        else
        {
            // add zeros to first place before decimal point
            if(places < 2 || places == 3)
            {
                buffer[--index] =
ZERO_SYMBOL;
            }
            else
            {
                // add the decimal point
                if(places == 2)
                {
                    buffer[--index] =
DECIMAL_SYMBOL;
                }
                else
                {
                    // add the negative
                    if(negativeFlag)
                    {
                        negativeFlag
= false;
buffer[--index] = NEGATIVE_SYMBOL;
                    }
                    else
                    {
                        // add the
money indicator
if(moneyFlag)
                    {
                        moneyFlag = false;
buffer[--index] = MONEY_SYMBOL;
                    }
                }
            }
        }
    }
}

```

```

else
buffer[--index] = NUMERIC_FILL;
}
}
}
} while(index);
//return len;
}

//
// copyOutDateTime
// Title : Copy date & time data out of class array
// Parameters : char * - buffer to copy date & time string into
// INT64b - value
// Return Value : int - Length of copy
// Comments : Fixed length
//

inline int copyOutDateTime(char *buffer,INT64b value)
{
    datetime timestamp;

    // break value into time/date components
    calcOutDateTime(value,&timestamp);

    // put month into buffer
    *buffer++ = (char) ((timestamp.month / 10) + '0');
    *buffer++ = (char) ((timestamp.month % 10) + '0');
    *buffer++ = DATE_DELIMITER;

    // put day into buffer
    *buffer++ = (char) ((timestamp.day / 10) + '0');
    *buffer++ = (char) ((timestamp.day % 10) + '0');
    *buffer++ = DATE_DELIMITER;

    // put year into buffer
    int year = timestamp.year;
    *buffer++ = (char) ((year / 1000) + '0');
    year = year% 1000;
    *buffer++ = (char) ((year / 100) + '0'); year = year
% 100;
    *buffer++ = (char) ((year / 10) + '0');
    *buffer++ = (char) ((year % 10) + '0');
    *buffer++ = CHAR_FILL;

    // put hour into buffer
    *buffer++ = (char) ((timestamp.hour / 10) +
'0');
    *buffer++ = (char) ((timestamp.hour % 10) +
'0');
    *buffer++ = TIME_DELIMITER;

    // put minute into buffer
    *buffer++ = (char) ((timestamp.minute / 10) +
'0');
    *buffer++ = (char) ((timestamp.minute % 10) +
'0');
    *buffer++ = TIME_DELIMITER;

    // put second into buffer

```

```

    *buffer++ = (char)((timestamp.second / 10) +
'0');
    *buffer++ = (char)((timestamp.second % 10) +
'0');

    *buffer = NULL; return DEFAULT_DATETIME_LEN;
}
//
// copyOutTime
//
// Title : Copy date data out of class array
// Parameters : char * - buffer to copy date string into
// INT64b - value
// Return Value : int - Length of copy
// Comments : Fixed length
//
inline int copyOutDate(char *buffer,INT64b value)
{
    datetime timestamp;

    // break value into time/date components
    calcOutDateTime(value,&timestamp);

    // put month into buffer
    *buffer++ = (char)((timestamp.month / 10) + '0');
    *buffer++ = (char)((timestamp.month % 10) + '0');
    *buffer++ = DATE_DELIMITER;

    // put day into buffer
    *buffer++ = (char)((timestamp.day / 10) + '0');
    *buffer++ = (char)((timestamp.day % 10) + '0');
    *buffer++ = DATE_DELIMITER;

    // put year into buffer
    int year = timestamp.year;
    *buffer++ = (char)((year / 1000) + '0'); year = year % 1000;
    *buffer++ = (char)((year / 100) + '0'); year = year % 100;
    *buffer++ = (char)((year / 10) + '0');
    *buffer++ = (char)((year % 10) + '0');
    *buffer++ = CHAR_FILL;

    *buffer = NULL;

    return DEFAULT_DATE_LEN;
}
//
// copyOutTime
//
// Title : Copy time data out of class array
// Parameters : char * - buffer to copy time string into
// INT64b - value
// Return Value : int - Length of copy
// Comments : Fixed length TBD
//
inline int copyOutTime(char *buffer,INT64b value)
{
    datetime timestamp;

    // break value into time/date components
    calcOutDateTime(value,&timestamp);

    // put hour into buffer
    *buffer++ = (char)((timestamp.hour / 10) + '0');
    *buffer++ = (char)((timestamp.hour % 10) + '0');
    *buffer++ = TIME_DELIMITER;

    // put minute into buffer
    *buffer++ = (char)((timestamp.minute / 10) + '0');
    *buffer++ = (char)((timestamp.minute % 10) + '0');
    *buffer++ = TIME_DELIMITER;

    // put second into buffer
    *buffer++ = (char)((timestamp.second / 10) + '0');
    *buffer++ = (char)((timestamp.second % 10) + '0');

    *buffer = NULL; return DEFAULT_TIME_LEN;
}
//
// copyOutDecimal64
//
// Title : Copy decimal data out of class array
// Parameters : char * - buffer to copy string 64 bit money into
// INT64b - value
// unsigned len - max number of
bytes to copy
// Return Value : int - Length of copy
// Comments :
//
inline int copyOutDecimal64(char *buffer,INT64b value,unsigned int len =
DEFAULT_DECIMAL64_LEN)
{
    unsigned int index = len;
    int places
= 0;
    bool negativeFlag = false;

    // NULL terminate string
    buffer[index] = NULL;

    // check length > 0
    if(!index) return len;

    // handle negative value
    if(value < 0)
    {
        negativeFlag = true;
        value = value * (-1);
    }

    // break off each digit from value, fill if needed
    do
    {
        if(value)
        {
            // get next digit and add to buffer
            buffer[--index] = (char)(value % 10 + '0');
            value /= 10; places++;

            if(places == 2 && index)
            {
                places++;
                buffer[--index] =
DECIMAL_SYMBOL;
            }
        }
        else
        {
            // add zeros to first place before decimal point
            on (i.e. 0.00)

```

```

        if(places < 2 || places == 3)
        {
            ZERO_SYMBOL;
            buffer[--index] =
        }
        else
        {
            // add the decimal point
            if(places == 2)
            {
                DECIMAL_SYMBOL;
                buffer[--index] =
            }
            else
            {
                // add the negative
                indicator
                if(negativeFlag)
                {
                    = false;
                    negativeFlag
                    buffer[--index] = NEGATIVE_SYMBOL;
                }
                else buffer[--index] =
                NUMERIC_FILL;
            }
        }
        // need to trace place for decimal point and
        zero fill
        places++;
    } while(index);
    return len;
}

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
// Macros
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
using namespace std;

#ifdef _DEBUG
    int debugFlag = 1;
#else
    int debugFlag = 0;
#endif

inline BYTE8b *debugFileName(BYTE8b *filePath)
{
    BYTE8b *fileName = filePath + strlen(filePath);

    while(fileName != filePath)
    {
        if(*fileName == '/' || *fileName == '\\' && *(fileName +
1))
            return (fileName + 1);

        fileName--;
    }

    return filePath;
}

#define DEBUGADDRESS(POINTER)    hex << (void *) POINTER << dec

```

```

#define ERRORMSG(TEXT)
\
EnterCriticalSection(&errorMutex);
\
\
errorStream
\
<< debugFileName(__FILE__)
\
<< "|" <<
\
__TIMESTAMP__ << "|" << __LINE__ << "|"
\
<< _getpid()
\
<< "|" << GetCurrentThreadId() << "|"
\
<< TEXT;

errorStream.flush();
\
LeaveCriticalSection(&errorMutex);

#ifdef _DEBUG
    #define DEBUGMSG(TEXT)
\
EnterCriticalSection(&debugMutex);
\
\
debugStream << debugFileName(__FILE__)
\
<< "|" <<
\
__TIMESTAMP__ << "|" << __LINE__ << "|"
\
<< _getpid()
\
<< "|" << GetCurrentThreadId() << "|"
\
<< TEXT;

debugStream.flush();
\
LeaveCriticalSection(&debugMutex);

#define DEBUGSTRING(TEXT,LENGTH)
\
    debugVarString(TEXT,LENGTH)
\
#else
    #define DEBUGMSG(TEXT);
    #define DEBUGSTRING(TEXT,LENGTH);
\
#endif
#endif /* _COMMON_TPCC */

```

Tpcclsapi.cpp

```

/*
*****
** Project      : AIX
** Component    : Performance/TPC-C Benchmark
** Name         : tpccsapi.cpp
** Title        : TPCC html processing
*****
** Copyright (c) 2003 IBM Corporation
** All rights reserved
*****
** History      :
**      Developed at IBM Austin by the AIX RS/6000
**      performance group.
**
** Comments     :
**
*****
*/

#include "stdafx.h"

#include "..\tpccCom\tpccCom.h"
#include "..\tpccCom\tpccCom_i.c"
#include <tpccsapi.hpp>

// For custom assert and trace handling with WebDbg.exe
[ module(name="tpccsapi", type="dll") ];
[ emitidl(restricted) ];

#define _WIN32_DCOM

////////////////////////////////////
// Globals
////////////////////////////////////

int          maxDataSize;
//max struct size of all txn(s)
int          numUsers;
//number of users that client will service.
int          dlvyQueueLen;
//static length of dlvy queue
int          dlvyThreads;
//number of dlvy threads to create
int          dlvyBufferFreeSlots;          //length of dlvy txn
queue
int          dlvyBufferSlotIndex;          //index into next
available slot in dlvy txn queue
int          dlvyBufferThreadIndex;        //thread
index into dlvy txn queue
int          nullDB;
//null db on client(bypass com call).

int          trace;

static DWORD          threadLSIndex;
//isapi thread local storage index
CRITICAL_SECTION     isapiLock;
//isapi lock
CRITICAL_SECTION     errorLock;
//error log file lock.
CRITICAL_SECTION     termLock;
//terminal array lock.
CRITICAL_SECTION     dlvyQueueLock;
//dlvy queue critical section lock
HANDLE                dlvyThreadDone =
INVALID_HANDLE_VALUE; //dlvy thread exit event
HANDLE                dlvyThreadSemaphore
= INVALID_HANDLE_VALUE; //dlvy thread wrk to do semaphore

```

```

int
dlvyThreadID = 0;

struct DLVYQUEUEDATA    *dlvyQueue;
//dlvy queue
HANDLE                  *dlvyThreadHandles;
//ptr to array of thread handles

TERM_ENTRY              *termArray;
//array of terminal entries to store each users info.
int                    termNextFree;
//next available slot in terminal array

FILE                    *htmlDebug          = NULL;
//html debug file
FILE                    *errorLog          = NULL;
//error file
FILE                    *htmlTrace        = NULL;

ofstream debugStream;
ofstream errorStream;
CRITICAL_SECTION debugMutex;
CRITICAL_SECTION errorMutex;

char                    dlvyLogPath[128]    = {NULL};
char                    errorLogFile[128]   = {NULL};
char                    htmlTraceLogFile[128] = {NULL};
char                    dbName[64]          = {NULL};
char                    dbType[16]         = {NULL};

typedef INT (*CONNECT_PTR)(char *dbName,void **connectHandle);
typedef INT (*DISCONNECT_PTR)(void *connectHandle);
typedef INT (*DLVY_FUNC_PTR)(dlvy_wrapper *dlvy,void
*connectHandle);
typedef INT (*NORD_FUNC_PTR)(nord_wrapper *nord,void
*connectHandle);
typedef INT (*PYMT_FUNC_PTR)(paym_wrapper *pymt,void
*connectHandle);
typedef INT (*ORDS_FUNC_PTR)(ords_wrapper *ords,void *connectHandle);
typedef INT (*STOK_FUNC_PTR)(stok_wrapper *stok,void *connectHandle);

HINSTANCE                dbInstance;
CONNECT_PTR              db_connect;
DISCONNECT_PTR           db_disconnect;
DLVY_FUNC_PTR            dlvyCall;

////////////////////////////////////
// Page functions arrays
////////////////////////////////////

typedef int (*pageFuncPtr) (htmlPhraser *commandBlock, TXN_HANDLE
*txnHandle);

pageFuncPtr htmlPageFunctions[MAX_TRANSACTIONS] =
{
    {doLoginForm},
    {doNewOrderForm},
    {doPaymentForm},
    {doOrderStatusForm},
    {doDeliveryForm},
    {doStockForm},
    {doExit},
    {doLoginResults},
    {doNewOrderResults},
    {doPaymentResults},
    {doOrderStatusResults},

```

```

        {doDeliveryResults},
        {doStockResults}
    };

extern "C" DWORD WINAPI
HttpExtensionProc(LPEXTENSION_CONTROL_BLOCK lpECB)
{
    struct TXN_HANDLE *txnHandle = NULL;

    txnHandle = (TXN_HANDLE *) TlsGetValue(threadLSIndex);

    if(txnHandle == NULL)
    {
        int rc = initTxnHandle(&txnHandle);
        if (rc != OK)
        {
            char response[256]; char htmlHeader[256];
            sprintf(response,"ERROR: Init txnHandle
function failed.\n");

            size_t htmlPageLen = strlen(response);

            //add content length and keep alive header
            sprintf(htmlHeader,HEADER,htmlPageLen);

            lpECB->ServerSupportFunction(lpECB->ConnID,HSE_REQ_SEND_RESPON
NSE_HEADER,"200 OK",NULL,(DWORD*)htmlHeader);

            lpECB->WriteClient(lpECB->ConnID,response,(LPDWORD)&htmlPageLen,0
);

            return
            HSE_STATUS_SUCCESS_AND_KEEP_CONN;
        }

        txnHandle = (TXN_HANDLE *)
TlsGetValue(threadLSIndex);
        if (txnHandle == NULL)
        {
            char response[256]; char htmlHeader[256];
            sprintf(response,"ERROR: Unable to retrieve
txnHandle from TLS.\n");

            size_t htmlPageLen = strlen(response);

            //add content length and keep alive header
            sprintf(htmlHeader,HEADER,htmlPageLen);

            lpECB->ServerSupportFunction(lpECB->ConnID,HSE_REQ_SEND_RESPON
NSE_HEADER,"200 OK",NULL,(DWORD*)htmlHeader);

            lpECB->WriteClient(lpECB->ConnID,response,(LPDWORD)&htmlPageLen,0
);

            return
            HSE_STATUS_SUCCESS_AND_KEEP_CONN;
        }

        try
        {
            txnHandle->urlString =
(char*)lpECB->lpszQueryString;

            DEBUGMSG("calling doHtml() w/ query string:" <<
txnHandle->urlString << endl);
            doHtml(txnHandle);

```

```

        size_t htmlPageLen;
        htmlPageLen = strlen(txnHandle->htmlPage);
        if(htmlPageLen >= 4096)
        {
            ERRORMSG("WARNING: HTML PAGE IS
>= 4096!, page size:"<<htmlPageLen<<endl);
        }
        //add content length and keep alive header
        sprintf(txnHandle->htmlHeader,HEADER,htmlPageLen);
        size_t headerLen = strlen(txnHandle->htmlHeader);
        if(headerLen >= 256)
        {
            ERRORMSG("WARNING: HTML
HEADER IS >= 256!, header size:"<<headerLen<<endl);
        }

        //write response to user

        lpECB->ServerSupportFunction(lpECB->ConnID,HSE_REQ_SEND_RESPON
NSE_HEADER,"200 OK",NULL,(DWORD*)txnHandle->htmlHeader);

        lpECB->WriteClient(lpECB->ConnID,txnHandle->htmlPage,(LPDWORD)&ht
mlPageLen,0);

        DEBUGMSG("HTML
PAGE-->"<<endl<<txnHandle->htmlHeader<<txnHandle->htmlPage<<endl);

    }
    catch (...)
    {
        char response[256];
        ZeroMemory(response,256);
        char *ptr = response;

        appendText(&ptr,"<HTML><BODY> Error : Unhandled
Exception </BODY><</HTML>");
        DWORD cbResponse = sizeof(response)-1 ;

        //write response to user

        lpECB->ServerSupportFunction(lpECB->ConnID,HSE_REQ_SEND_RESPON
NSE_HEADER,"200 OK",NULL,(DWORD*)response);

        lpECB->WriteClient(lpECB->ConnID,response,&cbResponse,0);
    }

    return HSE_STATUS_SUCCESS_AND_KEEP_CONN;
}

extern "C" BOOL WINAPI GetExtensionVersion(HSE_VERSION_INFO*
pVer)
{
    // Create the extension version string, and copy string to
HSE_VERSION_INFO structure.
    pVer->dwExtensionVersion =
MAKELONG(HSE_VERSION_MINOR, HSE_VERSION_MAJOR);

    // Copy description string into HSE_VERSION_INFO structure.
    strcpy(pVer->lpszExtensionDesc, "TPCC ISAPI Extension");

    // Initialize isapi critical section
    InitializeCriticalSection(&isapiLock);

    // Initialize error log critical section
    InitializeCriticalSection(&errorLock);

    // Initialize terminal critical section
    InitializeCriticalSection(&termLock);

```

```

// Initialize debug/error critical sections
if(debugFlag)
    InitializeCriticalSection(&debugMutex);
InitializeCriticalSection(&errorMutex);

// Read registry values
if(readRegistryValues() != OK)
    return(FALSE);

// Initialize terminal array
termArray = (TERM_ENTRY*)
calloc(numUsers,sizeof(TERM_ENTRY));
termNextFree = 1;

//open up error/debug streams
errorStream.rdbuf( )->open(errorLogFile,ios::out);
if(debugFlag)
    debugStream.rdbuf( )->open(htmlTraceLogFile,ios::out);

ERRORMSG("Error log file open."<<endl);

DEBUGMSG("Loading library for dlvy txn."<<endl);
int rc = getDBInstance();
if (rc != OK)
{
    ERRORMSG("Error, unable to load database dll,
rc:<<rc);
    DEBUGMSG("Error, unable to load database dll,
rc:<<rc);
    return FALSE;
}
DEBUGMSG("Library loaded for dlvy txn."<<endl);

DEBUGMSG("Calling initDlvy." <<endl);

if(initDlvy() != OK)
    return (FALSE);

DEBUGMSG("Initializing TLS." << endl);

// Initialize thread local storage index
threadLSIndex = TlsAlloc();
if (threadLSIndex == TLS_NULL)
{
    ERRORMSG("Isapi error: unable to initialize thread
local storage(TLS), rc:" << GetLastError()<<endl);
    return(FALSE);
}

DEBUGMSG("sizeof out_neword_struct: "<<sizeof(struct
out_neword_struct)<<endl);
DEBUGMSG("sizeof in_neword_struct: "<<sizeof(struct
in_neword_struct)<<endl);
DEBUGMSG("sizeof out_payment_struct: "<<sizeof(struct
out_payment_struct)<<endl);
DEBUGMSG("sizeof in_payment_struct: "<<sizeof(struct
in_payment_struct)<<endl);
DEBUGMSG("sizeof out_ordstat_struct: "<<sizeof(struct
out_ordstat_struct)<<endl);
DEBUGMSG("sizeof in_ordstat_struct: "<<sizeof(struct
in_ordstat_struct)<<endl);
DEBUGMSG("sizeof out_delivery_struct: "<<sizeof(struct
out_delivery_struct)<<endl);
DEBUGMSG("sizeof in_delivery_struct: "<<sizeof(struct
in_delivery_struct)<<endl);

```

```

DEBUGMSG("sizeof out_stocklev_struct: "<<sizeof(struct
out_stocklev_struct)<<endl);
DEBUGMSG("sizeof in_stocklev_struct: "<<sizeof(struct
in_stocklev_struct)<<endl);

//compute the max struct size for com data construct
maxDataSize = max(maxDataSize,sizeof(nord_wrapper));
maxDataSize = max(maxDataSize,sizeof(paym_wrapper));
maxDataSize = max(maxDataSize,sizeof(ords_wrapper));
maxDataSize = max(maxDataSize,sizeof(dlvy_wrapper));
maxDataSize = max(maxDataSize,sizeof(stok_wrapper));
maxDataSize += 10;

DEBUGMSG("max data struct size:"<<maxDataSize <<endl);

return true;
}

extern "C" BOOL WINAPI TerminateExtension(DWORD dwFlags)
{
    return true;
}

/*
*****
** Name          :          initTxnHandle
** Description   :
**              :          Isapi thread initializes
**              :          its own com interface
**              :          structure.
** Parameters    :
**              :          TXN_HANDLE**
isapi txn handle
** Returns      :
**              :          int - return code
** Comments     :
**
*****
*/
int initTxnHandle(TXN_HANDLE **txnHandle)
{
    DEBUGMSG("Inside init txn handle, getting isapiLock." << endl);
    EnterCriticalSection(&isapiLock);

    HRESULT hres = NULL;
    try
    {
        DEBUGMSG("Got ispaiLock, initializing txnHandle:
"<<DEBUGADDRESS(*txnHandle)<<endl);
        *txnHandle = (TXN_HANDLE *)
calloc(1,sizeof(TXN_HANDLE));
        if (*txnHandle == NULL)
        {
            ERRORMSG("Unable to allocated
TXN_HANDLE, rc:"<<GetLastError()<<endl);
            return ERR;
        };

        (*txnHandle)->comInterface.comHandle = NULL;
        DEBUGMSG("Initializing txnHandle com data buffer to
"<<maxDataSize<<"bytes"<<endl);
        (*txnHandle)->comInterface.txnBuffer = (char *)
CoTaskMemAlloc(maxDataSize);
        if (!((*txnHandle)->comInterface.txnBuffer))
        {
            ERRORMSG("CoTaskMemAlloc() failed of
size "<<maxDataSize<<"; rc: "<<hres<<endl);

```

```

        return(ERR);
    };
    DEBUGMSG("txnHandle com data buffer initialized to "
<< maxDataSize << "bytes" <<endl);

    DEBUGMSG("Calling CoInitialize with txnHandle:
"<<DEBUGADDRESS(*txnHandle)<<endl);
    hres =
CoInitializeEx(NULL,COINIT_MULTITHREADED);
    if (FAILED(hres))
    {
        ERRORMSG("CoInitializeEx() failed, rc :
"<<hres<<endl);
        return(ERR);
    };

    struct _timeb
startTime;
    struct _timeb
endTime;

    DEBUGMSG("Calling CoCreateInstance with
txnHandle:"<<DEBUGADDRESS(*txnHandle)<< endl);
    _ftime(&startTime);
    hres =
CoCreateInstance(CLSID_tpcc_com,NULL,CLSCTX_SERVER,IID_Itpcc_co
m,(void **)&(*txnHandle)->comInterface.comHandle);
    if (FAILED(hres))
    {
        _ftime(&endTime);
        //store error code in txnHandle
        ERRORMSG("CoCreateInstance() failed,
code:"<<HRESULT_CODE(hres)<<"
facility:"<<HRESULT_FACILITY(hres)<<"
        " hres:"<<hres<< " time
waiting:"<<
        (((endTime.time -
startTime.time)*1000)+
        (endTime.millitm -
startTime.millitm))/1000.0)<<endl);

        DEBUGMSG("CoCreateInstance() failed,
code:"<<HRESULT_CODE(hres)<<"
facility:"<<HRESULT_FACILITY(hres)<<"
        " hres:"<<hres<< " time
waiting:"<<
        (((endTime.time -
startTime.time)*1000)+
        (endTime.millitm -
startTime.millitm))/1000.0)<<endl);

        return(ERR);
    };

    _ftime(&endTime);
    DEBUGMSG("CoCreateInstance successful.txnHande
com initialized, time waiting for object to be activated:" <<
        (((endTime.time - startTime.time)*1000)+
startTime.millitm))/1000.0)<<endl);

    //call set complete to return object to pool.
(*txnHandle)->comInterface.comHandle->doSetComplete();

    //set the com buffers size

```

```

        DEBUGMSG("Setting txnHandle: " <<
DEBUGADDRESS(*txnHandle) << "com buffer size to " << maxDataSize<<
endl)
        (*txnHandle)->comInterface.size = maxDataSize;

        DEBUGMSG("txnHandle:
"<<DEBUGADDRESS(*txnHandle) <<"set to " << maxDataSize << endl);

        TlsSetValue(threadLSIndex,*txnHandle);

        DEBUGMSG("txnHandle:
"<<DEBUGADDRESS(*txnHandle) <<"stored in TLS" << endl);

ZeroMemory((*txnHandle)->htmlPage,MAX_HTML_PAGE_LEN);

ZeroMemory((*txnHandle)->htmlHeader,MAX_HTML_HEADER_LEN);

        LeaveCriticalSection(&isapiLock);
        return(OK);
    }
    catch(...)
    {
        DEBUGMSG("Unhandled exception in initTxnHandle,
unlocking isapi lock" <<endl);
        ERRORMSG("Unhandled exception in initTxnHandle,
unlocking isapi lock" <<endl);
        LeaveCriticalSection(&isapiLock);
    };

    return ERR;
}

/*
*****
** Name          :          getDBInstance
** Description   :          load db specific lib
**              :          based on dbType registry
**              :          value.
** Parameters    :
** Returns       :          int - return code
** Comments      :          This function only
**              :          exists for the dlvy threads
**              :          DlvY threads hold
**              :          direct connections to the database
**              :          and therefore need to
**              :          know what db interface to talk to.
*****
*/
int getDBInstance()
{
    if(nullDB)
    {
        dbInstance =
LoadLibrary("c:\\inetpub\\wwwroot\\tpcc\\nullDB.dll");
        if(dbInstance == NULL)
        {
            return ERR_NULL_DLL_NOT_LOADED;
        }
    }
    else if( (strcmp(dbType,"DB2") == 0) )
    {
        dbInstance =
LoadLibrary("c:\\inetpub\\wwwroot\\tpcc\\tpccDB2glue.dll");

```



```

        if(dbInstance == NULL)
        {
            return ERR_DB2_DLL_NOT_LOADED;
        }
    }
    else if( (strcmp(dbType,"ORACLE") == 0) )
    {
        return ERR_ORACLE_DLL_NOT_LOADED;
    }
    else
    {
        return ERR_UNKNOWN_DB;
    }

    db_connect =
(CONNECT_PTR)GetProcAddress(dbInstance,"connect_db");
    if(db_connect == NULL)
    {
        return ERR_CONNECT_ADDRESS_NOT_FOUND;
    }
    dlvyCall =
(DLVY_FUNC_PTR)GetProcAddress(dbInstance,"do_dlvy");
    if(dlvyCall == NULL)
    {
        return ERR_DLVY_ADDRESS_NOT_FOUND;
    }

    return OK;
}

```

```

/*
*****
** Name          :          initDlvy
** Description   :
**              initialize dlvy
** Parameters    :
**
** Returns      :
**              int - return code
** Comments     :
**
*****
*/

```

```

int initDlvy()
{
    // Initialize critical section
    InitializeCriticalSection(&dlvyQueueLock);

    //create dlvy queue
    dlvyQueue = (DLVYQUEUEDATA *)
calloc(dlvyQueueLen,sizeof(DLVYQUEUEDATA));

    dlvyThreadDone = CreateEvent(NULL,

TRUE,          //manual reset

FALSE, //initially not signalled.

NULL);
    if(dlvyThreadDone == NULL)
    {
        DEBUGMSG("Error: dlvyThreadDone handled init
failed, GetLastError:"<<GetLastError()<<endl);

        ERRORMSG("Error : dlvyThreadDone handled init
failed, GetLastError:"<<GetLastError()<<endl);
    }
}

```

```

        return ERR_DLVY_EVENT_INIT_FAILED;
    }

    //create dlvy semaphore
    dlvyThreadSemaphore =
CreateSemaphore(NULL,0,dlvyQueueLen,NULL);
    if(dlvyThreadSemaphore == NULL)
    {
        DEBUGMSG("Error: dlvyThreadSemaphore semaphore
init failed, GetLastError:"<<GetLastError()<<endl);
        ERRORMSG("Error: dlvyThreadSemaphore semaphore
init failed, GetLastError:"<<GetLastError()<<endl);
        return ERR_DLVY_SEMAPHORE_INIT_FAILED;
    }

    //set number of free slots available in queue
    dlvyBufferFreeSlots = dlvyQueueLen;

    //index into next available slot in dlvy txn queue
    dlvyBufferSlotIndex = 0;

    //thread index into dlvy txn queue
    dlvyBufferThreadIndex = 0;

    dlvyThreadHandles = new HANDLE[dlvyThreads];
    //create threads
    for(int threadCount = 0;threadCount < dlvyThreads;threadCount++)
    {
        dlvyThreadHandles[threadCount] =
(HANDLE)_beginthread(dlvyThreadEntry,0,NULL);
        if(dlvyThreadHandles[threadCount] ==
INVALID_HANDLE_VALUE)
            return ERR_DLVY_THREAD_FAILED;
    }

    return OK;
}

```

```

/*
*****
** Name          :          readRegistryValues
** Description   :
**              initialize isapi global
variables from registry
** Parameters    :
**
** Returns      :
**              int - return code
** Comments     :
**
*****
*/

```

```

int readRegistryValues()
{
    HKEY    registryKey;
    char    value[MAX_STRING_LEN];
    DWORD   regType;
    DWORD   regValue;
    DWORD   regValueSize = MAX_STRING_LEN;

    //open up registry key

    if(RegOpenKeyEx(HKEY_LOCAL_MACHINE,REGISTRY_SUB_KEY,0,K
EY_READ,&registryKey) != ERROR_SUCCESS)
        return ERR_UNABLE_TO_OPEN_REG;
}

```

```

//get null db flag
regValueSize = sizeof(regValue);
if(RegQueryValueEx(registryKey,NULL_DB,0,&regType,(BYTE
*)&regValue,&regValueSize) == ERROR_SUCCESS)
    nullDB = regValue;
else
    nullDB = 0;

//get num dlvy threads
regValueSize = sizeof(regValue);

if(RegQueryValueEx(registryKey,DELIVERY_THREADS,0,&regType,(BYT
E *)&regValue,&regValueSize) == ERROR_SUCCESS)
    dlvyThreads = regValue;
else
    dlvyThreads =
DEFAULT_DLVY_THREADS;

//get dlvy queue len
regValueSize = sizeof(regValue);

if(RegQueryValueEx(registryKey,DELIVERY_QUEUE_LEN,0,&regType,(B
YTE *)&regValue,&regValueSize) == ERROR_SUCCESS)
    dlvyQueueLen = regValue;
else
    dlvyQueueLen =
DEFAULT_DLVY_QUEUE_LEN;

//get the htmlTrace flag
regValueSize = sizeof(regValue);

if(RegQueryValueEx(registryKey,HTML_TRACE,0,&regType,(BYTE
*)&regValue,&regValueSize) == ERROR_SUCCESS)
    trace = regValue;
else
    trace = 0;

//get the client null db flag
regValueSize = sizeof(regValue);
if(RegQueryValueEx(registryKey,NULL_DB,0,&regType,(BYTE
*)&regValue,&regValueSize) == ERROR_SUCCESS)
    nullDB = regValue;
else
    nullDB = 0;

//get the num of users
regValueSize = sizeof(regValue);

if(RegQueryValueEx(registryKey,NUM_USERS,0,&regType,(BYTE
*)&regValue,&regValueSize) == ERROR_SUCCESS)
    numUsers = regValue;
else
    numUsers = DEFAULT_NUM_USERS;

//get dlvy log file path
regValueSize = sizeof(value);
if
(RegQueryValueEx(registryKey,DELIVERY_LOG_PATH,0,&regType,(BYT
E *)&value,&regValueSize)== ERROR_SUCCESS )
    strcpy(dlvyLogPath,value);
else
    strcpy(dlvyLogPath,DEFAULT_DLVY_LOG_PATH);

//get global error log file path/name
regValueSize = sizeof(value);

```

```

if
(RegQueryValueEx(registryKey,ERROR_LOG_FILE,0,&regType,(BYTE *)
&value,&regValueSize)== ERROR_SUCCESS )
    strcpy(errorLogFile,value);
else
    strcpy(errorLogFile,DEFAULT_ERROR_LOG_FILE);

//get global error log file path/name
regValueSize = sizeof(value);
if
(RegQueryValueEx(registryKey,HTML_TRACE_LOG_FILE,0,&regType,(B
YTE *) &value,&regValueSize)== ERROR_SUCCESS )
    strcpy(htmlTraceLogFile,value);
else
    strcpy(htmlTraceLogFile,DEFAULT_HTML_TRACE_LOG_FILE);

//get db name
regValueSize = sizeof(value);
if (RegQueryValueEx(registryKey,DB_NAME,0,&regType,(BYTE
*)&value,&regValueSize)== ERROR_SUCCESS )
    strcpy(dbName,value);
else
    strcpy(dbName,DEFAULT_DB_NAME);

//get db type
regValueSize = sizeof(value);
if (RegQueryValueEx(registryKey,DB_TYPE,0,&regType,(BYTE
*)&value,&regValueSize)== ERROR_SUCCESS )
    strcpy(dbType,value);

RegCloseKey(registryKey);

return OK;
}
/*
*****
** Name          : doLoginForm
** Description   :
**               HTML Login page entry point
** Parameters    :
**               htmlPhraser*   command
block
**               TXN_HANDLE*   txn handle
struct
** Returns      :
**               int - return code
** Comments     :
**
*****
*/
int doLoginForm(htmlPhraser *commandBlock, TXN_HANDLE *txnHandle)
{
    DEBUGMSG("Entering doLoginForm()."<<endl);
    char *html=txnHandle->htmlPage;

    DEBUGMSG("Creating html login page"<<endl);
    //begin html page
    appendText(&html,"<HTML><HEAD><TITLE>TPC-C Client
Home Page</TITLE></HEAD>"
                "<FORM
ACTION=\"\"
                APP_NAME
                \"\"
METHOD=\"GET\">"

```

```

Login.</H2>"
TYPE="hidden" NAME=""

<INPUT NAME=""

NAME=""

TYPE="submit" VALUE="Submit">"

html+=sprintf(html,"dlvy Queue Length:%d <BR> num dlvy threads:%d <BR>
dlvy queue free slots:%d <BR> isapi queue index:%d <BR> thread queue
index:%d <BR> </BODY></HTML>\n",
                dlvyQueueLen,
                dlvyThreads,
                dlvyBufferFreeSlots,
                dlvyBufferSlotIndex,
                dlvyBufferThreadIndex);

DEBUGMSG("Html login page done"<<endl);

return OK;
}

/*
*****
** Name          : doLoginResults
** Description    :
**               HTML Login results page entry
point
** Parameters    :
**               htmlPhraser*   command
block
**               TXN_HANDLE*   txn handle
struct
** Returns      :
**               int - return code
** Comments     :
**
*****
*/

int doLoginResults(htmlPhraser *commandBlock, TXN_HANDLE *txnHandle)
{
    char *html=txnHandle->htmlPage;

    //validate parameters
    if( (txnHandle->w_id = atoi(commandBlock->get_W_ID())) == 0 )
    {
        doLoginErrorPage(html,ERR_INVALID_W_ID);
        return OK;
    }
    if( (txnHandle->d_id = atoi(commandBlock->get_D_ID())) == 0 )
    {
        doLoginErrorPage(html,ERR_INVALID_D_ID);
        return OK;
    }
}

```

```

"<H2>Please
" <INPUT
CMD_TXN_ID
"\ VALUE=""
CMD_MENU
">"
"<H3>Warehouse
CMD_W_ID
"\ SIZE=6"
" District <INPUT
CMD_D_ID
"\ SIZE=2></H3>"
"<INPUT
"</FORM>";

```

```

//store user into terminal array,
//function will ERR if the terminal array is full
if( assignTerminal(txnHandle) != OK)
{
    doLoginErrorPage(html,ERR_TERMINAL_FULL);
    return OK;
};

appendText(&html,"<HTML><HEAD><TITLE>TPC-C Main
Menu</TITLE></HEAD>\r\n"
ACTION=""
APP_NAME
""
METHOD="GET">\r\n"
"<H3>Please Select
Transaction.</H3>\r\n");
html+=appendButtons(html);
html+=appendHiddenFields(html,txnHandle);
appendText(&html,"</FORM></BODY></HTML>");

return OK;
}

/*
*****
** Name          : doLoginErrorPage
** Description    :
**               HTML Login page entry point
** Parameters    :
**               char *       html page
buffer
**               char *       error
message
** Returns      :
**               int - return code
** Comments     :
**
*****
*/

int doLoginErrorPage(char *htmlPage,char *errorMessage)
{
    char *html=htmlPage;

    //begin html page
    appendText(&html,"<HTML><HEAD><TITLE>TPC-C Client
Home Page</TITLE></HEAD>"
ACTION=""
APP_NAME
""
METHOD="GET">);
appendText(&html,"<H2>Please Login.</H2>"
TYPE="hidden" NAME=""
CMD_TXN_ID
"\ VALUE=""
CMD_MENU
">"
"<H3>Warehouse
<INPUT NAME=""
CMD_W_ID
"\ SIZE=6"
" District <INPUT
NAME=""
CMD_D_ID

```

```

                                        "\" SIZE=2></H3>"
                                        "<INPUT
TYPE=\"submit\" VALUE=\"Submit\">"
                                        "</FORM>");
        appendText(&html,errorMessage);
        appendText(&html,"<BODY></HTML>");

        return OK;
    }

/*
*****
** Name                : doNewOrderForm
** Description         :
**                    HTML neworder page entry point
** Parameters         :
**                    htmlPhraser*    command
block
**                    TXN_HANDLE*    txn handle
struct
** Returns            :
**                    int - return code
** Comments           :
**
*****
*/
int doNewOrderForm(htmlPhraser *commandBlock, TXN_HANDLE
*txnHandle)
{
    char *html=txnHandle->htmlPage;

    appendText(&html,"<HTML><HEAD><TITLE>TPC-C New
Order</TITLE></HEAD>\r\n"
                                "<BODY><FORM
ACTION=\"\"
                                APP_NAME
                                \"\"
METHOD=\"GET\">\r\n"
                                "
                                <CENTER><H3>Please Fill In New Order Form.</H3></CENTER>\r\n"
                                //check if not needed
                                "Submit Transaction
                                <INPUT TYPE=\"submit\" NAME=\"\"
                                CMD_TXN_ID
                                \"\" VALUE=\"\"
                                CMD_NORD
                                \"\">");

    //append the hidden
    html+=appendHiddenFields(html,txnHandle);

    //int buffer for warehouse
    char buffer[15];
    appendText(&html," <PRE>"
                                "    1    2    3
//    4    5    6    7    8    9\r\n"
//
"123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890\r\n"
                                "Warehouse: ");
    appendText(&html,itoa(txnHandle->w_id,buffer,10),7,1);
    appendText(&html,"District: <INPUT NAME=\"\"
                                CMD_D_ID
                                \"\" SIZE=1>
Date:<BR>"
                                "Customer <INPUT NAME=\"\"

```

```

                                CMD_C_ID
                                "\" SIZE=6> Name:
                                Credit: %Disc.:<BR>"
                                "Order Number:
Number of Lines:    W_tax:    D_tax:<BR><BR>"
//                "    1    2    3
//    4    5    6    7    8    9\r\n"
//
"123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890\r\n"
                                "Supp_W Item_Num
Item_Name            Qty Stock B/G Price    Amount <BR> ");
//append the 15 items commands
html+=appendItems(html,NORD_ITEMS,ITEM_START);

//seal up html page
appendText(&html,"</PRE></BODY></HTML>");

return OK;
}

/*
*****
** Name                : doNewOrderResults
** Description         :
**                    HTML neworder page entry point
** Parameters         :
**                    htmlPhraser*    command
block
**                    TXN_HANDLE*    txn handle
struct
** Returns            :
**                    int - return code
** Comments           :
**
*****
*/
int doNewOrderResults(htmlPhraser *commandBlock, TXN_HANDLE
*txnHandle)
{
    DEBUGMSG("Entered doNewOrderResults" << endl);

    char *html=txnHandle->htmlPage;
    struct nord_wrapper *nord = NULL;

    DEBUGMSG("Casting COM txnBuffer to nord struct" <<endl);
    nord = (nord_wrapper*)txnHandle->comInterface.txnBuffer;
    ZeroMemory(nord,maxDataSize);
    DEBUGMSG("COM txnBuffer initialized, validating input
parameters" << endl);

    //set warehouse,district and customer id from command block
    nord->in_nord.s_W_ID = txnHandle->w_id;
    DEBUGMSG("nord w_id:" << nord->in_nord.s_W_ID << endl);

    if( (nord->in_nord.s_D_ID = atoi(commandBlock->get_D_ID()) )
        == 0)
    {
        doNewOrderErrorPage(html,ERR_INVALID_D_ID,commandBlock,txnHandle
        );
        return OK;
    }
    DEBUGMSG("nord d_id:" << nord->in_nord.s_D_ID << endl);

```

```

0)      if(nord->in_nord.s_C_ID = atoi(commandBlock->get_C_ID())) ==
        {
doNewOrderErrorPage(html,ERR_INVALID_C_ID,commandBlock,txnHandle
);
        return OK;
        }
        DEBUGMSG("nord c_id:" << nord->in_nord.s_C_ID << endl);

        int itemCmd          = ITEM_START;
        short itemComplete = 0;
        char field[256] = {NULL};

        for (int itemIndex=0;itemIndex<NORD_ITEMS;itemIndex++)
        {
            //supply warehouse
            if( *(commandBlock->get_ITEM_SUPP_W(itemIndex))

                if(
(nord->in_nord.in_item[nord->in_nord.s_O_OL_CNT].s_OL_SUPPLY_W_ID
= atoi(commandBlock->get_ITEM_SUPP_W(itemIndex))) == 0)
                {
doNewOrderErrorPage(html,ERR_INVALID_SUPPLY_W_ID,commandBlock
,txnHandle);
                    return OK;
                }
                else
                    itemComplete++;

                //item number
                if(
*(commandBlock->get_ITEM_ITEM_NUM(itemIndex))
                {
                    if(itemComplete==1)
                    {
                        if(
(nord->in_nord.in_item[nord->in_nord.s_O_OL_CNT].s_OL_I_ID =
atoi(commandBlock->get_ITEM_ITEM_NUM(itemIndex))) == 0)
                        {
doNewOrderErrorPage(html,ERR_INVALID_ITEM_NUM,commandBlock,txn
Handle);
                            return OK;
                        }
                        else
                            itemComplete++;
                    }
                    //missing previous value of item supp
                    warehouse, flag error
                    else
                    {
doNewOrderErrorPage(html,ERR_INVALID_SUPPLY_W_ID,commandBlock
,txnHandle);
                            return OK;
                        }
                    }
                    else if( (itemComplete==1) //nothing in the command
                    block, check to see if the previous item value is present
                    {
doNewOrderErrorPage(html,ERR_INVALID_ITEM_NUM,commandBlock,txn
Handle);
                            return OK;
                        }
                    }

```

```

//item qty
if(*(commandBlock->get_ITEM_QTY(itemIndex)))
{
    if(itemComplete==2)
    {
        if(
(nord->in_nord.in_item[nord->in_nord.s_O_OL_CNT].s_OL_QUANTITY =
atoi(commandBlock->get_ITEM_QTY(itemIndex))) == 0)
        {
doNewOrderErrorPage(html,ERR_INVALID_ITEM_OTY,commandBlock,txn
Handle);
            return OK;
        }
        else
            itemComplete++;
    }
    //missing previous value of item number
    else if (itemComplete == 1)
    {
doNewOrderErrorPage(html,ERR_INVALID_ITEM_NUM,commandBlock,txn
Handle);
            return OK;
        }
        //missing 1st value of supp warehouse
        else
        {
doNewOrderErrorPage(html,ERR_INVALID_SUPPLY_W_ID,commandBlock
,txnHandle);
            return OK;
        }
    }
    else if(itemComplete==2) //nothing in
the command block, check to see if the previous item values are present
    {
doNewOrderErrorPage(html,ERR_INVALID_ITEM_NUM,commandBlock,txn
Handle);
            return OK;
        }
    }
    DEBUGMSG("nord item:" <<
nord->in_nord.s_O_OL_CNT << "SUPPLY_W_ID:" <<
nord->in_nord.in_item[nord->in_nord.s_O_OL_CNT].s_OL_SUPPLY_W_ID
<<
        " OL_I_ID:" <<
nord->in_nord.in_item[nord->in_nord.s_O_OL_CNT].s_OL_I_ID << "
OL_QUANTITY:" <<
nord->in_nord.in_item[nord->in_nord.s_O_OL_CNT].s_OL_QUANTITY
<<endl);

        if(itemComplete == 3)
            nord->in_nord.s_O_OL_CNT++;

        itemComplete=0;
    }
    DEBUGMSG("complete nord
items:"<<nord->in_nord.s_O_OL_CNT<<" initializing remainig unused items "
<< NORD_ITEMS - nord->in_nord.s_O_OL_CNT << " to 0" <<endl);
    for(int
itemIndex=nord->in_nord.s_O_OL_CNT;itemIndex<NORD_ITEMS;itemIndex
++)
    {
nord->in_nord.in_item[itemIndex].s_OL_SUPPLY_W_ID=0;

```

```

        nord->in_nord.in_item[itemIndex].s_OL_I_ID = 0;
        nord->in_nord.in_item[itemIndex].s_OL_QUANTITY
=0;
    }
    DEBUGMSG("nord creating new order results html title page"
<<endl);

    appendText(&html,"<HTML><HEAD><TITLE>TPC-C New Order
Results</TITLE></HEAD>\r\n"
ACTION="\\"
APP_NAME
""
METHOD="\GET">\r\n");
//append menu buttons
html+=appendButtons(html);
html+=appendHiddenFields(html,txnHandle);

    appendText(&html,"</FORM><CENTER><H3>New Order</H3>
<BR></CENTER>"
" <PRE>"
" 1 2 3
// 4 5 6 7 8 9\r\n"
//
"123456789012345678901234567890123456789012345678901234567890123
456789012345678901234567890\r\n
""");
//assume failure
nord->out_nord.s_transtatus = -1;

    DEBUGMSG("nord executing COM interface function" << endl);
    HRESULThres;
    try
    {
        hres =
txnHandle->comInterface.comHandle->doNewOrder(&txnHandle->comInterfa
ce.size,(UCHAR*)&txnHandle->comInterface.txnBuffer);
    }
    catch(...)
    {
        html+=sprintf(html,"ERROR: nord com call caused
exeception to occur.</PRE></BODY></HTML>");
        ERRORMSG("ERROR : nord com call cause exeception
to occur,"<<endl);
        return OK;
    }

    if(FAILED(hres))
    {
        ERRORMSG("ERROR : nord com call failed, rc:" <<
hex << hres);
        DEBUGMSG("ERROR : nord com call failed, rc:" <<
hex << hres);
        return OK;
    }

//com call successful, return object back to pool.
hres = txnHandle->comInterface.comHandle->doSetComplete();
if(FAILED(hres))
{
    ERRORMSG("ERROR : nord setcomplete call failed,
rc:" << hex << hres);
    DEBUGMSG("ERROR : nord setcomplete call failed,
rc:" << hex << hres);
}

    nord = (nord_wrapper *)txnHandle->comInterface.txnBuffer;

```

```

    if(FAILED(hres))
    {
        html+=sprintf(html,"ERROR: nord com doSetComplete
failed, rc:%ld</PRE></BODY></HTML>",hres);
        ERRORMSG("ERROR : nord com doSetComplete
failed, rc:"<<DEBUGADDRESS(hres)<<endl);
        return OK;
    }

    DEBUGMSG("nord COM interface function successful,
s_transtatus:" << nord->out_nord.s_transtatus << endl);

    int rc = nord->out_nord.s_transtatus;

    char buffer[10];
    appendText(&html,"Warehouse: ");
    appendText(&html,ittoa(nord->in_nord.s_W_ID,buffer,10),6,1);

    appendText(&html,"District: ");
    appendText(&html,ittoa(nord->in_nord.s_D_ID,buffer,10),26,1);

    appendText(&html,"Date: ");
    if(rc == OK)
    {
        char dateTimeBuffer[50];
copyOutDateTime(dateTimeBuffer,nord->out_nord.s_O_ENTRY_D_time);
        appendText(&html,dateTimeBuffer);
    }
    appendText(&html," <BR>"
"Customer: ");
    appendText(&html,ittoa(nord->in_nord.s_C_ID,buffer,10),8,1);
    appendText(&html,"Name: ");

appendText(&html,nord->out_nord.s_C_LAST,LAST_NAME_LEN+3,1);

    appendText(&html,"Credit: ");
    appendText(&html,nord->out_nord.s_C_CREDIT,5,1);

    appendText(&html,"%Disc: ");
    if(rc == OK)
    {
html+=sprintf(html,"%2.2lf",nord->out_nord.s_C_DISCOUNT/100.0);
    }
    appendText(&html," <BR>"
"Order Number: ");
    if(rc != INVALID_STATUS)

appendText(&html,ittoa(nord->out_nord.s_O_ID,buffer,10),10,1);

        appendText(&html,"Number of Lines: ");

        if(rc != INVALID_STATUS)

appendText(&html,ittoa(nord->out_nord.s_O_OL_CNT,buffer,10),10,1);

        appendText(&html,"W_Tax: ");
        if(rc == OK)
        {

html+=sprintf(html,"%5.2lf",nord->out_nord.s_W_TAX/100.0);
        }

        appendText(&html," D_Tax: ");
        if(rc == OK)
        {

```

```

html+=sprintf(html,"%5.2lf",nord->out_nord.s_D_TAX/100.0);
    }
    appendText(&html," <BR> <BR>"
//          "      1      2      3      4      5      6
//      7      8      9\r\n"
//
"123456789012345678901234567890123456789012345678901234567890123
456789012345678901234567890\r\n"
        " Supp_W Item_Id Item_Name
Qty Stock B/G Price Amount <BR> ");

//display items
if (rc == OK)
{
    //display valid items
    for(int itemCount=0;itemCount <
nord->out_nord.s_O_OL_CNT;itemCount++)
    {

appendText(&html,ittoa(nord->in_nord.in_item[itemCount].s_OL_SUPPLY_W
_ID,buffer,10),8,1);

appendText(&html,ittoa(nord->in_nord.in_item[itemCount].s_OL_I_ID,buffer,1
0),10,1);

appendText(&html,nord->out_nord.item[itemCount].s_I_NAME,DEFAULT_S
TRING_LEN+1,1);

appendText(&html,ittoa(nord->in_nord.in_item[itemCount].s_OL_QUANTITY,
buffer,10),5,1);

appendText(&html,ittoa(nord->out_nord.item[itemCount].s_S_QUANTITY,buf
fer,10),7,1);

        html+=sprintf(html,"%c  $%-7.2lf $%-7.2lf
<BR> ",nord->out_nord.item[itemCount].s_brand_generic,

nord->out_nord.item[itemCount].s_I_PRICE/100.0,

nord->out_nord.item[itemCount].s_OL_AMOUNT/100.0);
    }
    //display blank line for remaining empty items in the
order
    for(int lineBreaks=0;lineBreaks <
(NORD_ITEMS-nord->out_nord.s_O_OL_CNT);lineBreaks++)
        appendText(&html," <BR>");
    }
    else
        appendText(&html," <BR> <BR> <BR> <BR> <BR>
<BR> <BR> <BR> <BR> <BR> <BR> <BR> <BR> <BR> <BR>");

    appendText(&html," \r\n <BR> ");

    html+=displayStatus(html,rc);
    if(rc == OK)
        html+=sprintf(html," Total:
$%.2lf",nord->out_nord.s_total_amount/100.0);
    else
        appendText(&html," Total: <BR>");

    appendText(&html,"</PRE></BODY> </HTML>");

    DEBUGMSG("nord html page complete. returning to calling
function" << endl);

    return OK;

```

```

}
/*
*****
** Name          : doNewOrderErrorPage
** Description   :
**              : HTML neworder page entry point
** Parameters   :
**              : char *          html result
page
**              : char *          error
message
**              : htmlPhraser*   command block
**              : TXN_HANDLE*    txn handle
struct
** Returns      :
**              : int - return code
** Comments     :
**
*****
*/

int doNewOrderErrorPage(char *htmlPage,char *message,htmlPhraser
*commandBlock,TXN_HANDLE *txnHandle)
{
    char *html=htmlPage;

    appendText(&html,"<HTML><HEAD><TITLE>TPC-C New
Order</TITLE></HEAD>\r\n"
                "<BODY><FORM
ACTION=\"\"
                APP_NAME
                \"\"
METHOD=\"GET\">\r\n"
                "<CENTER><H3>Please Fill In New Order Form.</H3></CENTER>\r\n"
                "Submit Transaction
<INPUT TYPE=\"submit\" NAME=\"\"
                CMD_TXN_ID
                \"\" VALUE=\"\"
                CMD_NORD
                \"\">");

    //append the hidden warehouse and district fields
    html+=appendHiddenFields(html,txnHandle);

    //int buffer for warehouse
    char buffer[15];
    /*appendText(&html,"<PRE>      1      2      3      4      5
6      7      8      9\r\n"
"123456789012345678901234567890123456789012345678901234567890123
456789012345678901234567890\r\n"
                "Warehouse: ");*/
    appendText(&html,"<PRE>Warehouse: ");
    appendText(&html,ittoa(txnHandle->w_id,buffer,10),7,1);
    appendText(&html,"District: <INPUT NAME=\"\"
                CMD_D_ID
                \"\" SIZE=1>
Date:<BR>"
                "Customer <INPUT NAME=\"\"
                CMD_C_ID
                \"\" SIZE=6> Name:
Credit:   %Disc.:<BR>"
                "Order Number:
Number of Lines:   W_tax:   D_tax:<BR><BR>"

```

```

4      5      6      7      8      9\r\n"
// " 1 2 3
// "123456789012345678901234567890123456789012345678901234567890123456789012345678901234567890\r\n"
Item_Name Qty Stock B/G Price Amount <BR> ";
" Supp_W Item_Num
//append the 15 items commands
html+=appendItems(html,NORD_ITEMS,ITEM_START);
appendText(&html,message);
//seal up html page
appendText(&html,"</PRE></BODY></HTML>");
return OK;
}
/*
*****
** Name : doPaymentForm
** Description :
** Parameters : HTML payment page entry point
** block htmlPhraser* command
** TXN_HANDLE* txn handle
struct
** Returns :
** int - return code
** Comments :
**
*****
*/
int doPaymentForm(htmlPhraser *commandBlock, TXN_HANDLE
*txnHandle)
{
char *html=txnHandle->htmlPage;
appendText(&html,"<HTML><HEAD><TITLE>TPC-C
Payment</TITLE></HEAD>\r\n"
" <BODY><FORM
ACTION=\"\"
APP_NAME
\"
METHOD=\"GET\">\r\n"
"<CENTER><H3>Please Fill In Payment Form.</H3></CENTER> <BR>\r\n"
"Submit Transaction
<INPUT TYPE=\"submit\" NAME=\"\"
CMD_TXN_ID
\" VALUE=\"\"
CMD_PYMT
\">");
html+=appendHiddenFields(html,txnHandle);
appendText(&html,"<BR><PRE>\r\n"
"Date:<BR>"
"Warehouse: ");
char buffer[15];
appendText(&html,itoa(txnHandle->w_id,buffer,10));
appendSpaces(&html,10);
appendText(&html,"District: <INPUT NAME=\"\"
CMD_D_ID
\" SIZE=1>\r\n<BR>"
"<BR><BR><BR>"

```

```

"Customer: "
"<INPUT NAME=\"\"
CMD_C_ID
\" SIZE=5>"
" "
"Cust-Warehouse: "
"<INPUT NAME=\"\"
CMD_C_W_ID
\" SIZE=5>"
" "
"Cust-District: "
"<INPUT NAME=\"\"
CMD_C_D_ID
\" SIZE=1><BR>"
"Name: <INPUT
NAME=\"\"
CMD_C_NAME
\" SIZE=20>";
appendText(&html," Since: <BR>"
" "
" "
Credit: <BR>"
" "
" "
"%Disc: <BR>"
"Amount Paid: "
"<INPUT NAME=\"\"
CMD_AMT_PAID
\" SIZE=10>"
" "
"New
Cust-Balance:<BR>"
"Credit Limit:<BR>"
"<BR>Cust-Data:<BR> <BR> <BR> <BR> </PRE>");
return OK;
}
/*
*****
** Name : doPaymentResults
** Description :
** Parameters : HTML neworder page entry point
** block htmlPhraser* command
** TXN_HANDLE* txn handle
struct
** Returns :
** int - return code
** Comments :
**
*****
*/
int doPaymentResults(htmlPhraser *commandBlock, TXN_HANDLE
*txnHandle)
{
char *html=txnHandle->htmlPage;
char buffer[50];
struct paym_wrapper *pymt = NULL;
pymt = (paym_wrapper*)txnHandle->comInterface.txnBuffer;
ZeroMemory(pymt,maxDataSize);
//set login warehouse id from command block
pymt->in_paym.s_W_ID = txnHandle->w_id;

```



```

        //set district from command block
        if( (pymt->in_paym.s_D_ID = atoi(commandBlock->get_D_ID()))
    == 0)
        {
doPaymentErrorPage(html,ERR_INVALID_D_ID,commandBlock,txnHandle);
            return OK;
        }

        //set customer id from command block
        if( (pymt->in_paym.s_C_ID = atoi(commandBlock->get_C_ID()))
    == 0)
        {
            if(*(commandBlock->get_C_NAME()) == NULL)
            {
                //no customer id nor customer last name
                specified.
doPaymentErrorPage(html,ERR_MISSING_C_ID_OR_CLAST,commandBlock,txnHandle);
                    return OK;
            }
            else
                strcpy(pymt->in_paym.s_C_LAST,commandBlock->get_C_NAME());
        }
        else
        {
            //make sure that the user only inserted just c_id
            if(*(commandBlock->get_C_NAME()) != NULL)
            {
doPaymentErrorPage(html,ERR_C_ID_OR_CLAST_ONLY,commandBlock,txnHandle);
                    return OK;
            }
        }

        //get customer warehouse id field
        if( (pymt->in_paym.s_C_W_ID =
    atoi(commandBlock->get_C_W_ID())) == 0)
        {
doPaymentErrorPage(html,ERR_INVALID_C_W_ID,commandBlock,txnHandle);
            return OK;
        }

        //get customer district id field
        if( (pymt->in_paym.s_C_D_ID =
    atoi(commandBlock->get_C_D_ID())) == 0)
        {
doPaymentErrorPage(html,ERR_INVALID_C_D_ID,commandBlock,txnHandle);
            return OK;
        }

        if(!copyInMoney64(commandBlock->get_AMT_PAID(),&pymt->in_paym.s_H
    _AMOUNT))
        {
doPaymentErrorPage(html,ERR_INVALID_PAYMENT_AMOUNT,commandBlock,txnHandle);
            return OK;
        }
    
```

```

        appendText(&html,"<HTML><HEAD><TITLE>TPC-C Payment
    Results</TITLE></HEAD>\r\n"
            "<BODY><FORM
    ACTION=\"\"
            APP_NAME
            \"\"
    METHOD=\"GET\">\r\n");
        html+=appendButtons(html);
        html+=appendHiddenFields(html,txnHandle);
appendText(&html,"</FORM><CENTER><H3>Payment</H3></CENTER>");
    };

        DEBUGMSG("Calling com entry api payment,
    w_id:<<pymt->in_paym.s_W_ID<<"
    d_id:<<pymt->in_paym.s_D_ID<<endl);

        //assume failure
        pymt->out_paym.s_transtatus = -1;
        HRESULThres;
        try
        {
            hres =
    txnHandle->comInterface.comHandle->doPayment(&txnHandle->comInterface
    .size,(UCHAR**)&txnHandle->comInterface.txnBuffer);
        }
        catch(...)
        {
            html+=sprintf(html,"ERROR: Com Payment call caused
    exeception to occur.</PRE></BODY></HTML>");
            ERRORMSG("ERROR : Com Payment call caused
    exeception to occur."<<endl);
            return OK;
        }

        if(FAILED(hres))
        {
            html+=sprintf(html,"ERROR: pymt com call failed,
    rc:%x</PRE></BODY></HTML>",hres);
            ERRORMSG("ERROR : pymt com call failed,
    rc:<<hres<<endl);
            return OK;
        }

        hres = txnHandle->comInterface.comHandle->doSetComplete();
        if(FAILED(hres))
        {
            html+=sprintf(html,"ERROR: pymt com doSetComplete
    failed, rc:%ld</PRE></BODY></HTML>",hres);
            ERRORMSG("ERROR : pymt com doSetComplete
    failed, rc:<<DEBUGADDRESS(hres)<<endl);
            return OK;
        }
        pymt = (paym_wrapper *)txnHandle->comInterface.txnBuffer;
        //get return code
        int rc = pymt->out_paym.s_transtatus;
        if( rc != OK)
        {
            html+=displayStatus(html,rc);
            appendText(&html,"</PRE></BODY></HTML>");
            ERRORMSG("Payment TXN ERROR"<<endl
    <<"pymt->in_paym.s_C_D_ID:"<<pymt->in_paym.s_C_D_ID<<endl
    <<"pymt->in_paym.s_C_ID:"<<pymt->in_paym.s_C_ID<<endl
    
```



```

appendText(&html,"<BR> <BR>Customer: ");
appendText(&html,itoa(pymt->out_paym.s_C_ID,buffer,10),5+1,1);

appendText(&html,"Cust-Warehouse: ");

appendText(&html,itoa(pymt->in_paym.s_C_W_ID,buffer,10),6+1,1);

appendText(&html,"Cust-District: ");
appendText(&html,itoa(pymt->in_paym.s_C_D_ID,buffer,10));

//add customer information
appendText(&html,"<BR>Name: ");

appendText(&html,pymt->out_paym.s_C_FIRST,FIRST_NAME_LEN+1,1);

appendText(&html,pymt->out_paym.s_C_MIDDLE,INITIALS_LEN+1,1);
DEBUGMSG("Last name:"<<pymt->out_paym.s_C_LAST<<endl);

appendText(&html,pymt->out_paym.s_C_LAST,LAST_NAME_LEN+5,1);

appendText(&html,"Since: ");
copyOutDate(buffer,pymt->out_paym.s_C_SINCE_time);
appendText(&html,buffer);

appendText(&html,"<BR>");
appendSpaces(&html,8);

appendText(&html,pymt->out_paym.s_C_STREET_1,STREET_LEN+20,1);
appendText(&html," Credit: ");
appendText(&html,pymt->out_paym.s_C_CREDIT);

appendText(&html,"<BR>");
appendSpaces(&html,8);

appendText(&html,pymt->out_paym.s_C_STREET_2,STREET_LEN+21,1);
appendText(&html,"%Disc: ");

html+=sprintf(html,"%2.2lf",pymt->out_paym.s_C_DISCOUNT/100.0);

appendText(&html,"<BR>");
appendSpaces(&html,8);

appendText(&html,pymt->out_paym.s_C_CITY,CITY_LEN+1,1);

appendText(&html,pymt->out_paym.s_C_STATE,STATE_LEN+1,1);

copyOutZip(buffer,pymt->out_paym.s_C_ZIP);
appendText(&html,buffer,15,1);

appendText(&html,"Phone: ");
copyOutPhone(buffer,pymt->out_paym.s_C_PHONE);
appendText(&html,buffer);

appendText(&html," <BR> <BR>Amount Paid: $");

html+=sprintf(html,"%-9.2lf",pymt->in_paym.s_H_AMOUNT/100.0);

appendText(&html," New Cust-Balance: $");

html+=sprintf(html,"%-9.2lf",pymt->out_paym.s_C_BALANCE/100.0);

appendText(&html,"<BR>Credit Limit: $");

html+=sprintf(html,"%-9.2lf",pymt->out_paym.s_C_CREDIT_LIM/100.0);

```

```

appendText(&html," <BR> <BR>Cust-Data: ");
if(pymt->out_paym.s_C_CREDIT[0] == 'B' &&
pymt->out_paym.s_C_CREDIT[1] == 'C')
{
appendCustData(&html,pymt->out_paym.s_C_DATA);
appendText(&html," <BR>");
}
else
appendText(&html," <BR> <BR> <BR>");

html+=displayStatus(html,rc);
appendText(&html,"</PRE></BODY></HTML>");

return OK;
}
/*
*****
** Name : doPaymentErrorPage
** Description :
** Parameters :
** char * html page
result
** char * error
message
** htmlPhraser * command block
** TXN_HANDLE* txn handle
struct
** Returns :
** int - return code
** Comments :
**
*****
*/

int doPaymentErrorPage(char *htmlPage,char *message,htmlPhraser
*commandBlock,TXN_HANDLE *txnHandle)
{
char *html=htmlPage;
appendText(&html,"<HTML><HEAD><TITLE>TPC-C
Payment</TITLE></HEAD>\r\n"
" <BODY><FORM
ACTION=\"\"
APP_NAME
\"\"
METHOD=\"GET\"\>\r\n"
"<CENTER><H3>Please Fill In Payment Form.</H3></CENTER> <BR>\r\n"
"Submit Transaction
<INPUT TYPE=\"submit\" NAME=\"\"
CMD_TXN_ID
\"\" VALUE=\"\"
CMD_PYMT
\"\">");
html+=appendHiddenFields(html,txnHandle);
appendText(&html,"<BR><PRE>\r\n"
"Date:<BR>
Warehouse: ");

char buffer[15];
appendText(&html,itoa(txnHandle->w_id,buffer,10));

appendSpaces(&html,10);
appendText(&html,"District: <INPUT NAME=\"\"
CMD_D_ID
\"\" SIZE=1>\r\n<BR>"
"<BR> <BR> <BR>
<BR>"

```

```

"Customer: "
"<INPUT NAME=\\"
CMD_C_ID
\\" SIZE=5>"
" "
"Cust-Warehouse: "
"<INPUT NAME=\\"
CMD_C_W_ID
\\" SIZE=6>"
" "
"Cust-District: "
"<INPUT NAME=\\"
CMD_C_D_ID
\\" SIZE=1><BR>"
"Name: <INPUT
NAME=\\"
CMD_C_NAME
\\" SIZE=20>";
appendText(&html," Since: <BR>"
" "
" "
" "
"Amount Paid: "
"<INPUT NAME=\\"
CMD_AMT_PAID
\\" SIZE=10>"
" "
"New
Cust-Balance:<BR>"
"Credit Limit:<BR>"
<BR> <BR> Cust-Data:<BR> <BR> <BR> <BR> <BR> ");
appendText(&html,message);
appendText(&html,"</PRE>");
return OK;
}
/*
*****
** Name : doOrderStatusForm
** Description :
** HTML orderStatus page entry
point
** Parameters :
** htmlPhraser* command
block
** TXN_HANDLE* txn handle
struct
** Returns :
** int - return code
** Comments :
**
*****
*/
int doOrderStatusForm(htmlPhraser *commandBlock,TXN_HANDLE
*txnHandle)
{
char *html=txnHandle->htmlPage;
appendText(&html,"<HTML><HEAD><TITLE>TPC-C Order
Status</TITLE></HEAD><BR></HTML>"
" <BODY><FORM
ACTION=\\"
APP_NAME
"Customer: "
"<INPUT NAME=\\"
METHOD=\\"GET\ ">\r\n"
"<CENTER><H3>Please Fill In Order Status Form.</H3></CENTER>"
"<BR>\r\n"
"Submit Transaction
"Submit Transaction
"Submit Transaction
CMD_TXN_ID
\\" VALUE=\\"
CMD_ORDS
\\">"
"<BR> ");
html+=appendHiddenFields(html,txnHandle);
appendText(&html,"<PRE>\r\n"
"Warehouse: ");
char buffer[15];
appendText(&html,itoa(txnHandle->w_id,buffer,10));
appendText(&html," District: <INPUT NAME=\\"
CMD_D_ID
\\" SIZE=1>\r\n<BR>"
"Customer: "
"<INPUT NAME=\\"
CMD_C_ID
\\" SIZE=5>"
" "
"Name: "
"<INPUT NAME=\\"
CMD_C_NAME
\\" SIZE=20><BR>"
"Cust-Balance: <BR>"
"Order-Number:
Entry-Date: Carrier-Number<BR>"
"Supply-W
Item-Num Oty Amount Delivery<BR></PRE>");
appendText(&html,"</BODY></HTML>");
return OK;
}
/*
*****
** Name : doOrderStatusResults
** Description :
** HTML orderStatus page entry
point
** Parameters :
** htmlPhraser* command
block
** char * html result
page
** Returns :
** int - return code
** Comments :
**
*****
*/
int doOrderStatusResults(htmlPhraser *commandBlock,TXN_HANDLE
*txnHandle)
{
char *html=txnHandle->htmlPage;
struct ords_wrapper *ords = NULL;
ords = (ords_wrapper *) txnHandle->comInterface.txnBuffer;
ZeroMemory(ords,maxDataSize);

```

```

//set warehouse login id from command blk
ords->in_ords.s_W_ID = txnHandle->w_id;

//set district login id from command blk
if( ords->in_ords.s_D_ID = atoi(commandBlock->get_D_ID())) ==
0)
{
doOrderStatusErrorPage(html,ERR_INVALID_D_ID,commandBlock,txnHandle);
return OK;
}
if( ords->in_ords.s_C_ID = atoi(commandBlock->get_C_ID())) ==
0)
{
if(*(commandBlock->get_C_NAME()) == NULL)
{
//no customer id nor customer last name
specified.
doOrderStatusErrorPage(html,ERR_MISSING_C_ID_OR_CLAST,commandBlock,txnHandle);
return OK;
}
else
strcpy(ords->in_ords.s_C_LAST,commandBlock->get_C_NAME());
else
{
//make sure that the user only inserted just c_id
if(*(commandBlock->get_C_NAME()) != NULL)
{
doOrderStatusErrorPage(html,ERR_C_ID_OR_CLAST_ONLY,commandBlock,txnHandle);
return OK;
}
}
appendText(&html,"<HTML><HEAD><TITLE>TPC-C Order
Status Results</TITLE></HEAD>\r\n"
" <BODY><FORM
ACTION=\"\"
APP_NAME
\"
METHOD=\"GET\">\r\n");
html+=appendButtons(html);
html+=appendHiddenFields(html,txnHandle);
appendText(&html,"</FORM>");
ords->out_ords.s_transtatus = -1;
HRESULThres;
try
{
hres =
txnHandle->comInterface.comHandle->doOrderStatus(&txnHandle->comInterface.comInterface.txnBuffer);
}
catch(...)
{
html+=sprintf(html,"ERROR: ords com call caused
exeception.</PRE></BODY></HTML>");
return OK;

```

```

}
if(FAILED(hres))
{
html+=sprintf(html,"ERROR: ords com call failed,
rc:%x</PRE></BODY></HTML>",hres);
ERRORMSG("ERROR : ords com call failed,
rc:"<<DEBUGADDRESS(hres)<<endl);
return OK;
}
hres = txnHandle->comInterface.comHandle->doSetComplete();
if(FAILED(hres))
{
html+=sprintf(html,"ERROR: ords com doSetComplete
failed, rc:%ld</PRE></BODY></HTML>",hres);
ERRORMSG("ERROR : ords com doSetComplete failed,
rc:"<<DEBUGADDRESS(hres)<<endl);
return OK;
}
ords = (ords_wrapper *)txnHandle->comInterface.txnBuffer;
int rc = ords->out_ords.s_transtatus;
if( rc != OK)
{
html+=displayStatus(html,rc);
appendText(&html,"</PRE></BODY></HTML>");
ERRORMSG("ERROR order status"<<endl
<<"ords->in_ords.s_C_ID:"<<ords->in_ords.s_C_ID<<endl
<<"ords->in_ords.s_C_LAST:"<<ords->in_ords.s_C_LAST<<endl
<<"ords->in_ords.s_D_ID:"<<ords->in_ords.s_D_ID<<endl
<<"ords->in_ords.s_W_ID:"<<ords->in_ords.s_W_ID<<endl
<<"ords->out_ords.deadlocks:"<<ords->out_ords.deadlocks<<endl
<<"ords->out_ords.s_C_BALANCE:"<<ords->out_ords.s_C_BALANCE<<endl
<<"ords->out_ords.s_C_FIRST:"<<ords->out_ords.s_C_FIRST<<endl
<<"ords->out_ords.s_C_ID:"<<ords->out_ords.s_C_ID<<endl
<<"ords->out_ords.s_C_ID:"<<ords->out_ords.s_C_ID<<endl
<<"ords->out_ords.s_C_MIDDLE:"<<ords->out_ords.s_C_MIDDLE<<endl
<<"ords->out_ords.s_O_CARRIER_ID:"<<ords->out_ords.s_O_CARRIER_ID
<<endl
<<"ords->out_ords.s_O_ENTRY_D_time:"<<ords->out_ords.s_O_ENTRY_D_
time<<endl
<<"ords->out_ords.s_O_ID:"<<ords->out_ords.s_O_ID<<endl
<<"ords->out_ords.s_ol_cnt:"<<ords->out_ords.s_ol_cnt<<endl);
return OK;
}
//start creating result body
appendText(&html,"</FORM><CENTER><H3>Order-Status</H3></CENTE
R>");
appendText(&html, "<BR><PRE>\r\nWarehouse: ");

```

```

char buffer[50];

appendText(&html,itoa(ords->in_ords.s_W_ID,buffer,10),6+1,1);
appendText(&html,"District: ");
appendText(&html,itoa(ords->in_ords.s_D_ID,buffer,10));
appendText(&html,"<BR>
                                     "Customer: ");

//get customer id
appendText(&html,itoa(ords->in_ords.s_C_ID,buffer,10),6+1,1);
appendText(&html,"Name: ");
//get first, middle, and last from wrapper

appendText(&html,ords->out_ords.s_C_FIRST,FIRST_NAME_LEN+1,1);
appendText(&html,ords->out_ords.s_C_MIDDLE,INITIALS_LEN+1,1);
appendText(&html,ords->out_ords.s_C_LAST,LAST_NAME_LEN+5,1);

//get customer balance from wrapper
appendText(&html,"r\nCust-Balance: $");
html+=sprintf(html,"%0.2lf",ords->out_ords.s_C_BALANCE/100.0);

//display order number, entry date, and carrier number
appendText(&html,"<BR> <BR>
                                     "Order-Number ");
appendText(&html,itoa(ords->out_ords.s_O_ID,buffer,10),12,1);
appendText(&html,"Entry-Date: ");
copyOutDate(buffer,ords->out_ords.s_O_ENTRY_D_time);
appendText(&html,buffer,22,1);

appendText(&html,"Carrier-Number: ");

appendText(&html,itoa(ords->out_ords.s_O_CARRIER_ID,buffer,10));

//add item title columns
appendText(&html,"<BR>
                                     "Supply-W  "
                                     "Item-Id  "
                                     "Qty    "
                                     "Amount  "
                                     "Delivery-Date<BR>
");

//display items
for (int
itemCount=0;itemCount<ords->out_ords.s_ol_cnt;itemCount++)
{
//appendSpaces(&html,2);

//get supp w

appendText(&html,itoa(ords->out_ords.item[itemCount].s_OL_SUPPLY_W_ID,buffer,10),11,1);

//get item num

appendText(&html,itoa(ords->out_ords.item[itemCount].s_OL_I_ID,buffer,10),11,1);

//get item qty

appendText(&html,itoa(ords->out_ords.item[itemCount].s_OL_QUANTITY,buffer,10),6,1);

//get item dollar amount

html+=sprintf(html,"$%-14.2lf",ords->out_ords.item[itemCount].s_OL_AMOUNT/100.0);

```

```

//get delivery date
copyOutDate(buffer,ords->out_ords.item[itemCount].s_OL_DELIVERY_D_time);
appendText(&html,buffer);
appendText(&html," <BR> ");
}

//append line breaks if item count is less than 15
for (int itemCount=0;itemCount <
(15-ords->out_ords.s_ol_cnt);itemCount++)
appendText(&html,"<BR> ");

html+=displayStatus(html,rc);

appendText(&html,"</PRE></BODY></HTML>");

return OK;
}

/*
*****
** Name          : doOrderStatusErrorPage
** Description   :
**               : HTML orderStatus error page
** Parameters    :
**               : char *          html page
result
**               : char *          error
message
**               : htmlPhraser*    command
block
**               : TXN_HANDLE*     txn handle
** Returns      :
**               : int - return code
** Comments     :
**
*****
*/

int doOrderStatusErrorPage(char *htmlPage,char *message,htmlPhraser
*commandBlock,TXN_HANDLE *txnHandle)
{
char *html=htmlPage;

appendText(&html,"<HTML><HEAD><TITLE>TPC-C Order
Status</TITLE></HEAD>\r\n"
                                     "<BODY><FORM
ACTION=\\"
                                     APP_NAME
                                     \""
METHOD=\\"GET\ ">\r\n"

"<CENTER><H3>Please Fill In Order Status Form.</H3></CENTER>
<BR>\r\n"

<INPUT TYPE=\\"submit\ " NAME=\\"
                                     "Submit Transaction
                                     CMD_TXN_ID
                                     \\" VALUE=\\"
                                     CMD_ORDS
                                     \\">
                                     "<BR> ");

html+=appendHiddenFields(html,txnHandle);

appendText(&html,"<PRE>\r\n"
                                     "Warehouse: ");

```

```

char buffer[15];
appendText(&html,itoa(txnHandle->w_id,buffer,10));

appendText(&html,"      District: <INPUT NAME=\"\"
                                CMD_D_ID
                                \"\|\" SIZE=1>\r\n<BR>\"
                                \"Customer: \"
                                <INPUT NAME=\"\"
                                CMD_C_ID
                                \"\|\" SIZE=5>\"
                                \" \"
                                \"Name: \"
                                <INPUT NAME=\"\"
                                CMD_C_NAME
                                \"\|\" SIZE=20><BR>\"
                                \"Cust-Balance: <BR>\"
                                \"Order-Number:
Entry-Date:      Carrier-Number<BR>\"
Item-Num  Qty    Amount    Delivery <BR>\"");
                                \"Supply-W
                                <BR>\"");
appendText(&html,message);
appendText(&html,\"</PRE><</BODY><</HTML>\"");
return OK;
}
/*
*****
** Name          : doDeliveryForm
** Description    :
**               HTML payment page entry point
** Parameters    :
**               htmlPhraser*    command
block
**               TXN_HANDLE*    txn handle
struct
** Returns       :
**               int - return code
** Comments      :
**
*****
*/
int doDeliveryForm(htmlPhraser *commandBlock,TXN_HANDLE
*txnHandle)
{
    char *html=txnHandle->htmlPage;

    appendText(&html,\"<HTML><HEAD><TITLE>TPC-C
Delivery</TITLE></HEAD>\r\n\"
                                \"<BODY><FORM
ACTION=\"\"
                                APP_NAME
                                \"\|\"
METHOD=\"GET\">\r\n\"
                                \"<CENTER><H3>Delivery.</H3></CENTER>\r\n\"
                                \"Submit Transaction
<INPUT TYPE=\"submit\" NAME=\"\"
                                CMD_TXN_ID
                                \"\|\" VALUE=\"\"
                                CMD_DLVY
                                \"\|\">\"");
    html+=appendHiddenFields(html,txnHandle);
    appendText(&html,\"<BR> <PRE>\"
                                \"Warehouse: \");
    char buffer[10];

```

```

appendText(&html,itoa(txnHandle->w_id,buffer,10));

appendText(&html,\" <BR> <BR>\"
                                \"Carrier Number: \"
                                \"<INPUT NAME=\"\"
                                CMD_CARRIER_NUM
                                \"\|\" SIZE=1>\"
                                \"</FORM><</PRE>\"");
appendText(&html,\"</BODY><</HTML>\"");
return OK;
}
/*
*****
** Name          : doDeliveryResults
** Description    :
**               HTML payment page entry point
** Parameters    :
**               htmlPhraser*    command
block
**               TXN_HANDLE*    txn handle
** Returns       :
**               int - return code
** Comments      :
**
*****
*/
int doDeliveryResults(htmlPhraser *commandBlock,TXN_HANDLE
*txnHandle)
{
    char *html = txnHandle->htmlPage;

    //declare delivery structure
    struct dlvy_wrapper    dlvy;

    //set warehouse login id from command blk
    dlvy.in_dlvvy.s_W_ID = txnHandle->w_id;

    //set the carrier id from command blk
    if( dlvy.in_dlvvy.s_O_CARRIER_ID =
atoi(commandBlock->get_CARRIER_NUM()) == 0)
    {
        doDeliveryErrorPage(html,ERR_INVALID_CARRIER,commandBlock,txnHan
dle);
        return OK;
    }

    //print title, add hidden fields , txn buttons
    appendText(&html,\"<HTML><HEAD><TITLE>TPC-C Delivery
Results</TITLE></HEAD>\r\n<BODY><<FORM ACTION=\"\"
                                APP_NAME
                                \"\|\"
METHOD=\"GET\">\r\n\"");
    html+=appendButtons(html);
    html+=appendHiddenFields(html,txnHandle);
    appendText(&html,
\"<FORM><CENTER><H3>Delivery</H3></CENTER>\"");
    int rc =
queueDlvyTxn(dlvy.in_dlvvy.s_W_ID,dlvy.in_dlvvy.s_O_CARRIER_ID);

```

```

        if( rc != OK)
        {
            html+=displayStatus(html,rc);
            appendText(&html,"</PRE></BODY></HTML>\r\n");

            ERRORMSG("ERROR: Unable to queue dlvy txn,
rc:"<<rc<<endl);
            return OK;
        }

        //start creating result body
        appendText(&html,"Warehouse: ");

        //get w_id from wrapper
        char buffer[15];
        appendText(&html,ittoa(dlvy.in_dlvy.s_W_ID,buffer,10));
        appendText(&html,"<BR> <BR>Carrier Number: ");

        //get carrier_id from wrapper

        appendText(&html,ittoa(dlvy.in_dlvy.s_O_CARRIER_ID,buffer,10));
        appendText(&html,"<BR> <BR>Execution Status: Delivery has
        been queued </PRE></BODY></HTML>");

        return OK;
    }
    /*
    *****
    ** Name          : doDeliveryErrorPage
    ** Description   :
    **              HTML payment error page entry
    point
    ** Parameters    :
    **              char *          html result
    page
    **              char *          error
    message
    **              htmlPhraser     command
    block
    **              TXN_HANDLE*     txn handle
    ** Returns      :
    **              int - return code
    ** Comments     :
    *****
    */
    int doDeliveryErrorPage(char *htmlPage,char *message,htmlPhraser
    *commandBlock, TXN_HANDLE *txnHandle)
    {
        char *html=htmlPage;

        appendText(&html,"<HTML><HEAD><TITLE>TPC-C
        Delivery</TITLE></HEAD>\r\n"
        "ACTION=\\""
        APP_NAME
        "\"
        METHOD=\\"GET\\"">\r\n"
        "<CENTER><H3>Delivery.</H3></CENTER>\r\n"
        "<INPUT TYPE=\\"submit\\"" NAME=\\""
        CMD_TXN_ID
        "\" VALUE=\\""
        CMD_DLVS
        "\">");
    }

```

```

        html+=appendHiddenFields(html,txnHandle);

        appendText(&html,"<BR> <PRE>"
        "Warehouse: ");

        char buffer[15];
        appendText(&html,ittoa(txnHandle->w_id,buffer,10));

        appendText(&html," <BR> <BR>"
        "Carrier Number: "
        "<INPUT NAME=\\""
        CMD_CARRIER_NUM
        "\" SIZE=1> <BR>");

        appendText(&html,message);
        appendText(&html,"</PRE></BODY></HTML>");

        return OK;
    }
    /*
    *****
    ** Name          : doStockForm
    ** Description   :
    **              HTML stock page entry point
    ** Parameters    :
    **              htmlPhraser     command
    block
    **              TXN_HANDLE*     txn handle
    ** Returns      :
    **              int - return code
    ** Comments     :
    *****
    */
    int doStockForm(htmlPhraser *commandBlock, TXN_HANDLE *txnHandle)
    {
        char *html=txnHandle->htmlPage;
        appendText(&html,"<HTML><HEAD><TITLE>TPC-C Stock
        Level</TITLE></HEAD>\r\n"
        "ACTION=\\""
        APP_NAME
        "\"
        METHOD=\\"GET\\"">\r\n"
        "<CENTER><H3>Please Fill In Stock Form.</H3></CENTER> <BR>\r\n"
        "Submit Transaction
        <INPUT TYPE=\\"submit\\"" NAME=\\""
        CMD_TXN_ID
        "\" VALUE=\\""
        CMD_STOK
        "\">");

        html+=appendHiddenFields(html,txnHandle);

        appendText(&html,"<PRE>"
        "Warehouse: ");

        char buffer[15];
        appendText(&html,ittoa(txnHandle->w_id,buffer,10),6+1,1);
        appendText(&html,"District: ");

        appendText(&html,ittoa(txnHandle->d_id,buffer,10));
        appendText(&html," <BR> <BR>"
        "Stock Level
        Threshold: "
    }

```



```

                                "<INPUT NAME=\"\"
CMD_STK_THRESHOLD
                                "\" SIZE=1> <BR>
<BR>"
                                "Low Stock: <BR>"
                                "</PRE>");
                                appendText(&html,"</FORM></BODY></HTML>");
                                return OK;
}
/*
*****
** Name          : doStockResults
** Description    :
**               HTML stock page entry point
** Parameters    :
**               htmlPhraser*    command
block
**               TXN_HANDLE*    txn handle
struct
** Returns      :
**               int - return code
** Comments     :
**
*****
*/
int doStockResults(htmlPhraser *commandBlock, TXN_HANDLE *txnHandle)
{
    char *html = txnHandle->htmlPage;

    struct stok_wrapper *stok;
    stok = (stok_wrapper*)txnHandle->comInterface.txnBuffer;
    ZeroMemory(stok,maxDataSize);

    //set warehouse login id from command blk
    stok->in_stok.s_W_ID = txnHandle->w_id;

    //set district login id from command blk
    stok->in_stok.s_D_ID = txnHandle->d_id;

    //set stock level threshold id from command blk
    if( (stok->in_stok.s_threshold =
atoi(commandBlock->get_STK_THRESHOLD()) == 0)
    {

doStockErrorPage(html,ERR_INVALID_THRESHOLD,commandBlock,txnHandle);
        return OK;
    }
    //assume failure, set s_transtatus to err
    stok->out_stok.s_transtatus = INVALID_STATUS;

    //print title, add hidden fields , txn buttons
    appendText(&html,"<HTML><HEAD><TITLE>TPC-C Stock
Level Results</TITLE></HEAD><\r\n\"
                                "<BODY><FORM
ACTION=\"\"
                                APP_NAME
                                \"\"
METHOD=\"GET\"><\r\n\"");

    html+=appendButtons(html);

    html+=appendHiddenFields(html,txnHandle);

```

```

                                appendText(&html,"</FORM>");
                                stok->out_stok.s_transtatus = -1;

                                DEBUGMSG("Calling com entry api for stock call,
w_id:"<<stok->in_stok.s_W_ID<<" d_id:"<<stok->in_stok.s_D_ID<<
                                " threshold:"<<stok->in_stok.s_threshold<<endl);

                                HRESULThres;
                                try
                                {
                                    hres =
txnHandle->comInterface.comHandle->doStockLevel(&txnHandle->comInterface.size,(UCHAR*)&txnHandle->comInterface.txnBuffer);
                                }
                                catch(...)
                                {
                                    html+=sprintf(html,"ERROR: Com Stock call caused
exeception to occur.</PRE></BODY></HTML>");
                                    ERRORMSG("ERROR : Com Stock call caused
exeception to occur."<<endl);
                                    return OK;
                                }

                                if(FAILED(hres))
                                {
                                    html+=sprintf(html,"ERROR: stok com call failed,
rc:%ld</PRE></BODY></HTML>",hres);
                                    ERRORMSG("ERROR : stok com call failed,
rc:"<<DEBUGADDRESS(hres)<<endl);
                                    return OK;
                                }

                                hres = txnHandle->comInterface.comHandle->doSetComplete();
                                if(FAILED(hres))
                                {
                                    html+=sprintf(html,"ERROR: stok com doSetComplete
failed, rc:%ld</PRE></BODY></HTML>",hres);
                                    ERRORMSG("ERROR : stok com doSetComplete failed,
rc:"<<DEBUGADDRESS(hres)<<endl);
                                    return OK;
                                }
                                stok = (stok_wrapper *)txnHandle->comInterface.txnBuffer;
                                int rc = stok->out_stok.s_transtatus;
                                if(rc != OK)
                                {
                                    html+=displayStatus(html,rc);
                                    appendText(&html,"</PRE></BODY></HTML>");
                                    ERRORMSG("ERROR stok txn failed"<<endl

<<"stok->in_stok.s_D_ID:"<<stok->in_stok.s_D_ID<<endl
<<"stok->in_stok.s_threshold:"<<stok->in_stok.s_threshold<<endl
<<"stok->in_stok.s_W_ID:"<<stok->in_stok.s_W_ID<<endl
<<"stok->out_stok.deadlocks:"<<stok->out_stok.deadlocks<<endl
<<"stok->out_stok.s_low_stock:"<<stok->out_stok.s_low_stock<<endl
<<"stok->out_stok.s_transtatus:"<<stok->out_stok.s_transtatus<<endl);
                                return OK;
                                }

                                //start creating result body

```

```

appendText(&html,"<FORM><CENTER><H3>Stock-Level</H3></CENTER>
>");
    appendText(&html,"<BR><PRE>\r\n"
                "Warehouse: ");

    //get w_id from wrapper
    char buffer[10];
    appendText(&html,itoa(stok->in_stok.s_W_ID,buffer,10),6+1,1);

    appendText(&html,"District: ");
    appendText(&html,itoa(stok->in_stok.s_D_ID,buffer,10));

    appendText(&html," <BR> <BR>"
                "Stock Level
Threshold: ");
    appendText(&html,itoa(stok->in_stok.s_threshold,buffer,10));

    appendText(&html," <BR> <BR>"
                "Low Stock: ");
    appendText(&html,itoa(stok->out_stok.s_low_stock,buffer,10));
    appendText(&html," <BR> <BR>");

    html+=displayStatus(html,rc);
    appendText(&html,"</PRE></BODY></HTML>");

    return OK;
}

```

```

/*
*****
** Name          : doStockErrorPage
** Description   :
**              : HTML stock page entry point
** Parameters    :
**              : char *          html result
page
**              : char *          query string
**              : htmlPhraser    command
block
**              : TXN_HANDLE *   handle for
this transaction
** Returns      :
**              : int - return code
** Comments     :
**
*****
*/

```

```

int doStockErrorPage(char *htmlPage,char *message,htmlPhraser
*commandBlock,TXN_HANDLE *txnHandle)
{
    char *html=htmlPage;

    appendText(&html,"<HTML><HEAD><TITLE>TPC-C Stock
Level</TITLE></HEAD>\r\n"
                "<BODY><FORM
ACTION=\\"
                APP_NAME
                "\\"
METHOD=\\"GET\\"">\r\n"
"<CENTER><H3>Please Fill In Stock Form.</H3></CENTER> <BR>\r\n"
                "Submit Transaction
<INPUT TYPE=\\"submit\\" NAME=\\"
                CMD_TXN_ID
                "\\" VALUE=\\"
                CMD_STOK

```

```

                "\>");
html+=appendHiddenFields(html,txnHandle);
    appendText(&html,"<PRE>"
                "Warehouse: ");

    char buffer[15];
    appendText(&html,itoa(txnHandle->w_id,buffer,10));
    appendSpaces(&html,2);
    appendText(&html,"District: ");
    appendText(&html,commandBlock->get_D_ID());
    appendText(&html," <BR> <BR>"
                "Stock Level
Threshold: "
                "<INPUT NAME=\\"
                CMD_STK_THRESHOLD
                "\\" SIZE=1> <BR>
<BR>"
                "Low Stock: <BR>");

    appendText(&html,message);

    appendText(&html,"</PRE></FORM></BODY></HTML>");

    return OK;
}

```

```

/*
*****
** Name          : doExit
** Description   :
**              : HTML exit page entry point
** Parameters    :
**              : htmlPhraser*   command
**              : TXN_HANDLE*   txn handle
struct
** Returns      :
**              : int - return code
** Comments     :
**
*****
*/

```

```

int doExit(htmlPhraser *commandBlock,TXN_HANDLE *txnHandle)
{
    return (doLoginForm(commandBlock,txnHandle));
}

```

```

/*
*****
** Name          : displayStatus
** Description   :
**              : appends status string to the html
page
** Parameters    :
**              : char*          html page
**              : int            rc
** Returns      :
**              : amount of characters the function
appended
**              : to the html page
** Comments     :
**
*****
*/

```

```

int displayStatus(char *htmlPage,int rc)
{
    char *html = htmlPage;

    appendText(&html,"");

    switch (rc)
    {
    case OK:
        appendText(&html,"Execution Status: Transaction
Committed",50,1);
        break;
    case INVALID_ITEM:
        appendText(&html,"Execution Status: Item number is not
valid",50,1);
        break;
    case INVALID_STATUS:
        appendText(&html,"Execution Status: ERROR: Rollback
INVALID_STATUS",50,1);
        break;
    case INVALID_COM_STATUS:
        appendText(&html,"Execution Status: ERROR: Rollback
COM FAILURE",50,1);
        break;
    case ERR_DLVY_QUEUE_FULL:
        appendText(&html,"Execution Status: ERROR: Rollback
DLVY QUEUE FULL",50,1);
        break;
    default:
        appendText(&html,"Execution Status: ERROR:
Rollback",50,1);

    };

    appendText(&html," ");

    return (int)(html - htmlPage);
}
/*
*****
** Name          : appendButtons
** Description    :
**               : append hidden field to recognize
user after login
**
** Parameters     :
**               : *htmlPage
html result page
**               : *TXN_HANDLE
txn handle
** Returns       :
**               : int
amount of characters the function appened
**
**               :
to the html page
** Comments      :
**
*****
*/
int appendHiddenFields(char *htmlPage,TXN_HANDLE *txnHandle)
{
    char *html = htmlPage;
    char buffer[15];

    appendText(&html,"<INPUT TYPE=\"hidden\" NAME=\"\"
CMD_TERM_ID
\" VALUE=\"\"");
    appendText(&html,itoa(txnHandle->term_id,buffer,10));

```

```

        appendText(&html,">\r\n");
        return (int)(html-htmlPage);
    }
}
/*
*****
** Name          : appendButtons
** Description    :
**               : appends buttons transaction
buttons to result page
** Parameters     :
**               : *htmlPage
**
** Returns       :
**               : amount of characters the function
appened
**               : to the html page
** Comments      :
**
*****
*/
int appendButtons(char *htmlPage)
{
    char *html = htmlPage;

    appendText(&html,"<INPUT TYPE=\"submit\" NAME=\"\"
CMD_TXN_ID
\" VALUE=\"\"
CMD_NORD
\">\r\n"
"<INPUT
TYPE=\"submit\" NAME=\"\"
CMD_TXN_ID
\" VALUE=\"\"
CMD_PYMT
\">\r\n"
"<INPUT
TYPE=\"submit\" NAME=\"\"
CMD_TXN_ID
\" VALUE=\"\"
CMD_ORDS
\">\r\n"
"<INPUT
TYPE=\"submit\" NAME=\"\"
CMD_TXN_ID
\" VALUE=\"\"
CMD_DLVY
\">\r\n"
"<INPUT
TYPE=\"submit\" NAME=\"\"
CMD_TXN_ID
\" VALUE=\"\"
CMD_STOK
\">\r\n"
"<INPUT
TYPE=\"submit\" NAME=\"\"
CMD_TXN_ID
\" VALUE=\"\"
CMD_EXIT
\">\r\n <BR>");

    return (int)(html - htmlPage);
}
/*
*****
** Name          : appendItems

```

```

** Description      :
**                  appends items to new order and
order status page
** Parameters      :
**                  *htmlPage
**                  html result page
**                  short
**                  items to append
**                  short
**                  item CMD id start
**
** Returns         :
**                  amount of characters the function
appended
**                  to the html page
** Comments       :
**
*****
*/
int appendItems(char *htmlPage,short itemCount,short cmdIDStart)
{
    char *html = htmlPage;
    char numBuffer[MAX_INT_BUFFER];

    for(int item=0;item < itemCount;item++)
    {
        appendText(&html,"<BR><INPUT NAME=\"");
        appendText(&html,ittoa(cmdIDStart++,numBuffer,10));
        appendText(&html,"\" SIZE=6> <INPUT NAME=\"");
        appendText(&html,ittoa(cmdIDStart++,numBuffer,10));
        appendText(&html,"\" SIZE=6>
<INPUT NAME=\"");
        appendText(&html,ittoa(cmdIDStart++,numBuffer,10));
        appendText(&html,"\" SIZE=2>\r\n");
    }

    return (int)(html - htmlPage);
}

/*
*****
** Name           : dlvyThreadEntry
** Description    :
**                  dlvy thread worker entry point
** Parameters     :
**
** Returns       :
**
** Comments     :
**                  All dlvy threads created by
initDly enter at
**                  this point. They must first make a
connection
**                  to the database, then go to sleep.
**
**                  Main isapi threads control dlvy
worker semaphore
**                  and signal when a dlvy txn is
queued.
**
*****
*/

void dlvyThreadEntry(void *)
{
    int          rc = 0;

```

```

    DEBUGMSG("dlvyThread " << GetCurrentThreadId() << " entered
dlvyThreadEntry, calling db_connect to db:" << dbName << endl);

    void *connectHandle;
    //connect to database.
    DEBUGMSG("ptr created. calling db_connect to db:" << dbName
<< endl);
    rc = db_connect(dbName,&connectHandle);

    if(rc != OK)
    {
        ERRORMSG("dlvyThread " << GetCurrentThreadId()
<<" unable to connect to database, rc:" << rc << endl);
        DEBUGMSG("dlvyThread " << GetCurrentThreadId()
<<" unable to connect to database, rc:" << rc << endl);
        return;
    }

    DEBUGMSG("dlvyThread " << GetCurrentThreadId() << " connect
to db:" << dbName << " successful" << endl);

    FILE *dlvyLog = NULL;
    char logFileName[MAX_STRING_LEN] = {NULL};

    EnterCriticalSection(&isapiLock);
    //open dlvy log file for this thread
    sprintf(logFileName,"%s\\del_%.txt",dlvyLogPath,dlvyThreadID);
    dlvyLog = fopen(logFileName,"w");
    if(!dlvyLog)
    {
        ERRORMSG("dlvyThread " << GetCurrentThreadId()
<<" unable to open dlvy log "
<< dlvyLogPath << "\\del_" <<
dlvyThreadID << endl);
        DEBUGMSG("dlvyThread " << GetCurrentThreadId()
<<" unable toopen dlvy log "
<< dlvyLogPath << "\\del_" <<
dlvyThreadID << endl);
        return;
    }

    //increment the global dlvy thread id
    dlvyThreadID++;

    LeaveCriticalSection(&isapiLock);

    DEBUGMSG("dlvyThread " << GetCurrentThreadId() <<" dlvy log
file name:" << logFileName << " open." << endl);

    HANDLE workerHandles[2];
    //handle array to store event to wait on

    struct DLVYQUEUEDATA          dlvyQueueData;
    //dlvy queue struct to store queued txn
    struct dlvy_wrapper           dlvyTxn;
    //dlvy wrapper of db2 structs

    struct _timeb
endQueueTime; //time stamp to queue removal time
    struct _timeb
endProcessTime; //time stamp for end process time

    char orderIDs[MAX_STRING_LEN] = {NULL};
    //string to store oids for each district
    int bytesWritten = 0;
    int dlvyCount = 0;

    DEBUGMSG("dlvyThread entering work loop" << endl);

```

```

//successful, while true
while(true)
{
    try
    {
handles" << endl);
        DEBUGMSG("dlvyThread initializing wait

work to do
        //wait for both program exit AND if there is
workerHandles[0] = dlvyThreadDone;
workerHandles[1] = dlvyThreadSemaphore;

        DEBUGMSG("dlvyThread going to sleep
waiting for wrk" << endl);

        rc =
WaitForMultipleObjects(2,&workerHandles[0],FALSE,INFINITE);

        DEBUGMSG("dlvyThread awake, checking
wake condition" << endl);

        if(rc == WAIT_OBJECT_0)
            break;
        else if(rc == (WAIT_OBJECT_0+1) )
        {
            DEBUGMSG("dlvyThread awake,
wake condition of dlvyThreadSemaphore" << endl);
        }

        DEBUGMSG("dlvyThread trying to enter
critical section" << endl);

        EnterCriticalSection(&dlvyQueueLock);

        DEBUGMSG("dlvyThread entered critical
section" << endl);

        //remove queued dlvy txn
dlvyQueueData.enqueueTime.time
= dlvyQueue[dlvyBufferThreadIndex].enqueueTime.time;
dlvyQueueData.enqueueTime.millitm
= dlvyQueue[dlvyBufferThreadIndex].enqueueTime.millitm;
dlvyQueueData.in_s_0_CARRIER_ID
= dlvyQueue[dlvyBufferThreadIndex].in_s_0_CARRIER_ID;
dlvyQueueData.warehouse
= dlvyQueue[dlvyBufferThreadIndex].warehouse;

        DEBUGMSG("dlvyThread removed dlvy:"
<< dlvyCount << ",w_id:" << dlvyQueueData.warehouse
<< " carrier_id:" << dlvyQueueData.in_s_0_CARRIER_ID << endl);

        DEBUGMSG("dlvyThread removed dlvy in
queue index: " <<dlvyBufferThreadIndex<< " w_id: " <<
dlvyQueueData.warehouse
<< " carrier_id: " << dlvyQueueData.in_s_0_CARRIER_ID << endl);

        //increment the number of free slots
dlvyBufferFreeSlots++;

        //increment the thread index to next slot in
dlvy queue
        dlvyBufferThreadIndex++;

```

```

        DEBUGMSG("dlvyThread incremented
amount of free slots:" << dlvyBufferFreeSlots << " and thread index:" <<
        dlvyBufferThreadIndex << endl);
        //check if we reached the end of dlvy queue, if
so, reset back index back to 0
        if(dlvyBufferThreadIndex == dlvyQueueLen)
        {
            DEBUGMSG("dlvyThread reset
dlvyBufferThreadIndex to 0, current dlvyBufferThreadIndex:" <<
dlvyBufferThreadIndex
            << " free
slots:"<<dlvyBufferFreeSlots<<endl);
            dlvyBufferThreadIndex=0;
        }
        DEBUGMSG("dlvyThread releasing critical
section" << endl);

        LeaveCriticalSection(&dlvyQueueLock);

        //take enqueue time
        _ftime(&endQueueTime);

        DEBUGMSG("dlvyThread executing txn
w_id:" << dlvyQueueData.warehouse
            << " carrier_id:" <<
dlvyQueueData.in_s_0_CARRIER_ID << endl);

        //prepare to call database
dlvyTxn.in_dlvy.s_O_CARRIER_ID
= dlvyQueueData.in_s_0_CARRIER_ID;
dlvyTxn.in_dlvy.s_W_ID
= dlvyQueueData.warehouse;
dlvyTxn.out_dlvy.s_transtatus
= -1;

        //increment dlvy count
dlvyCount++;

        DEBUGMSG("dlvyThread %d calling dlvy
txn" << rc << endl);

        //call dlvy txn
rc = dlvyCall(&dlvyTxn,connectHandle);

        _ftime(&endProcessTime);

rc = dlvyTxn.out_dlvy.s_transtatus;

        DEBUGMSG("dlvy txn response time:"<<
            (((endProcessTime.time -
endQueueTime.time)*1000)+
(endProcessTime.millitm - endQueueTime.millitm))/1000.0)<<
            "
w_id:"<<dlvyTxn.in_dlvy.s_W_ID<<" carrier:"
<<dlvyTxn.in_dlvy.s_O_CARRIER_ID<<
            "
deadLocks:"<<dlvyTxn.out_dlvy.deadlocks<<" rc:"<< rc <<endl);

        DEBUGMSG("dlvyThread dlvy s_transtatus:"
<< rc << endl);

        if(rc == OK)
        {
            bytesWritten=0;
            char *buffer = orderIDs;

```

```

        for(int
districtIndex=0;districtIndex <
DISTRICTS_PER_WAREHOUSE;districtIndex++)
        {
if(dlvTxn.out_dlv.s_O_ID[districtIndex] == 0)
        bytesWritten
= sprintf(buffer,"%nD_ID %d had no new orders",districtIndex);
        else
        bytesWritten
= sprintf(buffer,"%d ",dlvTxn.out_dlv.s_O_ID[districtIndex]);

        buffer+=bytesWritten;
        }
    }
    else
        sprintf(orderIDs,"%nDelivery
transaction failed");

fprintf(dlvLog,DELIVERY_LOG_SUCCESS_STR,
dlvyCount,
dlvyQueueData.enqueueTime.time,
dlvyQueueData.enqueueTime.millitm,
endQueueTime.time,
endQueueTime.millitm,
dlvyQueueData.warehouse,
dlvyQueueData.in_s_0_CARRIER_ID,
orderIDs,
endProcessTime.time,
endProcessTime.millitm);

        fflush(dlvLog);
    }
    catch(...)
    {
        ERRORMSG("ERROR: Unhandled
exception in dlvy thread. Thread exiting"<<endl);
        fprintf(dlvLog,"ERROR: Unhandled
exception in dlvy thread %ld. Thread exiting.\n",GetCurrentThreadId());
        fflush(dlvLog);

        LeaveCriticalSection(&dlvyQueueLock);
    }
} //end while true
}

/*
*****
** Name          : queueDlvyTxn
** Description   :
**              function queues dlvy txn in dlvy
queue
** Parameters    :
**              int          warehouse
**              short        carrier
** Returns       :
**              int          error code
** Comments     :
**              Function will queue
dlvy txn if 2 points are true
**              1) We have room in our
dlvy buffer
**              2) We writing over the
end of the queue
**
*****
*/
int queueDlvyTxn(int warehouse, short carrier_id)
{
    DEBUGMSG("Taking lock to queue dlvy txn.");
    EnterCriticalSection(&dlvyQueueLock);
    DEBUGMSG("Lock aquired to queue dlvy txn");
    if(dlvBufferFreeSlots)
    {
        DEBUGMSG("Checking if we are inserting at tail of
dlvy queue."<<endl);
        if( dlvyBufferSlotIndex == (dlvyBufferThreadIndex-1))
        {
            ERRORMSG("Error dlvy queue inserting
over unserviced queued dlvy txn."<<endl);
            DEBUGMSG("Error dlvy queue inserting
over unserviced queued dlvy txn."<<endl);
            LeaveCriticalSection(&dlvyQueueLock);
            return
ERR_DLVE_QUEUE_EATING_TAIL;
        }
        DEBUGMSG("free slots dlvy
queue:"<<dlvyBufferFreeSlots<<" inserting txn in slot: "
<<dlvyBufferSlotIndex<<
"w_id: "<<warehouse<<" carrier:
"<<carrier_id<<endl);
        dlvyQueue[dlvyBufferSlotIndex].warehouse =
warehouse;
        dlvyQueue[dlvyBufferSlotIndex].in_s_0_CARRIER_ID
= carrier_id;
        _ftime(&dlvyQueue[dlvyBufferSlotIndex].enqueueTime);

        //decrement the number of free slots in the buffer
dlvyBufferFreeSlots--;

        //increment the index to the next dlvy queue slot.
dlvyBufferSlotIndex++;

        DEBUGMSG("dlvy txn queued, slots available in
queue:"<<dlvyBufferFreeSlots<<" queue slot index:"<<dlvyBufferSlotIndex
<<"w_id:"<<warehouse<<"
carrier:"<<carrier_id<<endl);

        DEBUGMSG("dlvy txn queued, slots available in queue:
"<<dlvyBufferFreeSlots<<" queue slot index: "<<dlvyBufferSlotIndex
<<"w_id: "<<warehouse<<"
carrier: "<<carrier_id<<endl);

        if(dlvBufferSlotIndex == dlvyQueueLen)

```

```

        {
            DEBUGMSG("queue slot index hit end of
queue, reset to 0, current index:"<<dlvyBufferSlotIndex<<" free
slots:"<<dlvyBufferFreeSlots<<endl);
            dlvyBufferSlotIndex=0;
        }
    }
    else
    {
        //no slots available in dlvy buffer, release critical section
and return an nord->in_nord.in_item
        LeaveCriticalSection(&dlvyQueueLock);
        ERRORMSG("dlvy queue buffer full, increase the dlvy
queue length."<<endl);
        return ERR_DLVE_QUEUE_FULL;
    }

    LeaveCriticalSection(&dlvyQueueLock);

    //release semaphore to wake thread that there is work
    ReleaseSemaphore(dlvyThreadSemaphore,1,NULL);

    return OK;
}

/*
*****
** Name          : doHtml
** Description   :
**              HTML processing page entry
point
** Parameters   :
**              txn handle
** Returns      :
**              int - return code
** Comments     :
*****
*/

void doHtml(TXN_HANDLE *txnHandle)
{
    DEBUGMSG("Entered doHtml(), parsing query string:"<<
txnHandle->urlString <<" into command block"<< endl);
    htmlPhraser      commandBlock(txnHandle->urlString);
    DEBUGMSG("Query string parsed. command:"<<
commandBlock.getCommandId() <<" user's terminal id:" <<
commandBlock.get_TERM_ID() <<endl);

    int terminalID = atoi(commandBlock.get_TERM_ID());
    int commandID = commandBlock.getCommandId();

    DEBUGMSG("User sent in a terimal id:"<<terminalID<<" , checking
to see if user has logged in before"<<endl);
    if(terminalID > 0)
    {
        DEBUGMSG("Terminal id > 0, user has logged in
already, terminalID:"<<terminalID<<" retrieving warehouse district
pair"<<endl);
        if(getTerminal(terminalID,txnHandle) != OK)
            return;
        DEBUGMSG("User had valid terminal id, user's login
warehouse:"<<txnHandle->w_id<<" district:"<<txnHandle->d_id<<endl);
    }
    else
    {

```

```

        DEBUGMSG("User did not submit a terminal id or valid
terminal id, ensure that the user is trying to log in."<<endl);
        if (commandID != TXN_LOGIN) && (commandID !=
TXN_LOGIN_RESULTS) )
        {
            DEBUGMSG("ERROR : User has not logged
in."<<endl);
            ERRORMSG("ERROR : User has not logged
in."<<endl);
            sprintf(txnHandle->htmlPage,"ERROR: User
has not logged in or did not submit a valid terminal.");
            return;
        }
        DEBUGMSG("User is in process of logging in,
commandID:"<<commandID<<endl);
    }

    DEBUGMSG("Calling html page
function:"<<commandBlock.getCommandId()<<endl);
    int rc =
htmlPageFunctions[commandBlock.getCommandId()](&commandBlock,txnHa
ndle);
    DEBUGMSG("Return from html page
function:"<<commandBlock.getCommandId()<<endl);

    return;
}

/*
*****
** Name          : getTerminal
** Description   :
**              retrieves terminal information
based on terminal id
** Parameters   :
**              int
terminal id
** Returns      :
**              TERM_HANDLE* txn handle
** Comments     :
**              int - return code
*****
*/

int getTerminal(int terminal,TXN_HANDLE *txnHandle)
{
    //check to see if terminal id is out of range
    if(terminal >= numUsers)
    {
        //terminal id not valid.
        sprintf(txnHandle->htmlPage,"ERROR: Client does not
support more than %d users, terminal id:%d",numUsers,terminal);
        ERRORMSG("ERROR : Client does not support more
than "<<numUsers<<" users, terminal id:"<<terminal<<endl);
        return ERR;
    }

    //check if terminal id is points to a not in use terminal
    if(!(termArray+terminal)->terminalInUse)
    {
        sprintf(txnHandle->htmlPage,"ERROR: Terminal id
given points to a not in use terminal.");
        ERRORMSG("ERROR : Terminal id given points to a
not in use terminal."<<endl);
        return ERR;
    }
}

```

```

        DEBUGMSG("Storing terminal warehouse, district , and initial term
id for user:"<<terminal<<endl);

        //assign terminal values to txn_handle
txnHandle->d_id = termArray[terminal].d_id;
txnHandle->w_id = termArray[terminal].w_id;
txnHandle->term_id = termArray[terminal].terminalID;

        DEBUGMSG("Users terminal:"<<terminal<< ", stored
warehouse:"<<txnHandle->w_id<<
                " district:"<<txnHandle->d_id<<" terminalID
stored:"<<txnHandle->term_id<<endl);

        return OK;
}

/*
*****
** Name          : assignTerminal
** Description   :
**              : assigns terminal index to user
** Parameters    :
**              : TERM_HANDLE* txn handle
** Returns      :
**              : int - return code
** Comments     :
**
*****
*/
int assignTerminal(TXN_HANDLE *txnHandle)
{
    EnterCriticalSection(&termLock);

    //check if terminal array is full.
    if(termNextFree == numUsers)
    {
        LeaveCriticalSection(&termLock);
        return ERR;
    }

    DEBUGMSG("Storing user warehouse:"<<txnHandle->w_id<<"
district:"<< txnHandle->d_id<<
                " in terminal slot:"<<termNextFree<<endl);

    //store users w_id and d_id
    termArray[termNextFree].d_id = txnHandle->d_id;
    termArray[termNextFree].w_id = txnHandle->w_id;

    //set terminal slot to be in use
    termArray[termNextFree].terminalInUse = true;
    termArray[termNextFree].terminalID = termNextFree;
    //in txn handle, set the terminal id
    txnHandle->term_id = termNextFree;

    //increment to next free terminal.
    termNextFree++;

    DEBUGMSG("User warehouse:"<<txnHandle->w_id<<"
district:"<< txnHandle->d_id <<
                " stored in terminal slot:"<<txnHandle->term_id<<" next
terminal free:"<<termNextFree<<endl);

    LeaveCriticalSection(&termLock);

    return OK;
}

```

TpccIsapi.def

; tpccIsapi.def : declares the module parameters for the DLL.

```

LIBRARY "tpccIsapi"

EXPORTS
    HttpExtensionProc
    GetExtensionVersion
    TerminateExtension

```

TpccIsapi.hpp

```

/*
*****
** Project       : AIX
** Component    : Performance/TPC-W Benchmark
** Name         : tpccIsapi.hpp
** Title        : ISAPI interface for tpcc
*****
** Copyright (c) 2001,2002 IBM Corporation
** All rights reserved
*****
** History      :
**              : Developed at IBM Austin by the AIX RS/6000
**              : performance group.
**
** Comments    :
**
*****
*/

#ifdef __tpccISAPI_hpp__
#define __tpccISAPI_hpp__

#include <windows.h>
#include <httpext.h>

#include <tpcc.h>
#include <htmlPhraser.h>
#include <iomanip>

#include <db2tpcc.h>
#include <comsvcs.h>

////////////////////////////////////
// Terminal struct
////////////////////////////////////
struct TERM_ENTRY
{
    int          terminalID;
    bool        terminalInUse;
    int         w_id;
    short       d_id;
};

////////////////////////////////////
// COM interface
////////////////////////////////////
struct COM_HANDLE
{
    Itpcc_com *comHandle;
    char      *txnBuffer;
    int       size;
};

```



```

////////////////////////////////////
// TXN handle
////////////////////////////////////
struct TXN_HANDLE
{
    char    htmlPage[MAX_HTML_PAGE_LEN];
    char    htmlHeader[MAX_HTML_HEADER_LEN];
    char    *urlString;

    //user data
    int     w_id;
    int     d_id;
    int     sync_id;
    int     term_id;
    int     conn_id;

    COM_HANDLE    comInterface;
};

struct DLVYQUEUEDATA
{
    int     warehouse;
    short   in_s_0_CARRIER_ID;
    struct _timeb enqueueTime;
};

////////////////////////////////////
// Definitions
////////////////////////////////////
#define INVALID_ITEM          100
#define HEADER
"Content-Type:text/html\r\nContent-Length: %d\r\nConnection:
Keep-Alive\r\n\r\n"
#define TLS_NULL
0xFFFFFFFF
#define ACCESS_TIMEOUT        3600000
//One hour in milli
seconds

#define DELIVERY_LOG_SUCCESS_STR    "--Tran %d
Queue %d.%03d Start %d.%03d\r\nW_ID: %d CARRIER_ID: %d
%s\r\nend-time: %d.%03d\r\n"

////////////////////////////////////
// Function Prototypes
////////////////////////////////////

int initDlvy();
int initTxnHandle(TXN_HANDLE **txnHandle);
int closeTxnHandle(TXN_HANDLE *txnHandle);
int readRegistryValues();
int getTerminal(int terminal,TXN_HANDLE *txnHandle);
int assignTerminal(TXN_HANDLE *txnHandle);
int getDBInstance();

void doHtml(TXN_HANDLE *txnHandle);
int doLoginForm(htmlPhraser *commandBlock,TXN_HANDLE *txnHandle);
int doLoginResults(htmlPhraser *commandBlock,TXN_HANDLE
*txnHandle);
int doNewOrderForm(htmlPhraser *commandBlock,TXN_HANDLE
*txnHandle);
int doNewOrderResults(htmlPhraser *commandBlock,TXN_HANDLE
*txnHandle);
int doPaymentForm(htmlPhraser *commandBlock,TXN_HANDLE
*txnHandle);
int doPaymentResults(htmlPhraser *commandBlock,TXN_HANDLE
*txnHandle);

```

```

int doOrderStatusForm(htmlPhraser *commandBlock,TXN_HANDLE
*txnHandle);
int doOrderStatusResults(htmlPhraser *commandBlock,TXN_HANDLE
*txnHandle);
int doDeliveryForm(htmlPhraser *commandBlock,TXN_HANDLE
*txnHandle);
int doDeliveryResults(htmlPhraser *commandBlock,TXN_HANDLE
*txnHandle);
int doStockForm(htmlPhraser *commandBlock,TXN_HANDLE *txnHandle);
int doStockResults(htmlPhraser *commandBlock,TXN_HANDLE *txnHandle);
int doExit(htmlPhraser *commandBlock,TXN_HANDLE *txnHandle);

int doLoginErrorPage(char *htmlPage,char *message);
int doNewOrderErrorPage(char *htmlPage,char *message,htmlPhraser
*commandBlock, TXN_HANDLE *txnHandle);
int doPaymentErrorPage(char *htmlPage,char *message,htmlPhraser
*commandBlock, TXN_HANDLE *txnHandle);
int doOrderStatusErrorPage(char *htmlPage,char *message,htmlPhraser
*commandBlock, TXN_HANDLE *txnHandle);
int doDeliveryErrorPage(char *htmlPage,char *message,htmlPhraser
*commandBlock, TXN_HANDLE *txnHandle);
int doStockErrorPage(char *htmlPage,char *message,htmlPhraser
*commandBlock, TXN_HANDLE *txnHandle);

void dlvyThreadEntry(void *);
int queueDlvyTxn(int warehouse, short carrier_id);

int appendButtons(char *htmlPage);
int appendItems(char *htmlPage,short itemCount,short cmdIDStart);
int appendHiddenFields(char *htmlPage,TXN_HANDLE *txnHandle);

int displayStatus(char *htmlPage,int rc);

#endif

```

TpccIsapi.rc

```

// Microsoft Visual C++ generated resource script.
//
#include "resource.h"

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

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

#ifdef APSTUDIO_INVOKED
////////////////////////////////////
//
// TEXTINCLUDE
//
1 TEXTINCLUDE
BEGIN
    "resource.h\0"
END

2 TEXTINCLUDE
BEGIN
    "#include ""winres.h""\r\n"
    "#include ""atlsrvres.h""\r\n"

```

```

"\0"
END

3 TEXTINCLUDE
BEGIN

    "LANGUAGE LANG_ENGLISH,
    SUBLANG_ENGLISH_US\r\n"
    "#pragma code_page(1252)\r\n"
    "#include \"\"atlsrv.rc\"\"\r\n"
    "\0"

END

#endif // APSTUDIO_INVOKED

////////////////////////////////////
//
// 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 "040904e4"
        BEGIN
            VALUE "CompanyName", "TODO:
<Company name>"
            VALUE "FileDescription", "TODO: <File description>"
            VALUE "FileVersion", "1.0.0.1"
            VALUE "InternalName", "isapi.dll"
            VALUE "LegalCopyright", "TODO: (c) <Company name>. All rights
reserved."
            VALUE "OriginalFilename", "isapi.dll"
            VALUE "ProductName", "TODO: <Product name>"
            VALUE "ProductVersion", "1.0.0.1"
            VALUE "OLESelfRegister", ""
        END
    END
    BLOCK "VarFileInfo"
    BEGIN
        VALUE "Translation", 0x0409, 1252
    END
END

LANGUAGE 9, 1
#pragma code_page(1252)
////////////////////////////////////
//
// String Table
//

STRINGTABLE

```

```

BEGIN
    IDS_PROJNAME        "tpccIsapi"
END

////////////////////////////////////
#ifdef APSTUDIO_INVOKED
////////////////////////////////////
//
// Generated from the TEXTINCLUDE 3 resource.
//
LANGUAGE LANG_ENGLISH, SUBLANG_ENGLISH_US
#pragma code_page(1252)
#include "atlsrv.rc"

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

```

Appendix B: Database Design Scripts

create_tablespace.ddl

connect to tpcc;

-- WAR

create regular tablespace WAR_001 PAGESIZE 4096 managed by database using (

DEVICE 'C:\Containers\WAR\001' 128,
DEVICE 'C:\Containers\WAR\026' 128

) EXTENTSIZE 16 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace WAR_002 PAGESIZE 4096 managed by database using (

DEVICE 'C:\Containers\WAR\002' 128,
DEVICE 'C:\Containers\WAR\027' 128

) EXTENTSIZE 16 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace WAR_003 PAGESIZE 4096 managed by database using (

DEVICE 'C:\Containers\WAR\003' 128,
DEVICE 'C:\Containers\WAR\028' 128

) EXTENTSIZE 16 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace WAR_004 PAGESIZE 4096 managed by database using (

DEVICE 'C:\Containers\WAR\004' 128,
DEVICE 'C:\Containers\WAR\029' 128

) EXTENTSIZE 16 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace WAR_005 PAGESIZE 4096 managed by database using (

DEVICE 'C:\Containers\WAR\005' 128,
DEVICE 'C:\Containers\WAR\030' 128

) EXTENTSIZE 16 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace WAR_006 PAGESIZE 4096 managed by database using (

DEVICE 'C:\Containers\WAR\006' 128,
DEVICE 'C:\Containers\WAR\031' 128

) EXTENTSIZE 16 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace WAR_007 PAGESIZE 4096 managed by database using (

DEVICE 'C:\Containers\WAR\013' 128,
DEVICE 'C:\Containers\WAR\038' 128

) EXTENTSIZE 16 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace WAR_008 PAGESIZE 4096 managed by database using (

DEVICE 'C:\Containers\WAR\014' 128,
DEVICE 'C:\Containers\WAR\039' 128

) EXTENTSIZE 16 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace WAR_009 PAGESIZE 4096 managed by database using (

DEVICE 'C:\Containers\WAR\015' 128,
DEVICE 'C:\Containers\WAR\040' 128

) EXTENTSIZE 16 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace WAR_010 PAGESIZE 4096 managed by database using (

DEVICE 'C:\Containers\WAR\016' 128,

DEVICE 'C:\Containers\WAR\041' 128

) EXTENTSIZE 16 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace WAR_011 PAGESIZE 4096 managed by database using (

DEVICE 'C:\Containers\WAR\017' 128,
DEVICE 'C:\Containers\WAR\042' 128

) EXTENTSIZE 16 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace WAR_012 PAGESIZE 4096 managed by database using (

DEVICE 'C:\Containers\WAR\018' 128,
DEVICE 'C:\Containers\WAR\043' 128

) EXTENTSIZE 16 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace WAR_013 PAGESIZE 4096 managed by database using (

DEVICE 'C:\Containers\WAR\044' 128,
DEVICE 'C:\Containers\WAR\019' 128

) EXTENTSIZE 16 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace WAR_014 PAGESIZE 4096 managed by database using (

DEVICE 'C:\Containers\WAR\045' 128,
DEVICE 'C:\Containers\WAR\020' 128

) EXTENTSIZE 16 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace WAR_015 PAGESIZE 4096 managed by database using (

DEVICE 'C:\Containers\WAR\046' 128,
DEVICE 'C:\Containers\WAR\021' 128

) EXTENTSIZE 16 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace WAR_016 PAGESIZE 4096 managed by database using (

DEVICE 'C:\Containers\WAR\047' 128,
DEVICE 'C:\Containers\WAR\022' 128

) EXTENTSIZE 16 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace WAR_017 PAGESIZE 4096 managed by database using (

DEVICE 'C:\Containers\WAR\048' 128,
DEVICE 'C:\Containers\WAR\023' 128

) EXTENTSIZE 16 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace WAR_018 PAGESIZE 4096 managed by database using (

DEVICE 'C:\Containers\WAR\049' 128,
DEVICE 'C:\Containers\WAR\024' 128

) EXTENTSIZE 16 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace WAR_019 PAGESIZE 4096 managed by database using (

DEVICE 'C:\Containers\WAR\032' 128,
DEVICE 'C:\Containers\WAR\007' 128

) EXTENTSIZE 16 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace WAR_020 PAGESIZE 4096 managed by database using (

DEVICE 'C:\Containers\WAR\033' 128,
DEVICE 'C:\Containers\WAR\008' 128

) EXTENTSIZE 16 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace WAR_021 PAGESIZE 4096 managed by database using (

DEVICE 'C:\Containers\WAR\034' 128,
DEVICE 'C:\Containers\WAR\009' 128

) EXTENTSIZE 16 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;


```

DEVICE 'C:\Containers\DIS\032' 272,
DEVICE 'C:\Containers\DIS\007' 272
) EXTENTSIZE 32 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace DIS_020 PAGESIZE 4096 managed by database using
(
DEVICE 'C:\Containers\DIS\033' 272,
DEVICE 'C:\Containers\DIS\008' 272
) EXTENTSIZE 32 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace DIS_021 PAGESIZE 4096 managed by database using
(
DEVICE 'C:\Containers\DIS\034' 272,
DEVICE 'C:\Containers\DIS\009' 272
) EXTENTSIZE 32 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace DIS_022 PAGESIZE 4096 managed by database using
(
DEVICE 'C:\Containers\DIS\035' 272,
DEVICE 'C:\Containers\DIS\010' 272
) EXTENTSIZE 32 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace DIS_023 PAGESIZE 4096 managed by database using
(
DEVICE 'C:\Containers\DIS\036' 272,
DEVICE 'C:\Containers\DIS\011' 272
) EXTENTSIZE 32 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace DIS_024 PAGESIZE 4096 managed by database using
(
DEVICE 'C:\Containers\DIS\037' 272,
DEVICE 'C:\Containers\DIS\012' 272
) EXTENTSIZE 32 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace DIS_025 PAGESIZE 4096 managed by database using
(
DEVICE 'C:\Containers\DIS\050' 272,
DEVICE 'C:\Containers\DIS\025' 272
) EXTENTSIZE 32 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

-- ITM
create regular tablespace ITM PAGESIZE 8192 managed by database using (
DEVICE 'C:\Containers\ITM\001' 128,
DEVICE 'C:\Containers\ITM\002' 128,
DEVICE 'C:\Containers\ITM\003' 128,
DEVICE 'C:\Containers\ITM\004' 128,
DEVICE 'C:\Containers\ITM\005' 128,
DEVICE 'C:\Containers\ITM\006' 128,
DEVICE 'C:\Containers\ITM\007' 128,
DEVICE 'C:\Containers\ITM\008' 128,
DEVICE 'C:\Containers\ITM\009' 128,
DEVICE 'C:\Containers\ITM\010' 128,
DEVICE 'C:\Containers\ITM\011' 128,
DEVICE 'C:\Containers\ITM\012' 128,
DEVICE 'C:\Containers\ITM\013' 128,
DEVICE 'C:\Containers\ITM\014' 128,
DEVICE 'C:\Containers\ITM\015' 128,
DEVICE 'C:\Containers\ITM\016' 128,
DEVICE 'C:\Containers\ITM\017' 128,
DEVICE 'C:\Containers\ITM\018' 128,
DEVICE 'C:\Containers\ITM\019' 128,
DEVICE 'C:\Containers\ITM\020' 128,
DEVICE 'C:\Containers\ITM\021' 128,
DEVICE 'C:\Containers\ITM\022' 128,
DEVICE 'C:\Containers\ITM\023' 128,
DEVICE 'C:\Containers\ITM\024' 128,
DEVICE 'C:\Containers\ITM\025' 128,
DEVICE 'C:\Containers\ITM\026' 128,
DEVICE 'C:\Containers\ITM\027' 128,
DEVICE 'C:\Containers\ITM\028' 128,
DEVICE 'C:\Containers\ITM\029' 128,
DEVICE 'C:\Containers\ITM\030' 128,
DEVICE 'C:\Containers\ITM\031' 128,
DEVICE 'C:\Containers\ITM\032' 128,
DEVICE 'C:\Containers\ITM\033' 128,
DEVICE 'C:\Containers\ITM\034' 128,
DEVICE 'C:\Containers\ITM\035' 128,
DEVICE 'C:\Containers\ITM\036' 128,
DEVICE 'C:\Containers\ITM\037' 128,
DEVICE 'C:\Containers\ITM\038' 128,
DEVICE 'C:\Containers\ITM\039' 128,
DEVICE 'C:\Containers\ITM\040' 128,
DEVICE 'C:\Containers\ITM\041' 128,
DEVICE 'C:\Containers\ITM\042' 128,
DEVICE 'C:\Containers\ITM\043' 128,
DEVICE 'C:\Containers\ITM\044' 128,
DEVICE 'C:\Containers\ITM\045' 128,
DEVICE 'C:\Containers\ITM\046' 128,
DEVICE 'C:\Containers\ITM\047' 128,
DEVICE 'C:\Containers\ITM\048' 128,
DEVICE 'C:\Containers\ITM\049' 128,
DEVICE 'C:\Containers\ITM\050' 128
) EXTENTSIZE 16 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULT8K;

-- STK
create regular tablespace STK_001 PAGESIZE 4096 managed by database
using (
DEVICE 'C:\Containers\STK\001' 3937920,
DEVICE 'C:\Containers\STK\026' 3937920
) EXTENTSIZE 256 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace STK_002 PAGESIZE 4096 managed by database
using (
DEVICE 'C:\Containers\STK\002' 3937920,
DEVICE 'C:\Containers\STK\027' 3937920
) EXTENTSIZE 256 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace STK_003 PAGESIZE 4096 managed by database
using (
DEVICE 'C:\Containers\STK\003' 3937920,
DEVICE 'C:\Containers\STK\028' 3937920
) EXTENTSIZE 256 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace STK_004 PAGESIZE 4096 managed by database
using (
DEVICE 'C:\Containers\STK\004' 3937920,
DEVICE 'C:\Containers\STK\029' 3937920
) EXTENTSIZE 256 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace STK_005 PAGESIZE 4096 managed by database
using (
DEVICE 'C:\Containers\STK\005' 3937920,
DEVICE 'C:\Containers\STK\030' 3937920
) EXTENTSIZE 256 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace STK_006 PAGESIZE 4096 managed by database
using (
DEVICE 'C:\Containers\STK\006' 3937920,
DEVICE 'C:\Containers\STK\031' 3937920
) EXTENTSIZE 256 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace STK_007 PAGESIZE 4096 managed by database
using (
DEVICE 'C:\Containers\STK\013' 3937920,
DEVICE 'C:\Containers\STK\038' 3937920
) EXTENTSIZE 256 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

```



```

create regular tablespace NEWB_010 PAGESIZE 4096 managed by database
using (
DEVICE 'C:\Containers\NEWB\016' 42112,
DEVICE 'C:\Containers\NEWB\041' 42112
) EXTENTSIZE 256 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace NEWB_011 PAGESIZE 4096 managed by database
using (
DEVICE 'C:\Containers\NEWB\017' 42112,
DEVICE 'C:\Containers\NEWB\042' 42112
) EXTENTSIZE 256 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace NEWB_012 PAGESIZE 4096 managed by database
using (
DEVICE 'C:\Containers\NEWB\018' 42112,
DEVICE 'C:\Containers\NEWB\043' 42112
) EXTENTSIZE 256 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace NEWB_013 PAGESIZE 4096 managed by database
using (
DEVICE 'C:\Containers\NEWB\044' 42112,
DEVICE 'C:\Containers\NEWB\019' 42112
) EXTENTSIZE 256 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace NEWB_014 PAGESIZE 4096 managed by database
using (
DEVICE 'C:\Containers\NEWB\045' 42112,
DEVICE 'C:\Containers\NEWB\020' 42112
) EXTENTSIZE 256 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace NEWB_015 PAGESIZE 4096 managed by database
using (
DEVICE 'C:\Containers\NEWB\046' 42112,
DEVICE 'C:\Containers\NEWB\021' 42112
) EXTENTSIZE 256 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace NEWB_016 PAGESIZE 4096 managed by database
using (
DEVICE 'C:\Containers\NEWB\047' 42112,
DEVICE 'C:\Containers\NEWB\022' 42112
) EXTENTSIZE 256 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace NEWB_017 PAGESIZE 4096 managed by database
using (
DEVICE 'C:\Containers\NEWB\048' 42112,
DEVICE 'C:\Containers\NEWB\023' 42112
) EXTENTSIZE 256 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace NEWB_018 PAGESIZE 4096 managed by database
using (
DEVICE 'C:\Containers\NEWB\049' 42112,
DEVICE 'C:\Containers\NEWB\024' 42112
) EXTENTSIZE 256 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace NEWB_019 PAGESIZE 4096 managed by database
using (
DEVICE 'C:\Containers\NEWB\032' 42112,
DEVICE 'C:\Containers\NEWB\007' 42112
) EXTENTSIZE 256 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace NEWB_020 PAGESIZE 4096 managed by database
using (
DEVICE 'C:\Containers\NEWB\033' 42112,
DEVICE 'C:\Containers\NEWB\008' 42112
) EXTENTSIZE 256 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

```

```

create regular tablespace NEWB_021 PAGESIZE 4096 managed by database
using (
DEVICE 'C:\Containers\NEWB\034' 42112,
DEVICE 'C:\Containers\NEWB\009' 42112
) EXTENTSIZE 256 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace NEWB_022 PAGESIZE 4096 managed by database
using (
DEVICE 'C:\Containers\NEWB\035' 42112,
DEVICE 'C:\Containers\NEWB\010' 42112
) EXTENTSIZE 256 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace NEWB_023 PAGESIZE 4096 managed by database
using (
DEVICE 'C:\Containers\NEWB\036' 42112,
DEVICE 'C:\Containers\NEWB\011' 42112
) EXTENTSIZE 256 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace NEWB_024 PAGESIZE 4096 managed by database
using (
DEVICE 'C:\Containers\NEWB\037' 42112,
DEVICE 'C:\Containers\NEWB\012' 42112
) EXTENTSIZE 256 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

create regular tablespace NEWB_025 PAGESIZE 4096 managed by database
using (
DEVICE 'C:\Containers\NEWB\050' 42112,
DEVICE 'C:\Containers\NEWB\025' 42112
) EXTENTSIZE 256 PREFETCHSIZE 0 BUFFERPOOL IBMDEFAULTBP;

connect reset;

```

alter_tablespace.ddl

```

-----
-- Licensed Materials - Property of IBM
--
-- Governed under the terms of the International
-- License Agreement for Non-Warranted Sample Code.
--
-- (C) COPYRIGHT International Business Machines Corp. 1996 - 2002
-- All Rights Reserved.
--
-- US Government Users Restricted Rights - Use, duplication or
-- disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
-----

-- Set Bufferpools For Tablespaces
connect to tpcc;
alter tablespace WAR_001 bufferpool WAR1;
alter tablespace WAR_002 bufferpool WAR1;
alter tablespace WAR_003 bufferpool WAR1;
alter tablespace WAR_004 bufferpool WAR1;
alter tablespace WAR_005 bufferpool WAR1;
alter tablespace WAR_006 bufferpool WAR1;
alter tablespace WAR_007 bufferpool WAR1;
alter tablespace WAR_008 bufferpool WAR1;
alter tablespace WAR_009 bufferpool WAR1;
alter tablespace WAR_010 bufferpool WAR1;
alter tablespace WAR_011 bufferpool WAR1;
alter tablespace WAR_012 bufferpool WAR1;
alter tablespace WAR_013 bufferpool WAR2;
alter tablespace WAR_014 bufferpool WAR2;
alter tablespace WAR_015 bufferpool WAR2;
alter tablespace WAR_016 bufferpool WAR2;
alter tablespace WAR_017 bufferpool WAR2;
alter tablespace WAR_018 bufferpool WAR2;

```



```

alter tablespace ORDI_004 bufferpool ORD_I1;
alter tablespace ORDI_005 bufferpool ORD_I1;
alter tablespace ORDI_006 bufferpool ORD_I1;
alter tablespace ORDI_007 bufferpool ORD_I1;
alter tablespace ORDI_008 bufferpool ORD_I1;
alter tablespace ORDI_009 bufferpool ORD_I1;
alter tablespace ORDI_010 bufferpool ORD_I1;
alter tablespace ORDI_011 bufferpool ORD_I1;
alter tablespace ORDI_012 bufferpool ORD_I1;
alter tablespace ORDI_013 bufferpool ORD_I2;
alter tablespace ORDI_014 bufferpool ORD_I2;
alter tablespace ORDI_015 bufferpool ORD_I2;
alter tablespace ORDI_016 bufferpool ORD_I2;
alter tablespace ORDI_017 bufferpool ORD_I2;
alter tablespace ORDI_018 bufferpool ORD_I2;
alter tablespace ORDI_019 bufferpool ORD_I2;
alter tablespace ORDI_020 bufferpool ORD_I2;
alter tablespace ORDI_021 bufferpool ORD_I2;
alter tablespace ORDI_022 bufferpool ORD_I2;
alter tablespace ORDI_023 bufferpool ORD_I2;
alter tablespace ORDI_024 bufferpool ORD_I2;
alter tablespace ORDI_025 bufferpool ORD_I3;
alter tablespace OLN_001 bufferpool OLN1;
alter tablespace OLN_002 bufferpool OLN1;
alter tablespace OLN_003 bufferpool OLN1;
alter tablespace OLN_004 bufferpool OLN1;
alter tablespace OLN_005 bufferpool OLN1;
alter tablespace OLN_006 bufferpool OLN1;
alter tablespace OLN_007 bufferpool OLN1;
alter tablespace OLN_008 bufferpool OLN1;
alter tablespace OLN_009 bufferpool OLN1;
alter tablespace OLN_010 bufferpool OLN1;
alter tablespace OLN_011 bufferpool OLN1;
alter tablespace OLN_012 bufferpool OLN1;
alter tablespace OLN_013 bufferpool OLN2;
alter tablespace OLN_014 bufferpool OLN2;
alter tablespace OLN_015 bufferpool OLN2;
alter tablespace OLN_016 bufferpool OLN2;
alter tablespace OLN_017 bufferpool OLN2;
alter tablespace OLN_018 bufferpool OLN2;
alter tablespace OLN_019 bufferpool OLN2;
alter tablespace OLN_020 bufferpool OLN2;
alter tablespace OLN_021 bufferpool OLN2;
alter tablespace OLN_022 bufferpool OLN2;
alter tablespace OLN_023 bufferpool OLN2;
alter tablespace OLN_024 bufferpool OLN2;
alter tablespace OLN_025 bufferpool OLN3;
alter tablespace NEWA_001 bufferpool NEW1;
alter tablespace NEWA_002 bufferpool NEW1;
alter tablespace NEWA_003 bufferpool NEW1;
alter tablespace NEWA_004 bufferpool NEW1;
alter tablespace NEWA_005 bufferpool NEW1;
alter tablespace NEWA_006 bufferpool NEW1;
alter tablespace NEWA_007 bufferpool NEW1;
alter tablespace NEWA_008 bufferpool NEW1;
alter tablespace NEWA_009 bufferpool NEW1;
alter tablespace NEWA_010 bufferpool NEW1;
alter tablespace NEWA_011 bufferpool NEW1;
alter tablespace NEWA_012 bufferpool NEW1;
alter tablespace NEWA_013 bufferpool NEW2;
alter tablespace NEWA_014 bufferpool NEW2;
alter tablespace NEWA_015 bufferpool NEW2;
alter tablespace NEWA_016 bufferpool NEW2;
alter tablespace NEWA_017 bufferpool NEW2;
alter tablespace NEWA_018 bufferpool NEW2;
alter tablespace NEWA_019 bufferpool NEW2;
alter tablespace NEWA_020 bufferpool NEW2;
alter tablespace NEWA_021 bufferpool NEW2;

```

```

alter tablespace NEWA_022 bufferpool NEW2;
alter tablespace NEWA_023 bufferpool NEW2;
alter tablespace NEWA_024 bufferpool NEW2;
alter tablespace NEWA_025 bufferpool NEW3;
alter tablespace NEWB_001 bufferpool NEW1;
alter tablespace NEWB_002 bufferpool NEW1;
alter tablespace NEWB_003 bufferpool NEW1;
alter tablespace NEWB_004 bufferpool NEW1;
alter tablespace NEWB_005 bufferpool NEW1;
alter tablespace NEWB_006 bufferpool NEW1;
alter tablespace NEWB_007 bufferpool NEW1;
alter tablespace NEWB_008 bufferpool NEW1;
alter tablespace NEWB_009 bufferpool NEW1;
alter tablespace NEWB_010 bufferpool NEW1;
alter tablespace NEWB_011 bufferpool NEW1;
alter tablespace NEWB_012 bufferpool NEW1;
alter tablespace NEWB_013 bufferpool NEW2;
alter tablespace NEWB_014 bufferpool NEW2;
alter tablespace NEWB_015 bufferpool NEW2;
alter tablespace NEWB_016 bufferpool NEW2;
alter tablespace NEWB_017 bufferpool NEW2;
alter tablespace NEWB_018 bufferpool NEW2;
alter tablespace NEWB_019 bufferpool NEW2;
alter tablespace NEWB_020 bufferpool NEW2;
alter tablespace NEWB_021 bufferpool NEW2;
alter tablespace NEWB_022 bufferpool NEW2;
alter tablespace NEWB_023 bufferpool NEW2;
alter tablespace NEWB_024 bufferpool NEW2;
alter tablespace NEWB_025 bufferpool NEW3;
alter tablespace HST_001 bufferpool HST1;
alter tablespace HST_002 bufferpool HST1;
alter tablespace HST_003 bufferpool HST1;
alter tablespace HST_004 bufferpool HST1;
alter tablespace HST_005 bufferpool HST1;
alter tablespace HST_006 bufferpool HST1;
alter tablespace HST_007 bufferpool HST1;
alter tablespace HST_008 bufferpool HST1;
alter tablespace HST_009 bufferpool HST1;
alter tablespace HST_010 bufferpool HST1;
alter tablespace HST_011 bufferpool HST1;
alter tablespace HST_012 bufferpool HST1;
alter tablespace HST_013 bufferpool HST2;
alter tablespace HST_014 bufferpool HST2;
alter tablespace HST_015 bufferpool HST2;
alter tablespace HST_016 bufferpool HST2;
alter tablespace HST_017 bufferpool HST2;
alter tablespace HST_018 bufferpool HST2;
alter tablespace HST_019 bufferpool HST2;
alter tablespace HST_020 bufferpool HST2;
alter tablespace HST_021 bufferpool HST2;
alter tablespace HST_022 bufferpool HST2;
alter tablespace HST_023 bufferpool HST2;
alter tablespace HST_024 bufferpool HST2;
alter tablespace HST_025 bufferpool HST3;

```

```

connect reset;
terminate;

```

alter_bufferpool.ddl

```

-----
-- Licensed Materials - Property of IBM
--
-- Governed under the terms of the International
-- License Agreement for Non-Warranted Sample Code.
--
-- (C) COPYRIGHT International Business Machines Corp. 1996 - 2002

```

```
-- All Rights Reserved.
--
-- US Government Users Restricted Rights - Use, duplication or
-- disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
```

```
-----
-- Alter Size of Bufferpools
connect to tpcc;
```

```
alter bufferpool IBMDEFAULTBP deferred size 256;
alter bufferpool IBMDEFAULT8K deferred size 16;
alter bufferpool IBMDEFAULT16K deferred size 16;
alter bufferpool WAR1 deferred size 300;
alter bufferpool DIS1 deferred size 3200;
alter bufferpool ITM deferred size 1235;
alter bufferpool HST1 deferred size 624;
alter bufferpool NEW1 deferred size 193900;
alter bufferpool ORD1 deferred size 106200;
alter bufferpool CST1 deferred size 16160;
alter bufferpool STK1 deferred size 12600000;
alter bufferpool OLN1 deferred size 387800;
alter bufferpool CST_I1 deferred size 121200;
alter bufferpool ORD_I1 deferred size 282800;
alter bufferpool WAR2 deferred size 300;
alter bufferpool DIS2 deferred size 3200;
alter bufferpool HST2 deferred size 624;
alter bufferpool NEW2 deferred size 193900;
alter bufferpool ORD2 deferred size 106200;
alter bufferpool CST2 deferred size 16160;
alter bufferpool STK2 deferred size 12600000;
alter bufferpool OLN2 deferred size 387800;
alter bufferpool CST_I2 deferred size 121200;
alter bufferpool ORD_I2 deferred size 282800;
alter bufferpool WAR3 deferred size 26;
alter bufferpool DIS3 deferred size 270;
alter bufferpool HST3 deferred size 60;
alter bufferpool NEW3 deferred size 16320;
alter bufferpool ORD3 deferred size 8940;
alter bufferpool CST3 deferred size 1360;
alter bufferpool STK3 deferred size 930000;
alter bufferpool OLN3 deferred size 32640;
alter bufferpool CST_I3 deferred size 10200;
alter bufferpool ORD_I3 deferred size 23800;
connect reset;
terminate;
```

create_bufferpool.ddl

```
-----
-- Licensed Materials - Property of IBM
--
-- Governed under the terms of the International
-- License Agreement for Non-Warranted Sample Code.
--
-- (C) COPYRIGHT International Business Machines Corp. 1996 - 2002
-- All Rights Reserved.
--
-- US Government Users Restricted Rights - Use, duplication or
-- disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
```

```
-----
-- Create Bufferpools
connect to tpcc;
create bufferpool IBMDEFAULTBP size 1310720 pagesize 4096;
create bufferpool IBMDEFAULT8K size 655360 pagesize 8192;
create bufferpool IBMDEFAULT16K size 5000 pagesize 16384;
create bufferpool WAR1 size 1000 pagesize 4096;
create bufferpool DIS1 size 1000 pagesize 4096;
```

```
create bufferpool ITM size 1000 pagesize 8192;
create bufferpool HST1 size 1000 pagesize 16384;
create bufferpool NEW1 size 1000 pagesize 4096;
create bufferpool ORD1 size 1000 pagesize 8192;
create bufferpool CST1 size 1000 pagesize 4096;
create bufferpool STK1 size 1000 pagesize 4096;
create bufferpool OLN1 size 1000 pagesize 8192;
create bufferpool CST_I1 size 1000 pagesize 8192;
create bufferpool ORD_I1 size 1000 pagesize 8192;
create bufferpool WAR2 size 1000 pagesize 4096;
create bufferpool DIS2 size 1000 pagesize 4096;
create bufferpool HST2 size 1000 pagesize 16384;
create bufferpool NEW2 size 1000 pagesize 4096;
create bufferpool ORD2 size 1000 pagesize 8192;
create bufferpool CST2 size 1000 pagesize 4096;
create bufferpool STK2 size 1000 pagesize 4096;
create bufferpool OLN2 size 1000 pagesize 8192;
create bufferpool CST_I2 size 1000 pagesize 8192;
create bufferpool ORD_I2 size 1000 pagesize 8192;
create bufferpool WAR3 size 1000 pagesize 4096;
create bufferpool DIS3 size 1000 pagesize 4096;
create bufferpool HST3 size 1000 pagesize 16384;
create bufferpool NEW3 size 1000 pagesize 4096;
create bufferpool ORD3 size 1000 pagesize 8192;
create bufferpool CST3 size 1000 pagesize 4096;
create bufferpool STK3 size 1000 pagesize 4096;
create bufferpool OLN3 size 1000 pagesize 8192;
create bufferpool CST_I3 size 1000 pagesize 8192;
create bufferpool ORD_I3 size 1000 pagesize 8192;
connect reset;
terminate;
```

create_database.ddl

```
-----
-- Licensed Materials - Property of IBM
--
-- Governed under the terms of the International
-- License Agreement for Non-Warranted Sample Code.
--
-- (C) COPYRIGHT International Business Machines Corp. 1996 - 2002
-- All Rights Reserved.
--
-- US Government Users Restricted Rights - Use, duplication or
-- disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
```

```
-----
-- Create Database
drop database tpcc;
create database tpcc collate using identity;
```

alttbsp_pf_0.ddl

```
connect to TPCC;
alter tablespace CSTI_001 prefetchsize 0;
alter tablespace CSTI_002 prefetchsize 0;
alter tablespace CSTI_003 prefetchsize 0;
alter tablespace CSTI_004 prefetchsize 0;
alter tablespace CSTI_005 prefetchsize 0;
alter tablespace CSTI_006 prefetchsize 0;
alter tablespace CSTI_007 prefetchsize 0;
alter tablespace CSTI_008 prefetchsize 0;
alter tablespace CSTI_009 prefetchsize 0;
alter tablespace CSTI_010 prefetchsize 0;
alter tablespace CSTI_011 prefetchsize 0;
alter tablespace CSTI_012 prefetchsize 0;
```



```

alter tablespace WAR_016 prefetchsize 4096;
alter tablespace WAR_017 prefetchsize 4096;
alter tablespace WAR_018 prefetchsize 4096;
alter tablespace WAR_019 prefetchsize 4096;
alter tablespace WAR_020 prefetchsize 4096;
alter tablespace WAR_021 prefetchsize 4096;
alter tablespace WAR_022 prefetchsize 4096;
alter tablespace WAR_023 prefetchsize 4096;
alter tablespace WAR_024 prefetchsize 4096;
alter tablespace WAR_025 prefetchsize 4096;
connect reset;

```

crconst_customer.ddl

```

connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER1 OFF;
ALTER TABLE CUSTOMER1 DROP CONSTRAINT CUSTOMER1CKC;
ALTER TABLE CUSTOMER1 ADD CONSTRAINT CUSTOMER1CKC
CHECK (C_W_ID BETWEEN 1 AND 800);
SET INTEGRITY FOR CUSTOMER1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER2 OFF;
ALTER TABLE CUSTOMER2 DROP CONSTRAINT CUSTOMER2CKC;
ALTER TABLE CUSTOMER2 ADD CONSTRAINT CUSTOMER2CKC
CHECK (C_W_ID BETWEEN 801 AND 1600);
SET INTEGRITY FOR CUSTOMER2 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER3 OFF;
ALTER TABLE CUSTOMER3 DROP CONSTRAINT CUSTOMER3CKC;
ALTER TABLE CUSTOMER3 ADD CONSTRAINT CUSTOMER3CKC
CHECK (C_W_ID BETWEEN 1601 AND 2400);
SET INTEGRITY FOR CUSTOMER3 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER4 OFF;
ALTER TABLE CUSTOMER4 DROP CONSTRAINT CUSTOMER4CKC;
ALTER TABLE CUSTOMER4 ADD CONSTRAINT CUSTOMER4CKC
CHECK (C_W_ID BETWEEN 2401 AND 3200);
SET INTEGRITY FOR CUSTOMER4 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER5 OFF;
ALTER TABLE CUSTOMER5 DROP CONSTRAINT CUSTOMER5CKC;
ALTER TABLE CUSTOMER5 ADD CONSTRAINT CUSTOMER5CKC
CHECK (C_W_ID BETWEEN 3201 AND 4000);
SET INTEGRITY FOR CUSTOMER5 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER6 OFF;
ALTER TABLE CUSTOMER6 DROP CONSTRAINT CUSTOMER6CKC;
ALTER TABLE CUSTOMER6 ADD CONSTRAINT CUSTOMER6CKC
CHECK (C_W_ID BETWEEN 4001 AND 4800);
SET INTEGRITY FOR CUSTOMER6 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER7 OFF;
ALTER TABLE CUSTOMER7 DROP CONSTRAINT CUSTOMER7CKC;
ALTER TABLE CUSTOMER7 ADD CONSTRAINT CUSTOMER7CKC
CHECK (C_W_ID BETWEEN 4801 AND 5600);
SET INTEGRITY FOR CUSTOMER7 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER8 OFF;
ALTER TABLE CUSTOMER8 DROP CONSTRAINT CUSTOMER8CKC;

```

```

ALTER TABLE CUSTOMER8 ADD CONSTRAINT CUSTOMER8CKC
CHECK (C_W_ID BETWEEN 5601 AND 6400);
SET INTEGRITY FOR CUSTOMER8 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER9 OFF;
ALTER TABLE CUSTOMER9 DROP CONSTRAINT CUSTOMER9CKC;
ALTER TABLE CUSTOMER9 ADD CONSTRAINT CUSTOMER9CKC
CHECK (C_W_ID BETWEEN 6401 AND 7200);
SET INTEGRITY FOR CUSTOMER9 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER10 OFF;
ALTER TABLE CUSTOMER10 DROP CONSTRAINT CUSTOMER10CKC;
ALTER TABLE CUSTOMER10 ADD CONSTRAINT CUSTOMER10CKC
CHECK (C_W_ID BETWEEN 7201 AND 8000);
SET INTEGRITY FOR CUSTOMER10 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER11 OFF;
ALTER TABLE CUSTOMER11 DROP CONSTRAINT CUSTOMER11CKC;
ALTER TABLE CUSTOMER11 ADD CONSTRAINT CUSTOMER11CKC
CHECK (C_W_ID BETWEEN 8001 AND 8800);
SET INTEGRITY FOR CUSTOMER11 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER12 OFF;
ALTER TABLE CUSTOMER12 DROP CONSTRAINT CUSTOMER12CKC;
ALTER TABLE CUSTOMER12 ADD CONSTRAINT CUSTOMER12CKC
CHECK (C_W_ID BETWEEN 8801 AND 9600);
SET INTEGRITY FOR CUSTOMER12 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER13 OFF;
ALTER TABLE CUSTOMER13 DROP CONSTRAINT CUSTOMER13CKC;
ALTER TABLE CUSTOMER13 ADD CONSTRAINT CUSTOMER13CKC
CHECK (C_W_ID BETWEEN 9601 AND 10400);
SET INTEGRITY FOR CUSTOMER13 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER14 OFF;
ALTER TABLE CUSTOMER14 DROP CONSTRAINT CUSTOMER14CKC;
ALTER TABLE CUSTOMER14 ADD CONSTRAINT CUSTOMER14CKC
CHECK (C_W_ID BETWEEN 10401 AND 11200);
SET INTEGRITY FOR CUSTOMER14 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER15 OFF;
ALTER TABLE CUSTOMER15 DROP CONSTRAINT CUSTOMER15CKC;
ALTER TABLE CUSTOMER15 ADD CONSTRAINT CUSTOMER15CKC
CHECK (C_W_ID BETWEEN 11201 AND 12000);
SET INTEGRITY FOR CUSTOMER15 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER16 OFF;
ALTER TABLE CUSTOMER16 DROP CONSTRAINT CUSTOMER16CKC;
ALTER TABLE CUSTOMER16 ADD CONSTRAINT CUSTOMER16CKC
CHECK (C_W_ID BETWEEN 12001 AND 12800);
SET INTEGRITY FOR CUSTOMER16 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER17 OFF;
ALTER TABLE CUSTOMER17 DROP CONSTRAINT CUSTOMER17CKC;
ALTER TABLE CUSTOMER17 ADD CONSTRAINT CUSTOMER17CKC
CHECK (C_W_ID BETWEEN 12801 AND 13600);
SET INTEGRITY FOR CUSTOMER17 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;

```

```

SET INTEGRITY FOR CUSTOMER18 OFF;
ALTER TABLE CUSTOMER18 DROP CONSTRAINT CUSTOMER18CKC;
ALTER TABLE CUSTOMER18 ADD CONSTRAINT CUSTOMER18CKC
CHECK (C_W_ID BETWEEN 13601 AND 14400);
SET INTEGRITY FOR CUSTOMER18 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER19 OFF;
ALTER TABLE CUSTOMER19 DROP CONSTRAINT CUSTOMER19CKC;
ALTER TABLE CUSTOMER19 ADD CONSTRAINT CUSTOMER19CKC
CHECK (C_W_ID BETWEEN 14401 AND 15200);
SET INTEGRITY FOR CUSTOMER19 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER20 OFF;
ALTER TABLE CUSTOMER20 DROP CONSTRAINT CUSTOMER20CKC;
ALTER TABLE CUSTOMER20 ADD CONSTRAINT CUSTOMER20CKC
CHECK (C_W_ID BETWEEN 15201 AND 16000);
SET INTEGRITY FOR CUSTOMER20 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER21 OFF;
ALTER TABLE CUSTOMER21 DROP CONSTRAINT CUSTOMER21CKC;
ALTER TABLE CUSTOMER21 ADD CONSTRAINT CUSTOMER21CKC
CHECK (C_W_ID BETWEEN 16001 AND 16800);
SET INTEGRITY FOR CUSTOMER21 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER22 OFF;
ALTER TABLE CUSTOMER22 DROP CONSTRAINT CUSTOMER22CKC;
ALTER TABLE CUSTOMER22 ADD CONSTRAINT CUSTOMER22CKC
CHECK (C_W_ID BETWEEN 16801 AND 17600);
SET INTEGRITY FOR CUSTOMER22 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER23 OFF;
ALTER TABLE CUSTOMER23 DROP CONSTRAINT CUSTOMER23CKC;
ALTER TABLE CUSTOMER23 ADD CONSTRAINT CUSTOMER23CKC
CHECK (C_W_ID BETWEEN 17601 AND 18400);
SET INTEGRITY FOR CUSTOMER23 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER24 OFF;
ALTER TABLE CUSTOMER24 DROP CONSTRAINT CUSTOMER24CKC;
ALTER TABLE CUSTOMER24 ADD CONSTRAINT CUSTOMER24CKC
CHECK (C_W_ID BETWEEN 18401 AND 19200);
SET INTEGRITY FOR CUSTOMER24 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR CUSTOMER25 OFF;
ALTER TABLE CUSTOMER25 DROP CONSTRAINT CUSTOMER25CKC;
ALTER TABLE CUSTOMER25 ADD CONSTRAINT CUSTOMER25CKC
CHECK (C_W_ID >= 19201);
SET INTEGRITY FOR CUSTOMER25 ALL IMMEDIATE UNCHECKED;
connect reset;

```

crconst_district.ddl

```

connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT1 OFF;
ALTER TABLE DISTRICT1 DROP CONSTRAINT DISTRICT1CKC;
ALTER TABLE DISTRICT1 ADD CONSTRAINT DISTRICT1CKC CHECK
(D_W_ID BETWEEN 1 AND 800);
SET INTEGRITY FOR DISTRICT1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT2 OFF;
ALTER TABLE DISTRICT2 DROP CONSTRAINT DISTRICT2CKC;

```

```

ALTER TABLE DISTRICT2 ADD CONSTRAINT DISTRICT2CKC CHECK
(D_W_ID BETWEEN 801 AND 1600);
SET INTEGRITY FOR DISTRICT2 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT3 OFF;
ALTER TABLE DISTRICT3 DROP CONSTRAINT DISTRICT3CKC;
ALTER TABLE DISTRICT3 ADD CONSTRAINT DISTRICT3CKC CHECK
(D_W_ID BETWEEN 1601 AND 2400);
SET INTEGRITY FOR DISTRICT3 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT4 OFF;
ALTER TABLE DISTRICT4 DROP CONSTRAINT DISTRICT4CKC;
ALTER TABLE DISTRICT4 ADD CONSTRAINT DISTRICT4CKC CHECK
(D_W_ID BETWEEN 2401 AND 3200);
SET INTEGRITY FOR DISTRICT4 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT5 OFF;
ALTER TABLE DISTRICT5 DROP CONSTRAINT DISTRICT5CKC;
ALTER TABLE DISTRICT5 ADD CONSTRAINT DISTRICT5CKC CHECK
(D_W_ID BETWEEN 3201 AND 4000);
SET INTEGRITY FOR DISTRICT5 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT6 OFF;
ALTER TABLE DISTRICT6 DROP CONSTRAINT DISTRICT6CKC;
ALTER TABLE DISTRICT6 ADD CONSTRAINT DISTRICT6CKC CHECK
(D_W_ID BETWEEN 4001 AND 4800);
SET INTEGRITY FOR DISTRICT6 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT7 OFF;
ALTER TABLE DISTRICT7 DROP CONSTRAINT DISTRICT7CKC;
ALTER TABLE DISTRICT7 ADD CONSTRAINT DISTRICT7CKC CHECK
(D_W_ID BETWEEN 4801 AND 5600);
SET INTEGRITY FOR DISTRICT7 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT8 OFF;
ALTER TABLE DISTRICT8 DROP CONSTRAINT DISTRICT8CKC;
ALTER TABLE DISTRICT8 ADD CONSTRAINT DISTRICT8CKC CHECK
(D_W_ID BETWEEN 5601 AND 6400);
SET INTEGRITY FOR DISTRICT8 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT9 OFF;
ALTER TABLE DISTRICT9 DROP CONSTRAINT DISTRICT9CKC;
ALTER TABLE DISTRICT9 ADD CONSTRAINT DISTRICT9CKC CHECK
(D_W_ID BETWEEN 6401 AND 7200);
SET INTEGRITY FOR DISTRICT9 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT10 OFF;
ALTER TABLE DISTRICT10 DROP CONSTRAINT DISTRICT10CKC;
ALTER TABLE DISTRICT10 ADD CONSTRAINT DISTRICT10CKC
CHECK (D_W_ID BETWEEN 7201 AND 8000);
SET INTEGRITY FOR DISTRICT10 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT11 OFF;
ALTER TABLE DISTRICT11 DROP CONSTRAINT DISTRICT11CKC;
ALTER TABLE DISTRICT11 ADD CONSTRAINT DISTRICT11CKC
CHECK (D_W_ID BETWEEN 8001 AND 8800);
SET INTEGRITY FOR DISTRICT11 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;

```

```

SET INTEGRITY FOR DISTRICT12 OFF;
ALTER TABLE DISTRICT12 DROP CONSTRAINT DISTRICT12CKC;
ALTER TABLE DISTRICT12 ADD CONSTRAINT DISTRICT12CKC
CHECK (D_W_ID BETWEEN 8801 AND 9600);
SET INTEGRITY FOR DISTRICT12 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT13 OFF;
ALTER TABLE DISTRICT13 DROP CONSTRAINT DISTRICT13CKC;
ALTER TABLE DISTRICT13 ADD CONSTRAINT DISTRICT13CKC
CHECK (D_W_ID BETWEEN 9601 AND 10400);
SET INTEGRITY FOR DISTRICT13 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT14 OFF;
ALTER TABLE DISTRICT14 DROP CONSTRAINT DISTRICT14CKC;
ALTER TABLE DISTRICT14 ADD CONSTRAINT DISTRICT14CKC
CHECK (D_W_ID BETWEEN 10401 AND 11200);
SET INTEGRITY FOR DISTRICT14 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT15 OFF;
ALTER TABLE DISTRICT15 DROP CONSTRAINT DISTRICT15CKC;
ALTER TABLE DISTRICT15 ADD CONSTRAINT DISTRICT15CKC
CHECK (D_W_ID BETWEEN 11201 AND 12000);
SET INTEGRITY FOR DISTRICT15 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT16 OFF;
ALTER TABLE DISTRICT16 DROP CONSTRAINT DISTRICT16CKC;
ALTER TABLE DISTRICT16 ADD CONSTRAINT DISTRICT16CKC
CHECK (D_W_ID BETWEEN 12001 AND 12800);
SET INTEGRITY FOR DISTRICT16 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT17 OFF;
ALTER TABLE DISTRICT17 DROP CONSTRAINT DISTRICT17CKC;
ALTER TABLE DISTRICT17 ADD CONSTRAINT DISTRICT17CKC
CHECK (D_W_ID BETWEEN 12801 AND 13600);
SET INTEGRITY FOR DISTRICT17 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT18 OFF;
ALTER TABLE DISTRICT18 DROP CONSTRAINT DISTRICT18CKC;
ALTER TABLE DISTRICT18 ADD CONSTRAINT DISTRICT18CKC
CHECK (D_W_ID BETWEEN 13601 AND 14400);
SET INTEGRITY FOR DISTRICT18 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT19 OFF;
ALTER TABLE DISTRICT19 DROP CONSTRAINT DISTRICT19CKC;
ALTER TABLE DISTRICT19 ADD CONSTRAINT DISTRICT19CKC
CHECK (D_W_ID BETWEEN 14401 AND 15200);
SET INTEGRITY FOR DISTRICT19 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT20 OFF;
ALTER TABLE DISTRICT20 DROP CONSTRAINT DISTRICT20CKC;
ALTER TABLE DISTRICT20 ADD CONSTRAINT DISTRICT20CKC
CHECK (D_W_ID BETWEEN 15201 AND 16000);
SET INTEGRITY FOR DISTRICT20 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT21 OFF;
ALTER TABLE DISTRICT21 DROP CONSTRAINT DISTRICT21CKC;
ALTER TABLE DISTRICT21 ADD CONSTRAINT DISTRICT21CKC
CHECK (D_W_ID BETWEEN 16001 AND 16800);
SET INTEGRITY FOR DISTRICT21 ALL IMMEDIATE UNCHECKED;

```

```

connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT22 OFF;
ALTER TABLE DISTRICT22 DROP CONSTRAINT DISTRICT22CKC;
ALTER TABLE DISTRICT22 ADD CONSTRAINT DISTRICT22CKC
CHECK (D_W_ID BETWEEN 16801 AND 17600);
SET INTEGRITY FOR DISTRICT22 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT23 OFF;
ALTER TABLE DISTRICT23 DROP CONSTRAINT DISTRICT23CKC;
ALTER TABLE DISTRICT23 ADD CONSTRAINT DISTRICT23CKC
CHECK (D_W_ID BETWEEN 17601 AND 18400);
SET INTEGRITY FOR DISTRICT23 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT24 OFF;
ALTER TABLE DISTRICT24 DROP CONSTRAINT DISTRICT24CKC;
ALTER TABLE DISTRICT24 ADD CONSTRAINT DISTRICT24CKC
CHECK (D_W_ID BETWEEN 18401 AND 19200);
SET INTEGRITY FOR DISTRICT24 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR DISTRICT25 OFF;
ALTER TABLE DISTRICT25 DROP CONSTRAINT DISTRICT25CKC;
ALTER TABLE DISTRICT25 ADD CONSTRAINT DISTRICT25CKC
CHECK (D_W_ID >= 19201);
SET INTEGRITY FOR DISTRICT25 ALL IMMEDIATE UNCHECKED;
connect reset;

```

crconst_history.ddl

```

connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY1 OFF;
ALTER TABLE HISTORY1 DROP CONSTRAINT HISTORY1CKC;
ALTER TABLE HISTORY1 ADD CONSTRAINT HISTORY1CKC CHECK
(H_W_ID BETWEEN 1 AND 800);
SET INTEGRITY FOR HISTORY1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY2 OFF;
ALTER TABLE HISTORY2 DROP CONSTRAINT HISTORY2CKC;
ALTER TABLE HISTORY2 ADD CONSTRAINT HISTORY2CKC CHECK
(H_W_ID BETWEEN 801 AND 1600);
SET INTEGRITY FOR HISTORY2 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY3 OFF;
ALTER TABLE HISTORY3 DROP CONSTRAINT HISTORY3CKC;
ALTER TABLE HISTORY3 ADD CONSTRAINT HISTORY3CKC CHECK
(H_W_ID BETWEEN 1601 AND 2400);
SET INTEGRITY FOR HISTORY3 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY4 OFF;
ALTER TABLE HISTORY4 DROP CONSTRAINT HISTORY4CKC;
ALTER TABLE HISTORY4 ADD CONSTRAINT HISTORY4CKC CHECK
(H_W_ID BETWEEN 2401 AND 3200);
SET INTEGRITY FOR HISTORY4 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY5 OFF;
ALTER TABLE HISTORY5 DROP CONSTRAINT HISTORY5CKC;
ALTER TABLE HISTORY5 ADD CONSTRAINT HISTORY5CKC CHECK
(H_W_ID BETWEEN 3201 AND 4000);
SET INTEGRITY FOR HISTORY5 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;

```

```

SET INTEGRITY FOR HISTORY6 OFF;
ALTER TABLE HISTORY6 DROP CONSTRAINT HISTORY6CKC;
ALTER TABLE HISTORY6 ADD CONSTRAINT HISTORY6CKC CHECK
(H_W_ID BETWEEN 4001 AND 4800);
SET INTEGRITY FOR HISTORY6 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY7 OFF;
ALTER TABLE HISTORY7 DROP CONSTRAINT HISTORY7CKC;
ALTER TABLE HISTORY7 ADD CONSTRAINT HISTORY7CKC CHECK
(H_W_ID BETWEEN 4801 AND 5600);
SET INTEGRITY FOR HISTORY7 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY8 OFF;
ALTER TABLE HISTORY8 DROP CONSTRAINT HISTORY8CKC;
ALTER TABLE HISTORY8 ADD CONSTRAINT HISTORY8CKC CHECK
(H_W_ID BETWEEN 5601 AND 6400);
SET INTEGRITY FOR HISTORY8 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY9 OFF;
ALTER TABLE HISTORY9 DROP CONSTRAINT HISTORY9CKC;
ALTER TABLE HISTORY9 ADD CONSTRAINT HISTORY9CKC CHECK
(H_W_ID BETWEEN 6401 AND 7200);
SET INTEGRITY FOR HISTORY9 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY10 OFF;
ALTER TABLE HISTORY10 DROP CONSTRAINT HISTORY10CKC;
ALTER TABLE HISTORY10 ADD CONSTRAINT HISTORY10CKC
CHECK (H_W_ID BETWEEN 7201 AND 8000);
SET INTEGRITY FOR HISTORY10 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY11 OFF;
ALTER TABLE HISTORY11 DROP CONSTRAINT HISTORY11CKC;
ALTER TABLE HISTORY11 ADD CONSTRAINT HISTORY11CKC
CHECK (H_W_ID BETWEEN 8001 AND 8800);
SET INTEGRITY FOR HISTORY11 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY12 OFF;
ALTER TABLE HISTORY12 DROP CONSTRAINT HISTORY12CKC;
ALTER TABLE HISTORY12 ADD CONSTRAINT HISTORY12CKC
CHECK (H_W_ID BETWEEN 8801 AND 9600);
SET INTEGRITY FOR HISTORY12 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY13 OFF;
ALTER TABLE HISTORY13 DROP CONSTRAINT HISTORY13CKC;
ALTER TABLE HISTORY13 ADD CONSTRAINT HISTORY13CKC
CHECK (H_W_ID BETWEEN 9601 AND 10400);
SET INTEGRITY FOR HISTORY13 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY14 OFF;
ALTER TABLE HISTORY14 DROP CONSTRAINT HISTORY14CKC;
ALTER TABLE HISTORY14 ADD CONSTRAINT HISTORY14CKC
CHECK (H_W_ID BETWEEN 10401 AND 11200);
SET INTEGRITY FOR HISTORY14 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY15 OFF;
ALTER TABLE HISTORY15 DROP CONSTRAINT HISTORY15CKC;
ALTER TABLE HISTORY15 ADD CONSTRAINT HISTORY15CKC
CHECK (H_W_ID BETWEEN 11201 AND 12000);
SET INTEGRITY FOR HISTORY15 ALL IMMEDIATE UNCHECKED;

```

```

connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY16 OFF;
ALTER TABLE HISTORY16 DROP CONSTRAINT HISTORY16CKC;
ALTER TABLE HISTORY16 ADD CONSTRAINT HISTORY16CKC
CHECK (H_W_ID BETWEEN 12001 AND 12800);
SET INTEGRITY FOR HISTORY16 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY17 OFF;
ALTER TABLE HISTORY17 DROP CONSTRAINT HISTORY17CKC;
ALTER TABLE HISTORY17 ADD CONSTRAINT HISTORY17CKC
CHECK (H_W_ID BETWEEN 12801 AND 13600);
SET INTEGRITY FOR HISTORY17 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY18 OFF;
ALTER TABLE HISTORY18 DROP CONSTRAINT HISTORY18CKC;
ALTER TABLE HISTORY18 ADD CONSTRAINT HISTORY18CKC
CHECK (H_W_ID BETWEEN 13601 AND 14400);
SET INTEGRITY FOR HISTORY18 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY19 OFF;
ALTER TABLE HISTORY19 DROP CONSTRAINT HISTORY19CKC;
ALTER TABLE HISTORY19 ADD CONSTRAINT HISTORY19CKC
CHECK (H_W_ID BETWEEN 14401 AND 15200);
SET INTEGRITY FOR HISTORY19 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY20 OFF;
ALTER TABLE HISTORY20 DROP CONSTRAINT HISTORY20CKC;
ALTER TABLE HISTORY20 ADD CONSTRAINT HISTORY20CKC
CHECK (H_W_ID BETWEEN 15201 AND 16000);
SET INTEGRITY FOR HISTORY20 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY21 OFF;
ALTER TABLE HISTORY21 DROP CONSTRAINT HISTORY21CKC;
ALTER TABLE HISTORY21 ADD CONSTRAINT HISTORY21CKC
CHECK (H_W_ID BETWEEN 16001 AND 16800);
SET INTEGRITY FOR HISTORY21 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY22 OFF;
ALTER TABLE HISTORY22 DROP CONSTRAINT HISTORY22CKC;
ALTER TABLE HISTORY22 ADD CONSTRAINT HISTORY22CKC
CHECK (H_W_ID BETWEEN 16801 AND 17600);
SET INTEGRITY FOR HISTORY22 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY23 OFF;
ALTER TABLE HISTORY23 DROP CONSTRAINT HISTORY23CKC;
ALTER TABLE HISTORY23 ADD CONSTRAINT HISTORY23CKC
CHECK (H_W_ID BETWEEN 17601 AND 18400);
SET INTEGRITY FOR HISTORY23 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY24 OFF;
ALTER TABLE HISTORY24 DROP CONSTRAINT HISTORY24CKC;
ALTER TABLE HISTORY24 ADD CONSTRAINT HISTORY24CKC
CHECK (H_W_ID BETWEEN 18401 AND 19200);
SET INTEGRITY FOR HISTORY24 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR HISTORY25 OFF;
ALTER TABLE HISTORY25 DROP CONSTRAINT HISTORY25CKC;

```

```
ALTER TABLE HISTORY25 ADD CONSTRAINT HISTORY25CKC
CHECK (H_W_ID >= 19201);
SET INTEGRITY FOR HISTORY25 ALL IMMEDIATE UNCHECKED;
connect reset;
```

crconst_new_ordera.ddl

```
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA1 OFF;
ALTER TABLE NEW_ORDERA1 DROP CONSTRAINT
NEW_ORDERA1CKC;
ALTER TABLE NEW_ORDERA1 ADD CONSTRAINT
NEW_ORDERA1CKC CHECK ((NO_W_ID BETWEEN 1 AND 800) AND
(NO_O_ID <= 3675));
SET INTEGRITY FOR NEW_ORDERA1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA2 OFF;
ALTER TABLE NEW_ORDERA2 DROP CONSTRAINT
NEW_ORDERA2CKC;
ALTER TABLE NEW_ORDERA2 ADD CONSTRAINT
NEW_ORDERA2CKC CHECK ((NO_W_ID BETWEEN 801 AND 1600)
AND (NO_O_ID <= 3675));
SET INTEGRITY FOR NEW_ORDERA2 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA3 OFF;
ALTER TABLE NEW_ORDERA3 DROP CONSTRAINT
NEW_ORDERA3CKC;
ALTER TABLE NEW_ORDERA3 ADD CONSTRAINT
NEW_ORDERA3CKC CHECK ((NO_W_ID BETWEEN 1601 AND 2400)
AND (NO_O_ID <= 3675));
SET INTEGRITY FOR NEW_ORDERA3 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA4 OFF;
ALTER TABLE NEW_ORDERA4 DROP CONSTRAINT
NEW_ORDERA4CKC;
ALTER TABLE NEW_ORDERA4 ADD CONSTRAINT
NEW_ORDERA4CKC CHECK ((NO_W_ID BETWEEN 2401 AND 3200)
AND (NO_O_ID <= 3675));
SET INTEGRITY FOR NEW_ORDERA4 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA5 OFF;
ALTER TABLE NEW_ORDERA5 DROP CONSTRAINT
NEW_ORDERA5CKC;
ALTER TABLE NEW_ORDERA5 ADD CONSTRAINT
NEW_ORDERA5CKC CHECK ((NO_W_ID BETWEEN 3201 AND 4000)
AND (NO_O_ID <= 3675));
SET INTEGRITY FOR NEW_ORDERA5 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA6 OFF;
ALTER TABLE NEW_ORDERA6 DROP CONSTRAINT
NEW_ORDERA6CKC;
ALTER TABLE NEW_ORDERA6 ADD CONSTRAINT
NEW_ORDERA6CKC CHECK ((NO_W_ID BETWEEN 4001 AND 4800)
AND (NO_O_ID <= 3675));
SET INTEGRITY FOR NEW_ORDERA6 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA7 OFF;
ALTER TABLE NEW_ORDERA7 DROP CONSTRAINT
NEW_ORDERA7CKC;
```

```
ALTER TABLE NEW_ORDERA7 ADD CONSTRAINT
NEW_ORDERA7CKC CHECK ((NO_W_ID BETWEEN 4801 AND 5600)
AND (NO_O_ID <= 3675));
SET INTEGRITY FOR NEW_ORDERA7 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA8 OFF;
ALTER TABLE NEW_ORDERA8 DROP CONSTRAINT
NEW_ORDERA8CKC;
ALTER TABLE NEW_ORDERA8 ADD CONSTRAINT
NEW_ORDERA8CKC CHECK ((NO_W_ID BETWEEN 5601 AND 6400)
AND (NO_O_ID <= 3675));
SET INTEGRITY FOR NEW_ORDERA8 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA9 OFF;
ALTER TABLE NEW_ORDERA9 DROP CONSTRAINT
NEW_ORDERA9CKC;
ALTER TABLE NEW_ORDERA9 ADD CONSTRAINT
NEW_ORDERA9CKC CHECK ((NO_W_ID BETWEEN 6401 AND 7200)
AND (NO_O_ID <= 3675));
SET INTEGRITY FOR NEW_ORDERA9 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA10 OFF;
ALTER TABLE NEW_ORDERA10 DROP CONSTRAINT
NEW_ORDERA10CKC;
ALTER TABLE NEW_ORDERA10 ADD CONSTRAINT
NEW_ORDERA10CKC CHECK ((NO_W_ID BETWEEN 7201 AND 8000)
AND (NO_O_ID <= 3675));
SET INTEGRITY FOR NEW_ORDERA10 ALL IMMEDIATE
UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA11 OFF;
ALTER TABLE NEW_ORDERA11 DROP CONSTRAINT
NEW_ORDERA11CKC;
ALTER TABLE NEW_ORDERA11 ADD CONSTRAINT
NEW_ORDERA11CKC CHECK ((NO_W_ID BETWEEN 8001 AND 8800)
AND (NO_O_ID <= 3675));
SET INTEGRITY FOR NEW_ORDERA11 ALL IMMEDIATE
UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA12 OFF;
ALTER TABLE NEW_ORDERA12 DROP CONSTRAINT
NEW_ORDERA12CKC;
ALTER TABLE NEW_ORDERA12 ADD CONSTRAINT
NEW_ORDERA12CKC CHECK ((NO_W_ID BETWEEN 8801 AND 9600)
AND (NO_O_ID <= 3675));
SET INTEGRITY FOR NEW_ORDERA12 ALL IMMEDIATE
UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA13 OFF;
ALTER TABLE NEW_ORDERA13 DROP CONSTRAINT
NEW_ORDERA13CKC;
ALTER TABLE NEW_ORDERA13 ADD CONSTRAINT
NEW_ORDERA13CKC CHECK ((NO_W_ID BETWEEN 9601 AND 10400)
AND (NO_O_ID <= 3675));
SET INTEGRITY FOR NEW_ORDERA13 ALL IMMEDIATE
UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA14 OFF;
ALTER TABLE NEW_ORDERA14 DROP CONSTRAINT
NEW_ORDERA14CKC;
```

```

ALTER TABLE NEW_ORDERA14 ADD CONSTRAINT
NEW_ORDERA14CKC CHECK ((NO_W_ID BETWEEN 10401 AND 11200)
AND (NO_O_ID <= 3675));
SET INTEGRITY FOR NEW_ORDERA14 ALL IMMEDIATE
UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA15 OFF;
ALTER TABLE NEW_ORDERA15 DROP CONSTRAINT
NEW_ORDERA15CKC;
ALTER TABLE NEW_ORDERA15 ADD CONSTRAINT
NEW_ORDERA15CKC CHECK ((NO_W_ID BETWEEN 11201 AND 12000)
AND (NO_O_ID <= 3675));
SET INTEGRITY FOR NEW_ORDERA15 ALL IMMEDIATE
UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA16 OFF;
ALTER TABLE NEW_ORDERA16 DROP CONSTRAINT
NEW_ORDERA16CKC;
ALTER TABLE NEW_ORDERA16 ADD CONSTRAINT
NEW_ORDERA16CKC CHECK ((NO_W_ID BETWEEN 12001 AND 12800)
AND (NO_O_ID <= 3675));
SET INTEGRITY FOR NEW_ORDERA16 ALL IMMEDIATE
UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA17 OFF;
ALTER TABLE NEW_ORDERA17 DROP CONSTRAINT
NEW_ORDERA17CKC;
ALTER TABLE NEW_ORDERA17 ADD CONSTRAINT
NEW_ORDERA17CKC CHECK ((NO_W_ID BETWEEN 12801 AND 13600)
AND (NO_O_ID <= 3675));
SET INTEGRITY FOR NEW_ORDERA17 ALL IMMEDIATE
UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA18 OFF;
ALTER TABLE NEW_ORDERA18 DROP CONSTRAINT
NEW_ORDERA18CKC;
ALTER TABLE NEW_ORDERA18 ADD CONSTRAINT
NEW_ORDERA18CKC CHECK ((NO_W_ID BETWEEN 13601 AND 14400)
AND (NO_O_ID <= 3675));
SET INTEGRITY FOR NEW_ORDERA18 ALL IMMEDIATE
UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA19 OFF;
ALTER TABLE NEW_ORDERA19 DROP CONSTRAINT
NEW_ORDERA19CKC;
ALTER TABLE NEW_ORDERA19 ADD CONSTRAINT
NEW_ORDERA19CKC CHECK ((NO_W_ID BETWEEN 14401 AND 15200)
AND (NO_O_ID <= 3675));
SET INTEGRITY FOR NEW_ORDERA19 ALL IMMEDIATE
UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA20 OFF;
ALTER TABLE NEW_ORDERA20 DROP CONSTRAINT
NEW_ORDERA20CKC;
ALTER TABLE NEW_ORDERA20 ADD CONSTRAINT
NEW_ORDERA20CKC CHECK ((NO_W_ID BETWEEN 15201 AND 16000)
AND (NO_O_ID <= 3675));
SET INTEGRITY FOR NEW_ORDERA20 ALL IMMEDIATE
UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA21 OFF;

```

```

ALTER TABLE NEW_ORDERA21 DROP CONSTRAINT
NEW_ORDERA21CKC;
ALTER TABLE NEW_ORDERA21 ADD CONSTRAINT
NEW_ORDERA21CKC CHECK ((NO_W_ID BETWEEN 16001 AND 16800)
AND (NO_O_ID <= 3675));
SET INTEGRITY FOR NEW_ORDERA21 ALL IMMEDIATE
UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA22 OFF;
ALTER TABLE NEW_ORDERA22 DROP CONSTRAINT
NEW_ORDERA22CKC;
ALTER TABLE NEW_ORDERA22 ADD CONSTRAINT
NEW_ORDERA22CKC CHECK ((NO_W_ID BETWEEN 16801 AND 17600)
AND (NO_O_ID <= 3675));
SET INTEGRITY FOR NEW_ORDERA22 ALL IMMEDIATE
UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA23 OFF;
ALTER TABLE NEW_ORDERA23 DROP CONSTRAINT
NEW_ORDERA23CKC;
ALTER TABLE NEW_ORDERA23 ADD CONSTRAINT
NEW_ORDERA23CKC CHECK ((NO_W_ID BETWEEN 17601 AND 18400)
AND (NO_O_ID <= 3675));
SET INTEGRITY FOR NEW_ORDERA23 ALL IMMEDIATE
UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA24 OFF;
ALTER TABLE NEW_ORDERA24 DROP CONSTRAINT
NEW_ORDERA24CKC;
ALTER TABLE NEW_ORDERA24 ADD CONSTRAINT
NEW_ORDERA24CKC CHECK ((NO_W_ID BETWEEN 18401 AND 19200)
AND (NO_O_ID <= 3675));
SET INTEGRITY FOR NEW_ORDERA24 ALL IMMEDIATE
UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERA25 OFF;
ALTER TABLE NEW_ORDERA25 DROP CONSTRAINT
NEW_ORDERA25CKC;
ALTER TABLE NEW_ORDERA25 ADD CONSTRAINT
NEW_ORDERA25CKC CHECK ((NO_W_ID >= 19201) AND (NO_O_ID <=
3675));
SET INTEGRITY FOR NEW_ORDERA25 ALL IMMEDIATE
UNCHECKED;
connect reset;

```

crconst_new_orderb.ddl

```

connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB1 OFF;
ALTER TABLE NEW_ORDERB1 DROP CONSTRAINT
NEW_ORDERB1CKC;
ALTER TABLE NEW_ORDERB1 ADD CONSTRAINT
NEW_ORDERB1CKC CHECK ((NO_W_ID BETWEEN 1 AND 800) AND
(NO_O_ID >= 3676));
SET INTEGRITY FOR NEW_ORDERB1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB2 OFF;
ALTER TABLE NEW_ORDERB2 DROP CONSTRAINT
NEW_ORDERB2CKC;
ALTER TABLE NEW_ORDERB2 ADD CONSTRAINT
NEW_ORDERB2CKC CHECK ((NO_W_ID BETWEEN 801 AND 1600)
AND (NO_O_ID >= 3676));
SET INTEGRITY FOR NEW_ORDERB2 ALL IMMEDIATE UNCHECKED;

```

```

connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB3 OFF;
ALTER TABLE NEW_ORDERB3 DROP CONSTRAINT
NEW_ORDERB3CKC;
ALTER TABLE NEW_ORDERB3 ADD CONSTRAINT
NEW_ORDERB3CKC CHECK ((NO_W_ID BETWEEN 1601 AND 2400)
AND (NO_O_ID >= 3676));
SET INTEGRITY FOR NEW_ORDERB3 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB4 OFF;
ALTER TABLE NEW_ORDERB4 DROP CONSTRAINT
NEW_ORDERB4CKC;
ALTER TABLE NEW_ORDERB4 ADD CONSTRAINT
NEW_ORDERB4CKC CHECK ((NO_W_ID BETWEEN 2401 AND 3200)
AND (NO_O_ID >= 3676));
SET INTEGRITY FOR NEW_ORDERB4 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB5 OFF;
ALTER TABLE NEW_ORDERB5 DROP CONSTRAINT
NEW_ORDERB5CKC;
ALTER TABLE NEW_ORDERB5 ADD CONSTRAINT
NEW_ORDERB5CKC CHECK ((NO_W_ID BETWEEN 3201 AND 4000)
AND (NO_O_ID >= 3676));
SET INTEGRITY FOR NEW_ORDERB5 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB6 OFF;
ALTER TABLE NEW_ORDERB6 DROP CONSTRAINT
NEW_ORDERB6CKC;
ALTER TABLE NEW_ORDERB6 ADD CONSTRAINT
NEW_ORDERB6CKC CHECK ((NO_W_ID BETWEEN 4001 AND 4800)
AND (NO_O_ID >= 3676));
SET INTEGRITY FOR NEW_ORDERB6 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB7 OFF;
ALTER TABLE NEW_ORDERB7 DROP CONSTRAINT
NEW_ORDERB7CKC;
ALTER TABLE NEW_ORDERB7 ADD CONSTRAINT
NEW_ORDERB7CKC CHECK ((NO_W_ID BETWEEN 4801 AND 5600)
AND (NO_O_ID >= 3676));
SET INTEGRITY FOR NEW_ORDERB7 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB8 OFF;
ALTER TABLE NEW_ORDERB8 DROP CONSTRAINT
NEW_ORDERB8CKC;
ALTER TABLE NEW_ORDERB8 ADD CONSTRAINT
NEW_ORDERB8CKC CHECK ((NO_W_ID BETWEEN 5601 AND 6400)
AND (NO_O_ID >= 3676));
SET INTEGRITY FOR NEW_ORDERB8 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB9 OFF;
ALTER TABLE NEW_ORDERB9 DROP CONSTRAINT
NEW_ORDERB9CKC;
ALTER TABLE NEW_ORDERB9 ADD CONSTRAINT
NEW_ORDERB9CKC CHECK ((NO_W_ID BETWEEN 6401 AND 7200)
AND (NO_O_ID >= 3676));
SET INTEGRITY FOR NEW_ORDERB9 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB10 OFF;
ALTER TABLE NEW_ORDERB10 DROP CONSTRAINT
NEW_ORDERB10CKC;

```

```

ALTER TABLE NEW_ORDERB10 ADD CONSTRAINT
NEW_ORDERB10CKC CHECK ((NO_W_ID BETWEEN 7201 AND 8000)
AND (NO_O_ID >= 3676));
SET INTEGRITY FOR NEW_ORDERB10 ALL IMMEDIATE
UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB11 OFF;
ALTER TABLE NEW_ORDERB11 DROP CONSTRAINT
NEW_ORDERB11CKC;
ALTER TABLE NEW_ORDERB11 ADD CONSTRAINT
NEW_ORDERB11CKC CHECK ((NO_W_ID BETWEEN 8001 AND 8800)
AND (NO_O_ID >= 3676));
SET INTEGRITY FOR NEW_ORDERB11 ALL IMMEDIATE
UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB12 OFF;
ALTER TABLE NEW_ORDERB12 DROP CONSTRAINT
NEW_ORDERB12CKC;
ALTER TABLE NEW_ORDERB12 ADD CONSTRAINT
NEW_ORDERB12CKC CHECK ((NO_W_ID BETWEEN 8801 AND 9600)
AND (NO_O_ID >= 3676));
SET INTEGRITY FOR NEW_ORDERB12 ALL IMMEDIATE
UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB13 OFF;
ALTER TABLE NEW_ORDERB13 DROP CONSTRAINT
NEW_ORDERB13CKC;
ALTER TABLE NEW_ORDERB13 ADD CONSTRAINT
NEW_ORDERB13CKC CHECK ((NO_W_ID BETWEEN 9601 AND 10400)
AND (NO_O_ID >= 3676));
SET INTEGRITY FOR NEW_ORDERB13 ALL IMMEDIATE
UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB14 OFF;
ALTER TABLE NEW_ORDERB14 DROP CONSTRAINT
NEW_ORDERB14CKC;
ALTER TABLE NEW_ORDERB14 ADD CONSTRAINT
NEW_ORDERB14CKC CHECK ((NO_W_ID BETWEEN 10401 AND 11200)
AND (NO_O_ID >= 3676));
SET INTEGRITY FOR NEW_ORDERB14 ALL IMMEDIATE
UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB15 OFF;
ALTER TABLE NEW_ORDERB15 DROP CONSTRAINT
NEW_ORDERB15CKC;
ALTER TABLE NEW_ORDERB15 ADD CONSTRAINT
NEW_ORDERB15CKC CHECK ((NO_W_ID BETWEEN 11201 AND 12000)
AND (NO_O_ID >= 3676));
SET INTEGRITY FOR NEW_ORDERB15 ALL IMMEDIATE
UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB16 OFF;
ALTER TABLE NEW_ORDERB16 DROP CONSTRAINT
NEW_ORDERB16CKC;
ALTER TABLE NEW_ORDERB16 ADD CONSTRAINT
NEW_ORDERB16CKC CHECK ((NO_W_ID BETWEEN 12001 AND 12800)
AND (NO_O_ID >= 3676));
SET INTEGRITY FOR NEW_ORDERB16 ALL IMMEDIATE
UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB17 OFF;

```

```

ALTER TABLE NEW_ORDERB17 DROP CONSTRAINT
NEW_ORDERB17CKC;
ALTER TABLE NEW_ORDERB17 ADD CONSTRAINT
NEW_ORDERB17CKC CHECK ((NO_W_ID BETWEEN 12801 AND 13600)
AND (NO_O_ID >= 3676));
SET INTEGRITY FOR NEW_ORDERB17 ALL IMMEDIATE
UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB18 OFF;
ALTER TABLE NEW_ORDERB18 DROP CONSTRAINT
NEW_ORDERB18CKC;
ALTER TABLE NEW_ORDERB18 ADD CONSTRAINT
NEW_ORDERB18CKC CHECK ((NO_W_ID BETWEEN 13601 AND 14400)
AND (NO_O_ID >= 3676));
SET INTEGRITY FOR NEW_ORDERB18 ALL IMMEDIATE
UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB19 OFF;
ALTER TABLE NEW_ORDERB19 DROP CONSTRAINT
NEW_ORDERB19CKC;
ALTER TABLE NEW_ORDERB19 ADD CONSTRAINT
NEW_ORDERB19CKC CHECK ((NO_W_ID BETWEEN 14401 AND 15200)
AND (NO_O_ID >= 3676));
SET INTEGRITY FOR NEW_ORDERB19 ALL IMMEDIATE
UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB20 OFF;
ALTER TABLE NEW_ORDERB20 DROP CONSTRAINT
NEW_ORDERB20CKC;
ALTER TABLE NEW_ORDERB20 ADD CONSTRAINT
NEW_ORDERB20CKC CHECK ((NO_W_ID BETWEEN 15201 AND 16000)
AND (NO_O_ID >= 3676));
SET INTEGRITY FOR NEW_ORDERB20 ALL IMMEDIATE
UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB21 OFF;
ALTER TABLE NEW_ORDERB21 DROP CONSTRAINT
NEW_ORDERB21CKC;
ALTER TABLE NEW_ORDERB21 ADD CONSTRAINT
NEW_ORDERB21CKC CHECK ((NO_W_ID BETWEEN 16001 AND 16800)
AND (NO_O_ID >= 3676));
SET INTEGRITY FOR NEW_ORDERB21 ALL IMMEDIATE
UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB22 OFF;
ALTER TABLE NEW_ORDERB22 DROP CONSTRAINT
NEW_ORDERB22CKC;
ALTER TABLE NEW_ORDERB22 ADD CONSTRAINT
NEW_ORDERB22CKC CHECK ((NO_W_ID BETWEEN 16801 AND 17600)
AND (NO_O_ID >= 3676));
SET INTEGRITY FOR NEW_ORDERB22 ALL IMMEDIATE
UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB23 OFF;
ALTER TABLE NEW_ORDERB23 DROP CONSTRAINT
NEW_ORDERB23CKC;
ALTER TABLE NEW_ORDERB23 ADD CONSTRAINT
NEW_ORDERB23CKC CHECK ((NO_W_ID BETWEEN 17601 AND 18400)
AND (NO_O_ID >= 3676));
SET INTEGRITY FOR NEW_ORDERB23 ALL IMMEDIATE
UNCHECKED;
connect reset;

```

```

connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB24 OFF;
ALTER TABLE NEW_ORDERB24 DROP CONSTRAINT
NEW_ORDERB24CKC;
ALTER TABLE NEW_ORDERB24 ADD CONSTRAINT
NEW_ORDERB24CKC CHECK ((NO_W_ID BETWEEN 18401 AND 19200)
AND (NO_O_ID >= 3676));
SET INTEGRITY FOR NEW_ORDERB24 ALL IMMEDIATE
UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR NEW_ORDERB25 OFF;
ALTER TABLE NEW_ORDERB25 DROP CONSTRAINT
NEW_ORDERB25CKC;
ALTER TABLE NEW_ORDERB25 ADD CONSTRAINT
NEW_ORDERB25CKC CHECK ((NO_W_ID >= 19201) AND (NO_O_ID >=
3676));
SET INTEGRITY FOR NEW_ORDERB25 ALL IMMEDIATE
UNCHECKED;
connect reset;

```

crconst_order_line.ddl

```

connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE1 OFF;
ALTER TABLE ORDER_LINE1 DROP CONSTRAINT
ORDER_LINE1CKC;
ALTER TABLE ORDER_LINE1 ADD CONSTRAINT ORDER_LINE1CKC
CHECK (OL_W_ID BETWEEN 1 AND 800);
SET INTEGRITY FOR ORDER_LINE1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE2 OFF;
ALTER TABLE ORDER_LINE2 DROP CONSTRAINT
ORDER_LINE2CKC;
ALTER TABLE ORDER_LINE2 ADD CONSTRAINT ORDER_LINE2CKC
CHECK (OL_W_ID BETWEEN 801 AND 1600);
SET INTEGRITY FOR ORDER_LINE2 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE3 OFF;
ALTER TABLE ORDER_LINE3 DROP CONSTRAINT
ORDER_LINE3CKC;
ALTER TABLE ORDER_LINE3 ADD CONSTRAINT ORDER_LINE3CKC
CHECK (OL_W_ID BETWEEN 1601 AND 2400);
SET INTEGRITY FOR ORDER_LINE3 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE4 OFF;
ALTER TABLE ORDER_LINE4 DROP CONSTRAINT
ORDER_LINE4CKC;
ALTER TABLE ORDER_LINE4 ADD CONSTRAINT ORDER_LINE4CKC
CHECK (OL_W_ID BETWEEN 2401 AND 3200);
SET INTEGRITY FOR ORDER_LINE4 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE5 OFF;
ALTER TABLE ORDER_LINE5 DROP CONSTRAINT
ORDER_LINE5CKC;
ALTER TABLE ORDER_LINE5 ADD CONSTRAINT ORDER_LINE5CKC
CHECK (OL_W_ID BETWEEN 3201 AND 4000);
SET INTEGRITY FOR ORDER_LINE5 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE6 OFF;
ALTER TABLE ORDER_LINE6 DROP CONSTRAINT
ORDER_LINE6CKC;

```



```

ALTER TABLE ORDER_LINE6 ADD CONSTRAINT ORDER_LINE6CKC
CHECK (OL_W_ID BETWEEN 4001 AND 4800);
SET INTEGRITY FOR ORDER_LINE6 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE7 OFF;
ALTER TABLE ORDER_LINE7 DROP CONSTRAINT
ORDER_LINE7CKC;
ALTER TABLE ORDER_LINE7 ADD CONSTRAINT ORDER_LINE7CKC
CHECK (OL_W_ID BETWEEN 4801 AND 5600);
SET INTEGRITY FOR ORDER_LINE7 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE8 OFF;
ALTER TABLE ORDER_LINE8 DROP CONSTRAINT
ORDER_LINE8CKC;
ALTER TABLE ORDER_LINE8 ADD CONSTRAINT ORDER_LINE8CKC
CHECK (OL_W_ID BETWEEN 5601 AND 6400);
SET INTEGRITY FOR ORDER_LINE8 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE9 OFF;
ALTER TABLE ORDER_LINE9 DROP CONSTRAINT
ORDER_LINE9CKC;
ALTER TABLE ORDER_LINE9 ADD CONSTRAINT ORDER_LINE9CKC
CHECK (OL_W_ID BETWEEN 6401 AND 7200);
SET INTEGRITY FOR ORDER_LINE9 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE10 OFF;
ALTER TABLE ORDER_LINE10 DROP CONSTRAINT
ORDER_LINE10CKC;
ALTER TABLE ORDER_LINE10 ADD CONSTRAINT
ORDER_LINE10CKC CHECK (OL_W_ID BETWEEN 7201 AND 8000);
SET INTEGRITY FOR ORDER_LINE10 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE11 OFF;
ALTER TABLE ORDER_LINE11 DROP CONSTRAINT
ORDER_LINE11CKC;
ALTER TABLE ORDER_LINE11 ADD CONSTRAINT
ORDER_LINE11CKC CHECK (OL_W_ID BETWEEN 8001 AND 8800);
SET INTEGRITY FOR ORDER_LINE11 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE12 OFF;
ALTER TABLE ORDER_LINE12 DROP CONSTRAINT
ORDER_LINE12CKC;
ALTER TABLE ORDER_LINE12 ADD CONSTRAINT
ORDER_LINE12CKC CHECK (OL_W_ID BETWEEN 8801 AND 9600);
SET INTEGRITY FOR ORDER_LINE12 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE13 OFF;
ALTER TABLE ORDER_LINE13 DROP CONSTRAINT
ORDER_LINE13CKC;
ALTER TABLE ORDER_LINE13 ADD CONSTRAINT
ORDER_LINE13CKC CHECK (OL_W_ID BETWEEN 9601 AND 10400);
SET INTEGRITY FOR ORDER_LINE13 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE14 OFF;
ALTER TABLE ORDER_LINE14 DROP CONSTRAINT
ORDER_LINE14CKC;
ALTER TABLE ORDER_LINE14 ADD CONSTRAINT
ORDER_LINE14CKC CHECK (OL_W_ID BETWEEN 10401 AND 11200);
SET INTEGRITY FOR ORDER_LINE14 ALL IMMEDIATE UNCHECKED;
connect reset;

```

```

connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE15 OFF;
ALTER TABLE ORDER_LINE15 DROP CONSTRAINT
ORDER_LINE15CKC;
ALTER TABLE ORDER_LINE15 ADD CONSTRAINT
ORDER_LINE15CKC CHECK (OL_W_ID BETWEEN 11201 AND 12000);
SET INTEGRITY FOR ORDER_LINE15 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE16 OFF;
ALTER TABLE ORDER_LINE16 DROP CONSTRAINT
ORDER_LINE16CKC;
ALTER TABLE ORDER_LINE16 ADD CONSTRAINT
ORDER_LINE16CKC CHECK (OL_W_ID BETWEEN 12001 AND 12800);
SET INTEGRITY FOR ORDER_LINE16 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE17 OFF;
ALTER TABLE ORDER_LINE17 DROP CONSTRAINT
ORDER_LINE17CKC;
ALTER TABLE ORDER_LINE17 ADD CONSTRAINT
ORDER_LINE17CKC CHECK (OL_W_ID BETWEEN 12801 AND 13600);
SET INTEGRITY FOR ORDER_LINE17 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE18 OFF;
ALTER TABLE ORDER_LINE18 DROP CONSTRAINT
ORDER_LINE18CKC;
ALTER TABLE ORDER_LINE18 ADD CONSTRAINT
ORDER_LINE18CKC CHECK (OL_W_ID BETWEEN 13601 AND 14400);
SET INTEGRITY FOR ORDER_LINE18 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE19 OFF;
ALTER TABLE ORDER_LINE19 DROP CONSTRAINT
ORDER_LINE19CKC;
ALTER TABLE ORDER_LINE19 ADD CONSTRAINT
ORDER_LINE19CKC CHECK (OL_W_ID BETWEEN 14401 AND 15200);
SET INTEGRITY FOR ORDER_LINE19 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE20 OFF;
ALTER TABLE ORDER_LINE20 DROP CONSTRAINT
ORDER_LINE20CKC;
ALTER TABLE ORDER_LINE20 ADD CONSTRAINT
ORDER_LINE20CKC CHECK (OL_W_ID BETWEEN 15201 AND 16000);
SET INTEGRITY FOR ORDER_LINE20 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE21 OFF;
ALTER TABLE ORDER_LINE21 DROP CONSTRAINT
ORDER_LINE21CKC;
ALTER TABLE ORDER_LINE21 ADD CONSTRAINT
ORDER_LINE21CKC CHECK (OL_W_ID BETWEEN 16001 AND 16800);
SET INTEGRITY FOR ORDER_LINE21 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE22 OFF;
ALTER TABLE ORDER_LINE22 DROP CONSTRAINT
ORDER_LINE22CKC;
ALTER TABLE ORDER_LINE22 ADD CONSTRAINT
ORDER_LINE22CKC CHECK (OL_W_ID BETWEEN 16801 AND 17600);
SET INTEGRITY FOR ORDER_LINE22 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE23 OFF;
ALTER TABLE ORDER_LINE23 DROP CONSTRAINT
ORDER_LINE23CKC;

```

```

ALTER TABLE ORDER_LINE23 ADD CONSTRAINT
ORDER_LINE23CKC CHECK (OL_W_ID BETWEEN 17601 AND 18400);
SET INTEGRITY FOR ORDER_LINE23 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE24 OFF;
ALTER TABLE ORDER_LINE24 DROP CONSTRAINT
ORDER_LINE24CKC;
ALTER TABLE ORDER_LINE24 ADD CONSTRAINT
ORDER_LINE24CKC CHECK (OL_W_ID BETWEEN 18401 AND 19200);
SET INTEGRITY FOR ORDER_LINE24 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDER_LINE25 OFF;
ALTER TABLE ORDER_LINE25 DROP CONSTRAINT
ORDER_LINE25CKC;
ALTER TABLE ORDER_LINE25 ADD CONSTRAINT
ORDER_LINE25CKC CHECK (OL_W_ID >= 19201);
SET INTEGRITY FOR ORDER_LINE25 ALL IMMEDIATE UNCHECKED;
connect reset;

```

crconst_orders.ddl

```

connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS1 OFF;
ALTER TABLE ORDERS1 DROP CONSTRAINT ORDERS1CKC;
ALTER TABLE ORDERS1 ADD CONSTRAINT ORDERS1CKC CHECK
(O_W_ID BETWEEN 1 AND 800);
SET INTEGRITY FOR ORDERS1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS2 OFF;
ALTER TABLE ORDERS2 DROP CONSTRAINT ORDERS2CKC;
ALTER TABLE ORDERS2 ADD CONSTRAINT ORDERS2CKC CHECK
(O_W_ID BETWEEN 801 AND 1600);
SET INTEGRITY FOR ORDERS2 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS3 OFF;
ALTER TABLE ORDERS3 DROP CONSTRAINT ORDERS3CKC;
ALTER TABLE ORDERS3 ADD CONSTRAINT ORDERS3CKC CHECK
(O_W_ID BETWEEN 1601 AND 2400);
SET INTEGRITY FOR ORDERS3 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS4 OFF;
ALTER TABLE ORDERS4 DROP CONSTRAINT ORDERS4CKC;
ALTER TABLE ORDERS4 ADD CONSTRAINT ORDERS4CKC CHECK
(O_W_ID BETWEEN 2401 AND 3200);
SET INTEGRITY FOR ORDERS4 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS5 OFF;
ALTER TABLE ORDERS5 DROP CONSTRAINT ORDERS5CKC;
ALTER TABLE ORDERS5 ADD CONSTRAINT ORDERS5CKC CHECK
(O_W_ID BETWEEN 3201 AND 4000);
SET INTEGRITY FOR ORDERS5 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS6 OFF;
ALTER TABLE ORDERS6 DROP CONSTRAINT ORDERS6CKC;
ALTER TABLE ORDERS6 ADD CONSTRAINT ORDERS6CKC CHECK
(O_W_ID BETWEEN 4001 AND 4800);
SET INTEGRITY FOR ORDERS6 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS7 OFF;
ALTER TABLE ORDERS7 DROP CONSTRAINT ORDERS7CKC;

```

```

ALTER TABLE ORDERS7 ADD CONSTRAINT ORDERS7CKC CHECK
(O_W_ID BETWEEN 4801 AND 5600);
SET INTEGRITY FOR ORDERS7 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS8 OFF;
ALTER TABLE ORDERS8 DROP CONSTRAINT ORDERS8CKC;
ALTER TABLE ORDERS8 ADD CONSTRAINT ORDERS8CKC CHECK
(O_W_ID BETWEEN 5601 AND 6400);
SET INTEGRITY FOR ORDERS8 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS9 OFF;
ALTER TABLE ORDERS9 DROP CONSTRAINT ORDERS9CKC;
ALTER TABLE ORDERS9 ADD CONSTRAINT ORDERS9CKC CHECK
(O_W_ID BETWEEN 6401 AND 7200);
SET INTEGRITY FOR ORDERS9 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS10 OFF;
ALTER TABLE ORDERS10 DROP CONSTRAINT ORDERS10CKC;
ALTER TABLE ORDERS10 ADD CONSTRAINT ORDERS10CKC CHECK
(O_W_ID BETWEEN 7201 AND 8000);
SET INTEGRITY FOR ORDERS10 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS11 OFF;
ALTER TABLE ORDERS11 DROP CONSTRAINT ORDERS11CKC;
ALTER TABLE ORDERS11 ADD CONSTRAINT ORDERS11CKC CHECK
(O_W_ID BETWEEN 8001 AND 8800);
SET INTEGRITY FOR ORDERS11 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS12 OFF;
ALTER TABLE ORDERS12 DROP CONSTRAINT ORDERS12CKC;
ALTER TABLE ORDERS12 ADD CONSTRAINT ORDERS12CKC CHECK
(O_W_ID BETWEEN 8801 AND 9600);
SET INTEGRITY FOR ORDERS12 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS13 OFF;
ALTER TABLE ORDERS13 DROP CONSTRAINT ORDERS13CKC;
ALTER TABLE ORDERS13 ADD CONSTRAINT ORDERS13CKC CHECK
(O_W_ID BETWEEN 9601 AND 10400);
SET INTEGRITY FOR ORDERS13 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS14 OFF;
ALTER TABLE ORDERS14 DROP CONSTRAINT ORDERS14CKC;
ALTER TABLE ORDERS14 ADD CONSTRAINT ORDERS14CKC CHECK
(O_W_ID BETWEEN 10401 AND 11200);
SET INTEGRITY FOR ORDERS14 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS15 OFF;
ALTER TABLE ORDERS15 DROP CONSTRAINT ORDERS15CKC;
ALTER TABLE ORDERS15 ADD CONSTRAINT ORDERS15CKC CHECK
(O_W_ID BETWEEN 11201 AND 12000);
SET INTEGRITY FOR ORDERS15 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS16 OFF;
ALTER TABLE ORDERS16 DROP CONSTRAINT ORDERS16CKC;
ALTER TABLE ORDERS16 ADD CONSTRAINT ORDERS16CKC CHECK
(O_W_ID BETWEEN 12001 AND 12800);
SET INTEGRITY FOR ORDERS16 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;

```

```

SET INTEGRITY FOR ORDERS17 OFF;
ALTER TABLE ORDERS17 DROP CONSTRAINT ORDERS17CKC;
ALTER TABLE ORDERS17 ADD CONSTRAINT ORDERS17CKC CHECK
(O_W_ID BETWEEN 12801 AND 13600);
SET INTEGRITY FOR ORDERS17 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS18 OFF;
ALTER TABLE ORDERS18 DROP CONSTRAINT ORDERS18CKC;
ALTER TABLE ORDERS18 ADD CONSTRAINT ORDERS18CKC CHECK
(O_W_ID BETWEEN 13601 AND 14400);
SET INTEGRITY FOR ORDERS18 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS19 OFF;
ALTER TABLE ORDERS19 DROP CONSTRAINT ORDERS19CKC;
ALTER TABLE ORDERS19 ADD CONSTRAINT ORDERS19CKC CHECK
(O_W_ID BETWEEN 14401 AND 15200);
SET INTEGRITY FOR ORDERS19 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS20 OFF;
ALTER TABLE ORDERS20 DROP CONSTRAINT ORDERS20CKC;
ALTER TABLE ORDERS20 ADD CONSTRAINT ORDERS20CKC CHECK
(O_W_ID BETWEEN 15201 AND 16000);
SET INTEGRITY FOR ORDERS20 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS21 OFF;
ALTER TABLE ORDERS21 DROP CONSTRAINT ORDERS21CKC;
ALTER TABLE ORDERS21 ADD CONSTRAINT ORDERS21CKC CHECK
(O_W_ID BETWEEN 16001 AND 16800);
SET INTEGRITY FOR ORDERS21 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS22 OFF;
ALTER TABLE ORDERS22 DROP CONSTRAINT ORDERS22CKC;
ALTER TABLE ORDERS22 ADD CONSTRAINT ORDERS22CKC CHECK
(O_W_ID BETWEEN 16801 AND 17600);
SET INTEGRITY FOR ORDERS22 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS23 OFF;
ALTER TABLE ORDERS23 DROP CONSTRAINT ORDERS23CKC;
ALTER TABLE ORDERS23 ADD CONSTRAINT ORDERS23CKC CHECK
(O_W_ID BETWEEN 17601 AND 18400);
SET INTEGRITY FOR ORDERS23 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS24 OFF;
ALTER TABLE ORDERS24 DROP CONSTRAINT ORDERS24CKC;
ALTER TABLE ORDERS24 ADD CONSTRAINT ORDERS24CKC CHECK
(O_W_ID BETWEEN 18401 AND 19200);
SET INTEGRITY FOR ORDERS24 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR ORDERS25 OFF;
ALTER TABLE ORDERS25 DROP CONSTRAINT ORDERS25CKC;
ALTER TABLE ORDERS25 ADD CONSTRAINT ORDERS25CKC CHECK
(O_W_ID >= 19201);
SET INTEGRITY FOR ORDERS25 ALL IMMEDIATE UNCHECKED;
connect reset;

```

crconst_stock.ddl

```

connect to TPCC in share mode;
SET INTEGRITY FOR STOCK1 OFF;

```

```

ALTER TABLE STOCK1 DROP CONSTRAINT STOCK1CKC;
ALTER TABLE STOCK1 ADD CONSTRAINT STOCK1CKC CHECK
(S_W_ID BETWEEN 1 AND 800);
SET INTEGRITY FOR STOCK1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK2 OFF;
ALTER TABLE STOCK2 DROP CONSTRAINT STOCK2CKC;
ALTER TABLE STOCK2 ADD CONSTRAINT STOCK2CKC CHECK
(S_W_ID BETWEEN 801 AND 1600);
SET INTEGRITY FOR STOCK2 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK3 OFF;
ALTER TABLE STOCK3 DROP CONSTRAINT STOCK3CKC;
ALTER TABLE STOCK3 ADD CONSTRAINT STOCK3CKC CHECK
(S_W_ID BETWEEN 1601 AND 2400);
SET INTEGRITY FOR STOCK3 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK4 OFF;
ALTER TABLE STOCK4 DROP CONSTRAINT STOCK4CKC;
ALTER TABLE STOCK4 ADD CONSTRAINT STOCK4CKC CHECK
(S_W_ID BETWEEN 2401 AND 3200);
SET INTEGRITY FOR STOCK4 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK5 OFF;
ALTER TABLE STOCK5 DROP CONSTRAINT STOCK5CKC;
ALTER TABLE STOCK5 ADD CONSTRAINT STOCK5CKC CHECK
(S_W_ID BETWEEN 3201 AND 4000);
SET INTEGRITY FOR STOCK5 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK6 OFF;
ALTER TABLE STOCK6 DROP CONSTRAINT STOCK6CKC;
ALTER TABLE STOCK6 ADD CONSTRAINT STOCK6CKC CHECK
(S_W_ID BETWEEN 4001 AND 4800);
SET INTEGRITY FOR STOCK6 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK7 OFF;
ALTER TABLE STOCK7 DROP CONSTRAINT STOCK7CKC;
ALTER TABLE STOCK7 ADD CONSTRAINT STOCK7CKC CHECK
(S_W_ID BETWEEN 4801 AND 5600);
SET INTEGRITY FOR STOCK7 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK8 OFF;
ALTER TABLE STOCK8 DROP CONSTRAINT STOCK8CKC;
ALTER TABLE STOCK8 ADD CONSTRAINT STOCK8CKC CHECK
(S_W_ID BETWEEN 5601 AND 6400);
SET INTEGRITY FOR STOCK8 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK9 OFF;
ALTER TABLE STOCK9 DROP CONSTRAINT STOCK9CKC;
ALTER TABLE STOCK9 ADD CONSTRAINT STOCK9CKC CHECK
(S_W_ID BETWEEN 6401 AND 7200);
SET INTEGRITY FOR STOCK9 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK10 OFF;
ALTER TABLE STOCK10 DROP CONSTRAINT STOCK10CKC;
ALTER TABLE STOCK10 ADD CONSTRAINT STOCK10CKC CHECK
(S_W_ID BETWEEN 7201 AND 8000);
SET INTEGRITY FOR STOCK10 ALL IMMEDIATE UNCHECKED;
connect reset;

```

```

connect to TPCC in share mode;
SET INTEGRITY FOR STOCK11 OFF;
ALTER TABLE STOCK11 DROP CONSTRAINT STOCK11CKC;
ALTER TABLE STOCK11 ADD CONSTRAINT STOCK11CKC CHECK
(S_W_ID BETWEEN 8001 AND 8800);
SET INTEGRITY FOR STOCK11 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK12 OFF;
ALTER TABLE STOCK12 DROP CONSTRAINT STOCK12CKC;
ALTER TABLE STOCK12 ADD CONSTRAINT STOCK12CKC CHECK
(S_W_ID BETWEEN 8801 AND 9600);
SET INTEGRITY FOR STOCK12 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK13 OFF;
ALTER TABLE STOCK13 DROP CONSTRAINT STOCK13CKC;
ALTER TABLE STOCK13 ADD CONSTRAINT STOCK13CKC CHECK
(S_W_ID BETWEEN 9601 AND 10400);
SET INTEGRITY FOR STOCK13 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK14 OFF;
ALTER TABLE STOCK14 DROP CONSTRAINT STOCK14CKC;
ALTER TABLE STOCK14 ADD CONSTRAINT STOCK14CKC CHECK
(S_W_ID BETWEEN 10401 AND 11200);
SET INTEGRITY FOR STOCK14 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK15 OFF;
ALTER TABLE STOCK15 DROP CONSTRAINT STOCK15CKC;
ALTER TABLE STOCK15 ADD CONSTRAINT STOCK15CKC CHECK
(S_W_ID BETWEEN 11201 AND 12000);
SET INTEGRITY FOR STOCK15 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK16 OFF;
ALTER TABLE STOCK16 DROP CONSTRAINT STOCK16CKC;
ALTER TABLE STOCK16 ADD CONSTRAINT STOCK16CKC CHECK
(S_W_ID BETWEEN 12001 AND 12800);
SET INTEGRITY FOR STOCK16 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK17 OFF;
ALTER TABLE STOCK17 DROP CONSTRAINT STOCK17CKC;
ALTER TABLE STOCK17 ADD CONSTRAINT STOCK17CKC CHECK
(S_W_ID BETWEEN 12801 AND 13600);
SET INTEGRITY FOR STOCK17 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK18 OFF;
ALTER TABLE STOCK18 DROP CONSTRAINT STOCK18CKC;
ALTER TABLE STOCK18 ADD CONSTRAINT STOCK18CKC CHECK
(S_W_ID BETWEEN 13601 AND 14400);
SET INTEGRITY FOR STOCK18 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK19 OFF;
ALTER TABLE STOCK19 DROP CONSTRAINT STOCK19CKC;
ALTER TABLE STOCK19 ADD CONSTRAINT STOCK19CKC CHECK
(S_W_ID BETWEEN 14401 AND 15200);
SET INTEGRITY FOR STOCK19 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK20 OFF;
ALTER TABLE STOCK20 DROP CONSTRAINT STOCK20CKC;
ALTER TABLE STOCK20 ADD CONSTRAINT STOCK20CKC CHECK
(S_W_ID BETWEEN 15201 AND 16000);

```

```

SET INTEGRITY FOR STOCK20 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK21 OFF;
ALTER TABLE STOCK21 DROP CONSTRAINT STOCK21CKC;
ALTER TABLE STOCK21 ADD CONSTRAINT STOCK21CKC CHECK
(S_W_ID BETWEEN 16001 AND 16800);
SET INTEGRITY FOR STOCK21 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK22 OFF;
ALTER TABLE STOCK22 DROP CONSTRAINT STOCK22CKC;
ALTER TABLE STOCK22 ADD CONSTRAINT STOCK22CKC CHECK
(S_W_ID BETWEEN 16801 AND 17600);
SET INTEGRITY FOR STOCK22 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK23 OFF;
ALTER TABLE STOCK23 DROP CONSTRAINT STOCK23CKC;
ALTER TABLE STOCK23 ADD CONSTRAINT STOCK23CKC CHECK
(S_W_ID BETWEEN 17601 AND 18400);
SET INTEGRITY FOR STOCK23 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK24 OFF;
ALTER TABLE STOCK24 DROP CONSTRAINT STOCK24CKC;
ALTER TABLE STOCK24 ADD CONSTRAINT STOCK24CKC CHECK
(S_W_ID BETWEEN 18401 AND 19200);
SET INTEGRITY FOR STOCK24 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR STOCK25 OFF;
ALTER TABLE STOCK25 DROP CONSTRAINT STOCK25CKC;
ALTER TABLE STOCK25 ADD CONSTRAINT STOCK25CKC CHECK
(S_W_ID >= 19201);
SET INTEGRITY FOR STOCK25 ALL IMMEDIATE UNCHECKED;
connect reset;

```

crconst_warehouse.ddl

```

connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE1 OFF;
ALTER TABLE WAREHOUSE1 DROP CONSTRAINT
WAREHOUSE1CKC;
ALTER TABLE WAREHOUSE1 ADD CONSTRAINT WAREHOUSE1CKC
CHECK (W_ID BETWEEN 1 AND 800);
SET INTEGRITY FOR WAREHOUSE1 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE2 OFF;
ALTER TABLE WAREHOUSE2 DROP CONSTRAINT
WAREHOUSE2CKC;
ALTER TABLE WAREHOUSE2 ADD CONSTRAINT WAREHOUSE2CKC
CHECK (W_ID BETWEEN 801 AND 1600);
SET INTEGRITY FOR WAREHOUSE2 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE3 OFF;
ALTER TABLE WAREHOUSE3 DROP CONSTRAINT
WAREHOUSE3CKC;
ALTER TABLE WAREHOUSE3 ADD CONSTRAINT WAREHOUSE3CKC
CHECK (W_ID BETWEEN 1601 AND 2400);
SET INTEGRITY FOR WAREHOUSE3 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE4 OFF;

```

```

ALTER TABLE WAREHOUSE4 DROP CONSTRAINT
WAREHOUSE4CKC;
ALTER TABLE WAREHOUSE4 ADD CONSTRAINT WAREHOUSE4CKC
CHECK (W_ID BETWEEN 2401 AND 3200);
SET INTEGRITY FOR WAREHOUSE4 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE5 OFF;
ALTER TABLE WAREHOUSE5 DROP CONSTRAINT
WAREHOUSE5CKC;
ALTER TABLE WAREHOUSE5 ADD CONSTRAINT WAREHOUSE5CKC
CHECK (W_ID BETWEEN 3201 AND 4000);
SET INTEGRITY FOR WAREHOUSE5 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE6 OFF;
ALTER TABLE WAREHOUSE6 DROP CONSTRAINT
WAREHOUSE6CKC;
ALTER TABLE WAREHOUSE6 ADD CONSTRAINT WAREHOUSE6CKC
CHECK (W_ID BETWEEN 4001 AND 4800);
SET INTEGRITY FOR WAREHOUSE6 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE7 OFF;
ALTER TABLE WAREHOUSE7 DROP CONSTRAINT
WAREHOUSE7CKC;
ALTER TABLE WAREHOUSE7 ADD CONSTRAINT WAREHOUSE7CKC
CHECK (W_ID BETWEEN 4801 AND 5600);
SET INTEGRITY FOR WAREHOUSE7 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE8 OFF;
ALTER TABLE WAREHOUSE8 DROP CONSTRAINT
WAREHOUSE8CKC;
ALTER TABLE WAREHOUSE8 ADD CONSTRAINT WAREHOUSE8CKC
CHECK (W_ID BETWEEN 5601 AND 6400);
SET INTEGRITY FOR WAREHOUSE8 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE9 OFF;
ALTER TABLE WAREHOUSE9 DROP CONSTRAINT
WAREHOUSE9CKC;
ALTER TABLE WAREHOUSE9 ADD CONSTRAINT WAREHOUSE9CKC
CHECK (W_ID BETWEEN 6401 AND 7200);
SET INTEGRITY FOR WAREHOUSE9 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE10 OFF;
ALTER TABLE WAREHOUSE10 DROP CONSTRAINT
WAREHOUSE10CKC;
ALTER TABLE WAREHOUSE10 ADD CONSTRAINT
WAREHOUSE10CKC CHECK (W_ID BETWEEN 7201 AND 8000);
SET INTEGRITY FOR WAREHOUSE10 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE11 OFF;
ALTER TABLE WAREHOUSE11 DROP CONSTRAINT
WAREHOUSE11CKC;
ALTER TABLE WAREHOUSE11 ADD CONSTRAINT
WAREHOUSE11CKC CHECK (W_ID BETWEEN 8001 AND 8800);
SET INTEGRITY FOR WAREHOUSE11 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE12 OFF;
ALTER TABLE WAREHOUSE12 DROP CONSTRAINT
WAREHOUSE12CKC;
ALTER TABLE WAREHOUSE12 ADD CONSTRAINT
WAREHOUSE12CKC CHECK (W_ID BETWEEN 8801 AND 9600);

```

```

SET INTEGRITY FOR WAREHOUSE12 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE13 OFF;
ALTER TABLE WAREHOUSE13 DROP CONSTRAINT
WAREHOUSE13CKC;
ALTER TABLE WAREHOUSE13 ADD CONSTRAINT
WAREHOUSE13CKC CHECK (W_ID BETWEEN 9601 AND 10400);
SET INTEGRITY FOR WAREHOUSE13 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE14 OFF;
ALTER TABLE WAREHOUSE14 DROP CONSTRAINT
WAREHOUSE14CKC;
ALTER TABLE WAREHOUSE14 ADD CONSTRAINT
WAREHOUSE14CKC CHECK (W_ID BETWEEN 10401 AND 11200);
SET INTEGRITY FOR WAREHOUSE14 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE15 OFF;
ALTER TABLE WAREHOUSE15 DROP CONSTRAINT
WAREHOUSE15CKC;
ALTER TABLE WAREHOUSE15 ADD CONSTRAINT
WAREHOUSE15CKC CHECK (W_ID BETWEEN 11201 AND 12000);
SET INTEGRITY FOR WAREHOUSE15 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE16 OFF;
ALTER TABLE WAREHOUSE16 DROP CONSTRAINT
WAREHOUSE16CKC;
ALTER TABLE WAREHOUSE16 ADD CONSTRAINT
WAREHOUSE16CKC CHECK (W_ID BETWEEN 12001 AND 12800);
SET INTEGRITY FOR WAREHOUSE16 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE17 OFF;
ALTER TABLE WAREHOUSE17 DROP CONSTRAINT
WAREHOUSE17CKC;
ALTER TABLE WAREHOUSE17 ADD CONSTRAINT
WAREHOUSE17CKC CHECK (W_ID BETWEEN 12801 AND 13600);
SET INTEGRITY FOR WAREHOUSE17 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE18 OFF;
ALTER TABLE WAREHOUSE18 DROP CONSTRAINT
WAREHOUSE18CKC;
ALTER TABLE WAREHOUSE18 ADD CONSTRAINT
WAREHOUSE18CKC CHECK (W_ID BETWEEN 13601 AND 14400);
SET INTEGRITY FOR WAREHOUSE18 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE19 OFF;
ALTER TABLE WAREHOUSE19 DROP CONSTRAINT
WAREHOUSE19CKC;
ALTER TABLE WAREHOUSE19 ADD CONSTRAINT
WAREHOUSE19CKC CHECK (W_ID BETWEEN 14401 AND 15200);
SET INTEGRITY FOR WAREHOUSE19 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE20 OFF;
ALTER TABLE WAREHOUSE20 DROP CONSTRAINT
WAREHOUSE20CKC;
ALTER TABLE WAREHOUSE20 ADD CONSTRAINT
WAREHOUSE20CKC CHECK (W_ID BETWEEN 15201 AND 16000);
SET INTEGRITY FOR WAREHOUSE20 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE21 OFF;

```

```

ALTER TABLE WAREHOUSE21 DROP CONSTRAINT
WAREHOUSE21CKC;
ALTER TABLE WAREHOUSE21 ADD CONSTRAINT
WAREHOUSE21CKC CHECK (W_ID BETWEEN 16001 AND 16800);
SET INTEGRITY FOR WAREHOUSE21 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE22 OFF;
ALTER TABLE WAREHOUSE22 DROP CONSTRAINT
WAREHOUSE22CKC;
ALTER TABLE WAREHOUSE22 ADD CONSTRAINT
WAREHOUSE22CKC CHECK (W_ID BETWEEN 16801 AND 17600);
SET INTEGRITY FOR WAREHOUSE22 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE23 OFF;
ALTER TABLE WAREHOUSE23 DROP CONSTRAINT
WAREHOUSE23CKC;
ALTER TABLE WAREHOUSE23 ADD CONSTRAINT
WAREHOUSE23CKC CHECK (W_ID BETWEEN 17601 AND 18400);
SET INTEGRITY FOR WAREHOUSE23 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE24 OFF;
ALTER TABLE WAREHOUSE24 DROP CONSTRAINT
WAREHOUSE24CKC;
ALTER TABLE WAREHOUSE24 ADD CONSTRAINT
WAREHOUSE24CKC CHECK (W_ID BETWEEN 18401 AND 19200);
SET INTEGRITY FOR WAREHOUSE24 ALL IMMEDIATE UNCHECKED;
connect reset;
connect to TPCC in share mode;
SET INTEGRITY FOR WAREHOUSE25 OFF;
ALTER TABLE WAREHOUSE25 DROP CONSTRAINT
WAREHOUSE25CKC;
ALTER TABLE WAREHOUSE25 ADD CONSTRAINT
WAREHOUSE25CKC CHECK (W_ID >= 19201);
SET INTEGRITY FOR WAREHOUSE25 ALL IMMEDIATE UNCHECKED;
connect reset;

```

criidx_cust_idxb.ddl

```

connect to TPCC in share mode;
DROP INDEX CUST_IDXB1;
CREATE INDEX CUST_IDXB1
    ON CUSTOMER1(C_LAST, C_W_ID, C_D_ID, C_FIRST,
C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB2;
CREATE INDEX CUST_IDXB2
    ON CUSTOMER2(C_LAST, C_W_ID, C_D_ID, C_FIRST,
C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB3;
CREATE INDEX CUST_IDXB3
    ON CUSTOMER3(C_LAST, C_W_ID, C_D_ID, C_FIRST,
C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB4;
CREATE INDEX CUST_IDXB4
    ON CUSTOMER4(C_LAST, C_W_ID, C_D_ID, C_FIRST,
C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB5;

```

```

CREATE INDEX CUST_IDXB5
    ON CUSTOMER5(C_LAST, C_W_ID, C_D_ID, C_FIRST,
C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB6;
CREATE INDEX CUST_IDXB6
    ON CUSTOMER6(C_LAST, C_W_ID, C_D_ID, C_FIRST,
C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB7;
CREATE INDEX CUST_IDXB7
    ON CUSTOMER7(C_LAST, C_W_ID, C_D_ID, C_FIRST,
C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB8;
CREATE INDEX CUST_IDXB8
    ON CUSTOMER8(C_LAST, C_W_ID, C_D_ID, C_FIRST,
C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB9;
CREATE INDEX CUST_IDXB9
    ON CUSTOMER9(C_LAST, C_W_ID, C_D_ID, C_FIRST,
C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB10;
CREATE INDEX CUST_IDXB10
    ON CUSTOMER10(C_LAST, C_W_ID, C_D_ID, C_FIRST,
C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB11;
CREATE INDEX CUST_IDXB11
    ON CUSTOMER11(C_LAST, C_W_ID, C_D_ID, C_FIRST,
C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB12;
CREATE INDEX CUST_IDXB12
    ON CUSTOMER12(C_LAST, C_W_ID, C_D_ID, C_FIRST,
C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB13;
CREATE INDEX CUST_IDXB13
    ON CUSTOMER13(C_LAST, C_W_ID, C_D_ID, C_FIRST,
C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB14;
CREATE INDEX CUST_IDXB14
    ON CUSTOMER14(C_LAST, C_W_ID, C_D_ID, C_FIRST,
C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB15;
CREATE INDEX CUST_IDXB15
    ON CUSTOMER15(C_LAST, C_W_ID, C_D_ID, C_FIRST,
C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB16;
CREATE INDEX CUST_IDXB16

```

```

        ON CUSTOMER16(C_LAST, C_W_ID, C_D_ID, C_FIRST,
C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB17;
CREATE INDEX CUST_IDXB17
        ON CUSTOMER17(C_LAST, C_W_ID, C_D_ID, C_FIRST,
C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB18;
CREATE INDEX CUST_IDXB18
        ON CUSTOMER18(C_LAST, C_W_ID, C_D_ID, C_FIRST,
C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB19;
CREATE INDEX CUST_IDXB19
        ON CUSTOMER19(C_LAST, C_W_ID, C_D_ID, C_FIRST,
C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB20;
CREATE INDEX CUST_IDXB20
        ON CUSTOMER20(C_LAST, C_W_ID, C_D_ID, C_FIRST,
C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB21;
CREATE INDEX CUST_IDXB21
        ON CUSTOMER21(C_LAST, C_W_ID, C_D_ID, C_FIRST,
C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB22;
CREATE INDEX CUST_IDXB22
        ON CUSTOMER22(C_LAST, C_W_ID, C_D_ID, C_FIRST,
C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB23;
CREATE INDEX CUST_IDXB23
        ON CUSTOMER23(C_LAST, C_W_ID, C_D_ID, C_FIRST,
C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB24;
CREATE INDEX CUST_IDXB24
        ON CUSTOMER24(C_LAST, C_W_ID, C_D_ID, C_FIRST,
C_ID) PCTFREE 0;
connect reset;
connect to TPCC in share mode;
DROP INDEX CUST_IDXB25;
CREATE INDEX CUST_IDXB25
        ON CUSTOMER25(C_LAST, C_W_ID, C_D_ID, C_FIRST,
C_ID) PCTFREE 0;
connect reset;

```

cridx_ordr_idxb.ddl

```

connect to TPCC in share mode;
DROP INDEX ORDR_IDXB1;
CREATE INDEX ORDR_IDXB1
        ON ORDERS1(O_C_ID, O_W_ID, O_D_ID, O_ID DESC)
PCTFREE 20 LEVEL2 PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB2;

```

```

CREATE INDEX ORDR_IDXB2
        ON ORDERS2(O_C_ID, O_W_ID, O_D_ID, O_ID DESC)
PCTFREE 20 LEVEL2 PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB3;
CREATE INDEX ORDR_IDXB3
        ON ORDERS3(O_C_ID, O_W_ID, O_D_ID, O_ID DESC)
PCTFREE 20 LEVEL2 PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB4;
CREATE INDEX ORDR_IDXB4
        ON ORDERS4(O_C_ID, O_W_ID, O_D_ID, O_ID DESC)
PCTFREE 20 LEVEL2 PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB5;
CREATE INDEX ORDR_IDXB5
        ON ORDERS5(O_C_ID, O_W_ID, O_D_ID, O_ID DESC)
PCTFREE 20 LEVEL2 PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB6;
CREATE INDEX ORDR_IDXB6
        ON ORDERS6(O_C_ID, O_W_ID, O_D_ID, O_ID DESC)
PCTFREE 20 LEVEL2 PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB7;
CREATE INDEX ORDR_IDXB7
        ON ORDERS7(O_C_ID, O_W_ID, O_D_ID, O_ID DESC)
PCTFREE 20 LEVEL2 PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB8;
CREATE INDEX ORDR_IDXB8
        ON ORDERS8(O_C_ID, O_W_ID, O_D_ID, O_ID DESC)
PCTFREE 20 LEVEL2 PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB9;
CREATE INDEX ORDR_IDXB9
        ON ORDERS9(O_C_ID, O_W_ID, O_D_ID, O_ID DESC)
PCTFREE 20 LEVEL2 PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB10;
CREATE INDEX ORDR_IDXB10
        ON ORDERS10(O_C_ID, O_W_ID, O_D_ID, O_ID DESC)
PCTFREE 20 LEVEL2 PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB11;
CREATE INDEX ORDR_IDXB11
        ON ORDERS11(O_C_ID, O_W_ID, O_D_ID, O_ID DESC)
PCTFREE 20 LEVEL2 PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB12;
CREATE INDEX ORDR_IDXB12
        ON ORDERS12(O_C_ID, O_W_ID, O_D_ID, O_ID DESC)
PCTFREE 20 LEVEL2 PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB13;
CREATE INDEX ORDR_IDXB13

```

```

        ON ORDERS13(O_C_ID, O_W_ID, O_D_ID, O_ID DESC)
PCTFREE 20 LEVEL2 PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB14;
CREATE INDEX ORDR_IDXB14
        ON ORDERS14(O_C_ID, O_W_ID, O_D_ID, O_ID DESC)
PCTFREE 20 LEVEL2 PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB15;
CREATE INDEX ORDR_IDXB15
        ON ORDERS15(O_C_ID, O_W_ID, O_D_ID, O_ID DESC)
PCTFREE 20 LEVEL2 PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB16;
CREATE INDEX ORDR_IDXB16
        ON ORDERS16(O_C_ID, O_W_ID, O_D_ID, O_ID DESC)
PCTFREE 20 LEVEL2 PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB17;
CREATE INDEX ORDR_IDXB17
        ON ORDERS17(O_C_ID, O_W_ID, O_D_ID, O_ID DESC)
PCTFREE 20 LEVEL2 PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB18;
CREATE INDEX ORDR_IDXB18
        ON ORDERS18(O_C_ID, O_W_ID, O_D_ID, O_ID DESC)
PCTFREE 20 LEVEL2 PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB19;
CREATE INDEX ORDR_IDXB19
        ON ORDERS19(O_C_ID, O_W_ID, O_D_ID, O_ID DESC)
PCTFREE 20 LEVEL2 PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB20;
CREATE INDEX ORDR_IDXB20
        ON ORDERS20(O_C_ID, O_W_ID, O_D_ID, O_ID DESC)
PCTFREE 20 LEVEL2 PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB21;
CREATE INDEX ORDR_IDXB21
        ON ORDERS21(O_C_ID, O_W_ID, O_D_ID, O_ID DESC)
PCTFREE 20 LEVEL2 PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB22;
CREATE INDEX ORDR_IDXB22
        ON ORDERS22(O_C_ID, O_W_ID, O_D_ID, O_ID DESC)
PCTFREE 20 LEVEL2 PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB23;
CREATE INDEX ORDR_IDXB23
        ON ORDERS23(O_C_ID, O_W_ID, O_D_ID, O_ID DESC)
PCTFREE 20 LEVEL2 PCTFREE 20;
connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB24;
CREATE INDEX ORDR_IDXB24
        ON ORDERS24(O_C_ID, O_W_ID, O_D_ID, O_ID DESC)
PCTFREE 20 LEVEL2 PCTFREE 20;

```

```

connect reset;
connect to TPCC in share mode;
DROP INDEX ORDR_IDXB25;
CREATE INDEX ORDR_IDXB25
        ON ORDERS25(O_C_ID, O_W_ID, O_D_ID, O_ID DESC)
PCTFREE 20 LEVEL2 PCTFREE 20;
connect reset;

```

crtb_customer.ddl

```

connect to TPCC in share mode;
DROP TABLE CUSTOMER1;
CREATE TABLE CUSTOMER1
(
    C_ID      INTEGER      NOT NULL,
    C_STATE   CHAR(2)     NOT NULL,
    C_ZIP     CHAR(9)     NOT NULL,
    C_PHONE   CHAR(16)    NOT NULL,
    C_SINCE   BIGINT      NOT NULL,
    C_CREDIT_LIM BIGINT    NOT NULL,
    C_MIDDLE  CHAR(2)     NOT NULL,
    C_CREDIT  CHAR(2)     NOT NULL,
    C_DISCOUNT INTEGER    NOT NULL,
    C_DATA    VARCHAR(500) NOT NULL,
    C_LAST    VARCHAR(16)  NOT NULL,
    C_FIRST   VARCHAR(16)  NOT NULL,
    C_STREET_1 VARCHAR(20) NOT NULL,
    C_STREET_2 VARCHAR(20) NOT NULL,
    C_CITY    VARCHAR(20)  NOT NULL,
    C_D_ID    SMALLINT    NOT NULL,
    C_W_ID    INTEGER      NOT NULL,
    C_DELIVERY_CNT INTEGER  NOT NULL,
    C_BALANCE BIGINT      NOT NULL,
    C_YTD_PAYMENT BIGINT   NOT NULL,
    C_PAYMENT_CNT INTEGER  NOT NULL
)
IN CST_001
INDEX IN CSTI_001
ORGANIZE BY KEY SEQUENCE (
    C_ID STARTING FROM 1 ENDING AT 3000,
    C_W_ID STARTING FROM 1 ENDING AT 800,
    C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER2;
CREATE TABLE CUSTOMER2
(
    C_ID      INTEGER      NOT NULL,
    C_STATE   CHAR(2)     NOT NULL,
    C_ZIP     CHAR(9)     NOT NULL,
    C_PHONE   CHAR(16)    NOT NULL,
    C_SINCE   BIGINT      NOT NULL,
    C_CREDIT_LIM BIGINT    NOT NULL,
    C_MIDDLE  CHAR(2)     NOT NULL,
    C_CREDIT  CHAR(2)     NOT NULL,
    C_DISCOUNT INTEGER    NOT NULL,
    C_DATA    VARCHAR(500) NOT NULL,
    C_LAST    VARCHAR(16)  NOT NULL,
    C_FIRST   VARCHAR(16)  NOT NULL,
    C_STREET_1 VARCHAR(20) NOT NULL,
    C_STREET_2 VARCHAR(20) NOT NULL,
    C_CITY    VARCHAR(20)  NOT NULL,
    C_D_ID    SMALLINT    NOT NULL,
    C_W_ID    INTEGER      NOT NULL,
    C_DELIVERY_CNT INTEGER  NOT NULL,

```



```

C_BALANCE BIGINT NOT NULL,
C_YTD_PAYMENT BIGINT NOT NULL,
C_PAYMENT_CNT INTEGER NOT NULL
)
IN CST_002
INDEX IN CSTI_002
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 801 ENDING AT 1600,
C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

connect reset;

connect to TPCC in share mode;

DROP TABLE CUSTOMER3;

CREATE TABLE CUSTOMER3

```

(
C_ID INTEGER NOT NULL,
C_STATE CHAR(2) NOT NULL,
C_ZIP CHAR(9) NOT NULL,
C_PHONE CHAR(16) NOT NULL,
C_SINCE BIGINT NOT NULL,
C_CREDIT_LIM BIGINT NOT NULL,
C_MIDDLE CHAR(2) NOT NULL,
C_CREDIT CHAR(2) NOT NULL,
C_DISCOUNT INTEGER NOT NULL,
C_DATA VARCHAR(500) NOT NULL,
C_LAST VARCHAR(16) NOT NULL,
C_FIRST VARCHAR(16) NOT NULL,
C_STREET_1 VARCHAR(20) NOT NULL,
C_STREET_2 VARCHAR(20) NOT NULL,
C_CITY VARCHAR(20) NOT NULL,
C_D_ID SMALLINT NOT NULL,
C_W_ID INTEGER NOT NULL,
C_DELIVERY_CNT INTEGER NOT NULL,
C_BALANCE BIGINT NOT NULL,
C_YTD_PAYMENT BIGINT NOT NULL,
C_PAYMENT_CNT INTEGER NOT NULL
)
IN CST_003
INDEX IN CSTI_003
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 1601 ENDING AT 2400,
C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

connect reset;

connect to TPCC in share mode;

DROP TABLE CUSTOMER4;

CREATE TABLE CUSTOMER4

```

(
C_ID INTEGER NOT NULL,
C_STATE CHAR(2) NOT NULL,
C_ZIP CHAR(9) NOT NULL,
C_PHONE CHAR(16) NOT NULL,
C_SINCE BIGINT NOT NULL,
C_CREDIT_LIM BIGINT NOT NULL,
C_MIDDLE CHAR(2) NOT NULL,
C_CREDIT CHAR(2) NOT NULL,
C_DISCOUNT INTEGER NOT NULL,
C_DATA VARCHAR(500) NOT NULL,
C_LAST VARCHAR(16) NOT NULL,
C_FIRST VARCHAR(16) NOT NULL,
C_STREET_1 VARCHAR(20) NOT NULL,
C_STREET_2 VARCHAR(20) NOT NULL,
C_CITY VARCHAR(20) NOT NULL,
C_D_ID SMALLINT NOT NULL,

```

```

C_W_ID INTEGER NOT NULL,
C_DELIVERY_CNT INTEGER NOT NULL,
C_BALANCE BIGINT NOT NULL,
C_YTD_PAYMENT BIGINT NOT NULL,
C_PAYMENT_CNT INTEGER NOT NULL
)
IN CST_004
INDEX IN CSTI_004
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 2401 ENDING AT 3200,
C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

connect reset;

connect to TPCC in share mode;

DROP TABLE CUSTOMER5;

CREATE TABLE CUSTOMER5

```

(
C_ID INTEGER NOT NULL,
C_STATE CHAR(2) NOT NULL,
C_ZIP CHAR(9) NOT NULL,
C_PHONE CHAR(16) NOT NULL,
C_SINCE BIGINT NOT NULL,
C_CREDIT_LIM BIGINT NOT NULL,
C_MIDDLE CHAR(2) NOT NULL,
C_CREDIT CHAR(2) NOT NULL,
C_DISCOUNT INTEGER NOT NULL,
C_DATA VARCHAR(500) NOT NULL,
C_LAST VARCHAR(16) NOT NULL,
C_FIRST VARCHAR(16) NOT NULL,
C_STREET_1 VARCHAR(20) NOT NULL,
C_STREET_2 VARCHAR(20) NOT NULL,
C_CITY VARCHAR(20) NOT NULL,
C_D_ID SMALLINT NOT NULL,
C_W_ID INTEGER NOT NULL,
C_DELIVERY_CNT INTEGER NOT NULL,
C_BALANCE BIGINT NOT NULL,
C_YTD_PAYMENT BIGINT NOT NULL,
C_PAYMENT_CNT INTEGER NOT NULL
)
IN CST_005
INDEX IN CSTI_005
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 3201 ENDING AT 4000,
C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

connect reset;

connect to TPCC in share mode;

DROP TABLE CUSTOMER6;

CREATE TABLE CUSTOMER6

```

(
C_ID INTEGER NOT NULL,
C_STATE CHAR(2) NOT NULL,
C_ZIP CHAR(9) NOT NULL,
C_PHONE CHAR(16) NOT NULL,
C_SINCE BIGINT NOT NULL,
C_CREDIT_LIM BIGINT NOT NULL,
C_MIDDLE CHAR(2) NOT NULL,
C_CREDIT CHAR(2) NOT NULL,
C_DISCOUNT INTEGER NOT NULL,
C_DATA VARCHAR(500) NOT NULL,
C_LAST VARCHAR(16) NOT NULL,
C_FIRST VARCHAR(16) NOT NULL,
C_STREET_1 VARCHAR(20) NOT NULL,
C_STREET_2 VARCHAR(20) NOT NULL,

```

```

C_CITY    VARCHAR(20) NOT NULL,
C_D_ID    SMALLINT  NOT NULL,
C_W_ID    INTEGER    NOT NULL,
C_DELIVERY_CNT INTEGER NOT NULL,
C_BALANCE BIGINT     NOT NULL,
C_YTD_PAYMENT BIGINT NOT NULL,
C_PAYMENT_CNT INTEGER NOT NULL
)
IN CST_006
INDEX IN CSTI_006
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 4001 ENDING AT 4800,
C_D_ID STARTING FROM 1 ENDING AT 10
)
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER7;
CREATE TABLE CUSTOMER7

```

```

(
C_ID      INTEGER    NOT NULL,
C_STATE   CHAR(2)    NOT NULL,
C_ZIP     CHAR(9)    NOT NULL,
C_PHONE   CHAR(16)   NOT NULL,
C_SINCE   BIGINT    NOT NULL,
C_CREDIT_LIM BIGINT  NOT NULL,
C_MIDDLE  CHAR(2)    NOT NULL,
C_CREDIT  CHAR(2)    NOT NULL,
C_DISCOUNT INTEGER  NOT NULL,
C_DATA    VARCHAR(500) NOT NULL,
C_LAST    VARCHAR(16) NOT NULL,
C_FIRST   VARCHAR(16) NOT NULL,
C_STREET_1 VARCHAR(20) NOT NULL,
C_STREET_2 VARCHAR(20) NOT NULL,
C_CITY    VARCHAR(20) NOT NULL,
C_D_ID    SMALLINT  NOT NULL,
C_W_ID    INTEGER    NOT NULL,
C_DELIVERY_CNT INTEGER NOT NULL,
C_BALANCE BIGINT     NOT NULL,
C_YTD_PAYMENT BIGINT NOT NULL,
C_PAYMENT_CNT INTEGER NOT NULL
)
IN CST_007
INDEX IN CSTI_007
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 4801 ENDING AT 5600,
C_D_ID STARTING FROM 1 ENDING AT 10
)
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER8;
CREATE TABLE CUSTOMER8

```

```

(
C_ID      INTEGER    NOT NULL,
C_STATE   CHAR(2)    NOT NULL,
C_ZIP     CHAR(9)    NOT NULL,
C_PHONE   CHAR(16)   NOT NULL,
C_SINCE   BIGINT    NOT NULL,
C_CREDIT_LIM BIGINT  NOT NULL,
C_MIDDLE  CHAR(2)    NOT NULL,
C_CREDIT  CHAR(2)    NOT NULL,
C_DISCOUNT INTEGER  NOT NULL,
C_DATA    VARCHAR(500) NOT NULL,
C_LAST    VARCHAR(16) NOT NULL,
C_FIRST   VARCHAR(16) NOT NULL,

```

```

C_STREET_1 VARCHAR(20) NOT NULL,
C_STREET_2 VARCHAR(20) NOT NULL,
C_CITY    VARCHAR(20) NOT NULL,
C_D_ID    SMALLINT  NOT NULL,
C_W_ID    INTEGER    NOT NULL,
C_DELIVERY_CNT INTEGER NOT NULL,
C_BALANCE BIGINT     NOT NULL,
C_YTD_PAYMENT BIGINT NOT NULL,
C_PAYMENT_CNT INTEGER NOT NULL
)
IN CST_008
INDEX IN CSTI_008
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 5601 ENDING AT 6400,
C_D_ID STARTING FROM 1 ENDING AT 10
)
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER9;
CREATE TABLE CUSTOMER9

```

```

(
C_ID      INTEGER    NOT NULL,
C_STATE   CHAR(2)    NOT NULL,
C_ZIP     CHAR(9)    NOT NULL,
C_PHONE   CHAR(16)   NOT NULL,
C_SINCE   BIGINT    NOT NULL,
C_CREDIT_LIM BIGINT  NOT NULL,
C_MIDDLE  CHAR(2)    NOT NULL,
C_CREDIT  CHAR(2)    NOT NULL,
C_DISCOUNT INTEGER  NOT NULL,
C_DATA    VARCHAR(500) NOT NULL,
C_LAST    VARCHAR(16) NOT NULL,
C_FIRST   VARCHAR(16) NOT NULL,
C_STREET_1 VARCHAR(20) NOT NULL,
C_STREET_2 VARCHAR(20) NOT NULL,
C_CITY    VARCHAR(20) NOT NULL,
C_D_ID    SMALLINT  NOT NULL,
C_W_ID    INTEGER    NOT NULL,
C_DELIVERY_CNT INTEGER NOT NULL,
C_BALANCE BIGINT     NOT NULL,
C_YTD_PAYMENT BIGINT NOT NULL,
C_PAYMENT_CNT INTEGER NOT NULL
)
IN CST_009
INDEX IN CSTI_009
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 6401 ENDING AT 7200,
C_D_ID STARTING FROM 1 ENDING AT 10
)
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE CUSTOMER10;
CREATE TABLE CUSTOMER10

```

```

(
C_ID      INTEGER    NOT NULL,
C_STATE   CHAR(2)    NOT NULL,
C_ZIP     CHAR(9)    NOT NULL,
C_PHONE   CHAR(16)   NOT NULL,
C_SINCE   BIGINT    NOT NULL,
C_CREDIT_LIM BIGINT  NOT NULL,
C_MIDDLE  CHAR(2)    NOT NULL,
C_CREDIT  CHAR(2)    NOT NULL,
C_DISCOUNT INTEGER  NOT NULL,
C_DATA    VARCHAR(500) NOT NULL,

```

```

C_LAST    VARCHAR(16) NOT NULL,
C_FIRST   VARCHAR(16) NOT NULL,
C_STREET_1 VARCHAR(20) NOT NULL,
C_STREET_2 VARCHAR(20) NOT NULL,
C_CITY    VARCHAR(20) NOT NULL,
C_D_ID    SMALLINT  NOT NULL,
C_W_ID    INTEGER   NOT NULL,
C_DELIVERY_CNT INTEGER NOT NULL,
C_BALANCE BIGINT   NOT NULL,
C_YTD_PAYMENT BIGINT NOT NULL,
C_PAYMENT_CNT INTEGER NOT NULL
)
IN CST_010
INDEX IN CSTI_010
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 7201 ENDING AT 8000,
C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

connect reset;

connect to TPCC in share mode;

DROP TABLE CUSTOMER11;

CREATE TABLE CUSTOMER11

```

(
C_ID      INTEGER   NOT NULL,
C_STATE   CHAR(2)   NOT NULL,
C_ZIP     CHAR(9)   NOT NULL,
C_PHONE   CHAR(16)  NOT NULL,
C_SINCE   BIGINT   NOT NULL,
C_CREDIT_LIM BIGINT NOT NULL,
C_MIDDLE  CHAR(2)   NOT NULL,
C_CREDIT  CHAR(2)   NOT NULL,
C_DISCOUNT INTEGER NOT NULL,
C_DATA    VARCHAR(500) NOT NULL,
C_LAST    VARCHAR(16) NOT NULL,
C_FIRST   VARCHAR(16) NOT NULL,
C_STREET_1 VARCHAR(20) NOT NULL,
C_STREET_2 VARCHAR(20) NOT NULL,
C_CITY    VARCHAR(20) NOT NULL,
C_D_ID    SMALLINT  NOT NULL,
C_W_ID    INTEGER   NOT NULL,
C_DELIVERY_CNT INTEGER NOT NULL,
C_BALANCE BIGINT   NOT NULL,
C_YTD_PAYMENT BIGINT NOT NULL,
C_PAYMENT_CNT INTEGER NOT NULL
)
IN CST_011
INDEX IN CSTI_011
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 8001 ENDING AT 8800,
C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

connect reset;

connect to TPCC in share mode;

DROP TABLE CUSTOMER12;

CREATE TABLE CUSTOMER12

```

(
C_ID      INTEGER   NOT NULL,
C_STATE   CHAR(2)   NOT NULL,
C_ZIP     CHAR(9)   NOT NULL,
C_PHONE   CHAR(16)  NOT NULL,
C_SINCE   BIGINT   NOT NULL,
C_CREDIT_LIM BIGINT NOT NULL,
C_MIDDLE  CHAR(2)   NOT NULL,
C_CREDIT  CHAR(2)   NOT NULL,

```

```

C_DISCOUNT INTEGER NOT NULL,
C_DATA    VARCHAR(500) NOT NULL,
C_LAST    VARCHAR(16) NOT NULL,
C_FIRST   VARCHAR(16) NOT NULL,
C_STREET_1 VARCHAR(20) NOT NULL,
C_STREET_2 VARCHAR(20) NOT NULL,
C_CITY    VARCHAR(20) NOT NULL,
C_D_ID    SMALLINT  NOT NULL,
C_W_ID    INTEGER   NOT NULL,
C_DELIVERY_CNT INTEGER NOT NULL,
C_BALANCE BIGINT   NOT NULL,
C_YTD_PAYMENT BIGINT NOT NULL,
C_PAYMENT_CNT INTEGER NOT NULL
)
IN CST_012
INDEX IN CSTI_012
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 8801 ENDING AT 9600,
C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

connect reset;

connect to TPCC in share mode;

DROP TABLE CUSTOMER13;

CREATE TABLE CUSTOMER13

```

(
C_ID      INTEGER   NOT NULL,
C_STATE   CHAR(2)   NOT NULL,
C_ZIP     CHAR(9)   NOT NULL,
C_PHONE   CHAR(16)  NOT NULL,
C_SINCE   BIGINT   NOT NULL,
C_CREDIT_LIM BIGINT NOT NULL,
C_MIDDLE  CHAR(2)   NOT NULL,
C_CREDIT  CHAR(2)   NOT NULL,
C_DISCOUNT INTEGER NOT NULL,
C_DATA    VARCHAR(500) NOT NULL,
C_LAST    VARCHAR(16) NOT NULL,
C_FIRST   VARCHAR(16) NOT NULL,
C_STREET_1 VARCHAR(20) NOT NULL,
C_STREET_2 VARCHAR(20) NOT NULL,
C_CITY    VARCHAR(20) NOT NULL,
C_D_ID    SMALLINT  NOT NULL,
C_W_ID    INTEGER   NOT NULL,
C_DELIVERY_CNT INTEGER NOT NULL,
C_BALANCE BIGINT   NOT NULL,
C_YTD_PAYMENT BIGINT NOT NULL,
C_PAYMENT_CNT INTEGER NOT NULL
)
IN CST_013
INDEX IN CSTI_013
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 9601 ENDING AT 10400,
C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

connect reset;

connect to TPCC in share mode;

DROP TABLE CUSTOMER14;

CREATE TABLE CUSTOMER14

```

(
C_ID      INTEGER   NOT NULL,
C_STATE   CHAR(2)   NOT NULL,
C_ZIP     CHAR(9)   NOT NULL,
C_PHONE   CHAR(16)  NOT NULL,
C_SINCE   BIGINT   NOT NULL,
C_CREDIT_LIM BIGINT NOT NULL,

```

```

C_MIDDLE CHAR(2) NOT NULL,
C_CREDIT CHAR(2) NOT NULL,
C_DISCOUNT INTEGER NOT NULL,
C_DATA VARCHAR(500) NOT NULL,
C_LAST VARCHAR(16) NOT NULL,
C_FIRST VARCHAR(16) NOT NULL,
C_STREET_1 VARCHAR(20) NOT NULL,
C_STREET_2 VARCHAR(20) NOT NULL,
C_CITY VARCHAR(20) NOT NULL,
C_D_ID SMALLINT NOT NULL,
C_W_ID INTEGER NOT NULL,
C_DELIVERY_CNT INTEGER NOT NULL,
C_BALANCE BIGINT NOT NULL,
C_YTD_PAYMENT BIGINT NOT NULL,
C_PAYMENT_CNT INTEGER NOT NULL
)
IN CST_014
INDEX IN CSTI_014
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 10401 ENDING AT 11200,
C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

connect reset;

connect to TPCC in share mode;

DROP TABLE CUSTOMER15;

CREATE TABLE CUSTOMER15

```

(
C_ID INTEGER NOT NULL,
C_STATE CHAR(2) NOT NULL,
C_ZIP CHAR(9) NOT NULL,
C_PHONE CHAR(16) NOT NULL,
C_SINCE BIGINT NOT NULL,
C_CREDIT_LIM BIGINT NOT NULL,
C_MIDDLE CHAR(2) NOT NULL,
C_CREDIT CHAR(2) NOT NULL,
C_DISCOUNT INTEGER NOT NULL,
C_DATA VARCHAR(500) NOT NULL,
C_LAST VARCHAR(16) NOT NULL,
C_FIRST VARCHAR(16) NOT NULL,
C_STREET_1 VARCHAR(20) NOT NULL,
C_STREET_2 VARCHAR(20) NOT NULL,
C_CITY VARCHAR(20) NOT NULL,
C_D_ID SMALLINT NOT NULL,
C_W_ID INTEGER NOT NULL,
C_DELIVERY_CNT INTEGER NOT NULL,
C_BALANCE BIGINT NOT NULL,
C_YTD_PAYMENT BIGINT NOT NULL,
C_PAYMENT_CNT INTEGER NOT NULL
)
IN CST_015
INDEX IN CSTI_015
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 11201 ENDING AT 12000,
C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

connect reset;

connect to TPCC in share mode;

DROP TABLE CUSTOMER16;

CREATE TABLE CUSTOMER16

```

(
C_ID INTEGER NOT NULL,
C_STATE CHAR(2) NOT NULL,
C_ZIP CHAR(9) NOT NULL,
C_PHONE CHAR(16) NOT NULL,

```

```

C_SINCE BIGINT NOT NULL,
C_CREDIT_LIM BIGINT NOT NULL,
C_MIDDLE CHAR(2) NOT NULL,
C_CREDIT CHAR(2) NOT NULL,
C_DISCOUNT INTEGER NOT NULL,
C_DATA VARCHAR(500) NOT NULL,
C_LAST VARCHAR(16) NOT NULL,
C_FIRST VARCHAR(16) NOT NULL,
C_STREET_1 VARCHAR(20) NOT NULL,
C_STREET_2 VARCHAR(20) NOT NULL,
C_CITY VARCHAR(20) NOT NULL,
C_D_ID SMALLINT NOT NULL,
C_W_ID INTEGER NOT NULL,
C_DELIVERY_CNT INTEGER NOT NULL,
C_BALANCE BIGINT NOT NULL,
C_YTD_PAYMENT BIGINT NOT NULL,
C_PAYMENT_CNT INTEGER NOT NULL
)
IN CST_016
INDEX IN CSTI_016
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 12001 ENDING AT 12800,
C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

connect reset;

connect to TPCC in share mode;

DROP TABLE CUSTOMER17;

CREATE TABLE CUSTOMER17

```

(
C_ID INTEGER NOT NULL,
C_STATE CHAR(2) NOT NULL,
C_ZIP CHAR(9) NOT NULL,
C_PHONE CHAR(16) NOT NULL,
C_SINCE BIGINT NOT NULL,
C_CREDIT_LIM BIGINT NOT NULL,
C_MIDDLE CHAR(2) NOT NULL,
C_CREDIT CHAR(2) NOT NULL,
C_DISCOUNT INTEGER NOT NULL,
C_DATA VARCHAR(500) NOT NULL,
C_LAST VARCHAR(16) NOT NULL,
C_FIRST VARCHAR(16) NOT NULL,
C_STREET_1 VARCHAR(20) NOT NULL,
C_STREET_2 VARCHAR(20) NOT NULL,
C_CITY VARCHAR(20) NOT NULL,
C_D_ID SMALLINT NOT NULL,
C_W_ID INTEGER NOT NULL,
C_DELIVERY_CNT INTEGER NOT NULL,
C_BALANCE BIGINT NOT NULL,
C_YTD_PAYMENT BIGINT NOT NULL,
C_PAYMENT_CNT INTEGER NOT NULL
)
IN CST_017
INDEX IN CSTI_017
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 12801 ENDING AT 13600,
C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

connect reset;

connect to TPCC in share mode;

DROP TABLE CUSTOMER18;

CREATE TABLE CUSTOMER18

```

(
C_ID INTEGER NOT NULL,
C_STATE CHAR(2) NOT NULL,

```

```

C_ZIP CHAR(9) NOT NULL,
C_PHONE CHAR(16) NOT NULL,
C_SINCE BIGINT NOT NULL,
C_CREDIT_LIM BIGINT NOT NULL,
C_MIDDLE CHAR(2) NOT NULL,
C_CREDIT CHAR(2) NOT NULL,
C_DISCOUNT INTEGER NOT NULL,
C_DATA VARCHAR(500) NOT NULL,
C_LAST VARCHAR(16) NOT NULL,
C_FIRST VARCHAR(16) NOT NULL,
C_STREET_1 VARCHAR(20) NOT NULL,
C_STREET_2 VARCHAR(20) NOT NULL,
C_CITY VARCHAR(20) NOT NULL,
C_D_ID SMALLINT NOT NULL,
C_W_ID INTEGER NOT NULL,
C_DELIVERY_CNT INTEGER NOT NULL,
C_BALANCE BIGINT NOT NULL,
C_YTD_PAYMENT BIGINT NOT NULL,
C_PAYMENT_CNT INTEGER NOT NULL
)
IN CST_018
INDEX IN CSTI_018
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 13601 ENDING AT 14400,
C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

connect reset;

connect to TPCC in share mode;

DROP TABLE CUSTOMER19;

CREATE TABLE CUSTOMER19

```

(
C_ID INTEGER NOT NULL,
C_STATE CHAR(2) NOT NULL,
C_ZIP CHAR(9) NOT NULL,
C_PHONE CHAR(16) NOT NULL,
C_SINCE BIGINT NOT NULL,
C_CREDIT_LIM BIGINT NOT NULL,
C_MIDDLE CHAR(2) NOT NULL,
C_CREDIT CHAR(2) NOT NULL,
C_DISCOUNT INTEGER NOT NULL,
C_DATA VARCHAR(500) NOT NULL,
C_LAST VARCHAR(16) NOT NULL,
C_FIRST VARCHAR(16) NOT NULL,
C_STREET_1 VARCHAR(20) NOT NULL,
C_STREET_2 VARCHAR(20) NOT NULL,
C_CITY VARCHAR(20) NOT NULL,
C_D_ID SMALLINT NOT NULL,
C_W_ID INTEGER NOT NULL,
C_DELIVERY_CNT INTEGER NOT NULL,
C_BALANCE BIGINT NOT NULL,
C_YTD_PAYMENT BIGINT NOT NULL,
C_PAYMENT_CNT INTEGER NOT NULL
)
IN CST_019
INDEX IN CSTI_019
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 14401 ENDING AT 15200,
C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

connect reset;

connect to TPCC in share mode;

DROP TABLE CUSTOMER20;

CREATE TABLE CUSTOMER20

```
(
```

```

C_ID INTEGER NOT NULL,
C_STATE CHAR(2) NOT NULL,
C_ZIP CHAR(9) NOT NULL,
C_PHONE CHAR(16) NOT NULL,
C_SINCE BIGINT NOT NULL,
C_CREDIT_LIM BIGINT NOT NULL,
C_MIDDLE CHAR(2) NOT NULL,
C_CREDIT CHAR(2) NOT NULL,
C_DISCOUNT INTEGER NOT NULL,
C_DATA VARCHAR(500) NOT NULL,
C_LAST VARCHAR(16) NOT NULL,
C_FIRST VARCHAR(16) NOT NULL,
C_STREET_1 VARCHAR(20) NOT NULL,
C_STREET_2 VARCHAR(20) NOT NULL,
C_CITY VARCHAR(20) NOT NULL,
C_D_ID SMALLINT NOT NULL,
C_W_ID INTEGER NOT NULL,
C_DELIVERY_CNT INTEGER NOT NULL,
C_BALANCE BIGINT NOT NULL,
C_YTD_PAYMENT BIGINT NOT NULL,
C_PAYMENT_CNT INTEGER NOT NULL
)
IN CST_020
INDEX IN CSTI_020
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 15201 ENDING AT 16000,
C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

connect reset;

connect to TPCC in share mode;

DROP TABLE CUSTOMER21;

CREATE TABLE CUSTOMER21

```

(
C_ID INTEGER NOT NULL,
C_STATE CHAR(2) NOT NULL,
C_ZIP CHAR(9) NOT NULL,
C_PHONE CHAR(16) NOT NULL,
C_SINCE BIGINT NOT NULL,
C_CREDIT_LIM BIGINT NOT NULL,
C_MIDDLE CHAR(2) NOT NULL,
C_CREDIT CHAR(2) NOT NULL,
C_DISCOUNT INTEGER NOT NULL,
C_DATA VARCHAR(500) NOT NULL,
C_LAST VARCHAR(16) NOT NULL,
C_FIRST VARCHAR(16) NOT NULL,
C_STREET_1 VARCHAR(20) NOT NULL,
C_STREET_2 VARCHAR(20) NOT NULL,
C_CITY VARCHAR(20) NOT NULL,
C_D_ID SMALLINT NOT NULL,
C_W_ID INTEGER NOT NULL,
C_DELIVERY_CNT INTEGER NOT NULL,
C_BALANCE BIGINT NOT NULL,
C_YTD_PAYMENT BIGINT NOT NULL,
C_PAYMENT_CNT INTEGER NOT NULL
)
IN CST_021
INDEX IN CSTI_021
ORGANIZE BY KEY SEQUENCE (
C_ID STARTING FROM 1 ENDING AT 3000,
C_W_ID STARTING FROM 16001 ENDING AT 16800,
C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

connect reset;

connect to TPCC in share mode;

DROP TABLE CUSTOMER22;

```

CREATE TABLE CUSTOMER22
(
  C_ID      INTEGER    NOT NULL,
  C_STATE   CHAR(2)    NOT NULL,
  C_ZIP     CHAR(9)    NOT NULL,
  C_PHONE   CHAR(16)   NOT NULL,
  C_SINCE   BIGINT     NOT NULL,
  C_CREDIT_LIM BIGINT  NOT NULL,
  C_MIDDLE  CHAR(2)    NOT NULL,
  C_CREDIT  CHAR(2)    NOT NULL,
  C_DISCOUNT INTEGER  NOT NULL,
  C_DATA    VARCHAR(500) NOT NULL,
  C_LAST    VARCHAR(16) NOT NULL,
  C_FIRST   VARCHAR(16) NOT NULL,
  C_STREET_1 VARCHAR(20) NOT NULL,
  C_STREET_2 VARCHAR(20) NOT NULL,
  C_CITY    VARCHAR(20) NOT NULL,
  C_D_ID    SMALLINT   NOT NULL,
  C_W_ID    INTEGER    NOT NULL,
  C_DELIVERY_CNT INTEGER NOT NULL,
  C_BALANCE BIGINT     NOT NULL,
  C_YTD_PAYMENT BIGINT NOT NULL,
  C_PAYMENT_CNT INTEGER NOT NULL
)
IN CST_022
INDEX IN CSTI_022
ORGANIZE BY KEY SEQUENCE (
  C_ID STARTING FROM 1 ENDING AT 3000,
  C_W_ID STARTING FROM 16801 ENDING AT 17600,
  C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

connect reset;

```

connect to TPCC in share mode;
DROP TABLE CUSTOMER23;
CREATE TABLE CUSTOMER23

```

```

(
  C_ID      INTEGER    NOT NULL,
  C_STATE   CHAR(2)    NOT NULL,
  C_ZIP     CHAR(9)    NOT NULL,
  C_PHONE   CHAR(16)   NOT NULL,
  C_SINCE   BIGINT     NOT NULL,
  C_CREDIT_LIM BIGINT  NOT NULL,
  C_MIDDLE  CHAR(2)    NOT NULL,
  C_CREDIT  CHAR(2)    NOT NULL,
  C_DISCOUNT INTEGER  NOT NULL,
  C_DATA    VARCHAR(500) NOT NULL,
  C_LAST    VARCHAR(16) NOT NULL,
  C_FIRST   VARCHAR(16) NOT NULL,
  C_STREET_1 VARCHAR(20) NOT NULL,
  C_STREET_2 VARCHAR(20) NOT NULL,
  C_CITY    VARCHAR(20) NOT NULL,
  C_D_ID    SMALLINT   NOT NULL,
  C_W_ID    INTEGER    NOT NULL,
  C_DELIVERY_CNT INTEGER NOT NULL,
  C_BALANCE BIGINT     NOT NULL,
  C_YTD_PAYMENT BIGINT NOT NULL,
  C_PAYMENT_CNT INTEGER NOT NULL
)
IN CST_023
INDEX IN CSTI_023
ORGANIZE BY KEY SEQUENCE (
  C_ID STARTING FROM 1 ENDING AT 3000,
  C_W_ID STARTING FROM 17601 ENDING AT 18400,
  C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

connect reset;

```

connect to TPCC in share mode;
DROP TABLE CUSTOMER24;
CREATE TABLE CUSTOMER24

```

```

(
  C_ID      INTEGER    NOT NULL,
  C_STATE   CHAR(2)    NOT NULL,
  C_ZIP     CHAR(9)    NOT NULL,
  C_PHONE   CHAR(16)   NOT NULL,
  C_SINCE   BIGINT     NOT NULL,
  C_CREDIT_LIM BIGINT  NOT NULL,
  C_MIDDLE  CHAR(2)    NOT NULL,
  C_CREDIT  CHAR(2)    NOT NULL,
  C_DISCOUNT INTEGER  NOT NULL,
  C_DATA    VARCHAR(500) NOT NULL,
  C_LAST    VARCHAR(16) NOT NULL,
  C_FIRST   VARCHAR(16) NOT NULL,
  C_STREET_1 VARCHAR(20) NOT NULL,
  C_STREET_2 VARCHAR(20) NOT NULL,
  C_CITY    VARCHAR(20) NOT NULL,
  C_D_ID    SMALLINT   NOT NULL,
  C_W_ID    INTEGER    NOT NULL,
  C_DELIVERY_CNT INTEGER NOT NULL,
  C_BALANCE BIGINT     NOT NULL,
  C_YTD_PAYMENT BIGINT NOT NULL,
  C_PAYMENT_CNT INTEGER NOT NULL
)
IN CST_024
INDEX IN CSTI_024
ORGANIZE BY KEY SEQUENCE (
  C_ID STARTING FROM 1 ENDING AT 3000,
  C_W_ID STARTING FROM 18401 ENDING AT 19200,
  C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

connect reset;

```

connect to TPCC in share mode;
DROP TABLE CUSTOMER25;
CREATE TABLE CUSTOMER25

```

```

(
  C_ID      INTEGER    NOT NULL,
  C_STATE   CHAR(2)    NOT NULL,
  C_ZIP     CHAR(9)    NOT NULL,
  C_PHONE   CHAR(16)   NOT NULL,
  C_SINCE   BIGINT     NOT NULL,
  C_CREDIT_LIM BIGINT  NOT NULL,
  C_MIDDLE  CHAR(2)    NOT NULL,
  C_CREDIT  CHAR(2)    NOT NULL,
  C_DISCOUNT INTEGER  NOT NULL,
  C_DATA    VARCHAR(500) NOT NULL,
  C_LAST    VARCHAR(16) NOT NULL,
  C_FIRST   VARCHAR(16) NOT NULL,
  C_STREET_1 VARCHAR(20) NOT NULL,
  C_STREET_2 VARCHAR(20) NOT NULL,
  C_CITY    VARCHAR(20) NOT NULL,
  C_D_ID    SMALLINT   NOT NULL,
  C_W_ID    INTEGER    NOT NULL,
  C_DELIVERY_CNT INTEGER NOT NULL,
  C_BALANCE BIGINT     NOT NULL,
  C_YTD_PAYMENT BIGINT NOT NULL,
  C_PAYMENT_CNT INTEGER NOT NULL
)
IN CST_025
INDEX IN CSTI_025
ORGANIZE BY KEY SEQUENCE (
  C_ID STARTING FROM 1 ENDING AT 3000,
  C_W_ID STARTING FROM 19201 ENDING AT 20000,
  C_D_ID STARTING FROM 1 ENDING AT 10
)

```

```
ALLOW OVERFLOW;
connect reset;
```

crtb_item.ddl

```
connect to TPCC in share mode;
DROP TABLE ITEM;
CREATE TABLE ITEM
(
  I_NAME CHAR(24) NOT NULL,
  I_PRICE INTEGER NOT NULL,
  I_DATA VARCHAR(50) NOT NULL,
  I_IM_ID INTEGER NOT NULL,
  I_ID INTEGER NOT NULL
)
IN ITM
INDEX IN ITM
ORGANIZE BY KEY SEQUENCE (
  I_ID STARTING FROM 1 ENDING AT 10000
)
ALLOW OVERFLOW;
ALTER TABLE ITEM LOCKSIZE TABLE;
connect reset;
```

crtb_district.ddl

```
connect to TPCC in share mode;
DROP TABLE DISTRICT1;
CREATE TABLE DISTRICT1
(
  D_NEXT_O_ID INTEGER NOT NULL,
  D_TAX INTEGER NOT NULL,
  D_YTD BIGINT NOT NULL,
  D_NAME CHAR(10) NOT NULL,
  D_STREET_1 CHAR(20) NOT NULL,
  D_STREET_2 CHAR(20) NOT NULL,
  D_CITY CHAR(20) NOT NULL,
  D_STATE CHAR(2) NOT NULL,
  D_ZIP CHAR(9) NOT NULL,
  D_ID SMALLINT NOT NULL,
  D_W_ID INTEGER NOT NULL
)
IN DIS_001
INDEX IN DIS_001
ORGANIZE BY KEY SEQUENCE (
  D_ID STARTING FROM 1 ENDING AT 10,
  D_W_ID STARTING FROM 1 ENDING AT 800
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT2;
CREATE TABLE DISTRICT2
(
  D_NEXT_O_ID INTEGER NOT NULL,
  D_TAX INTEGER NOT NULL,
  D_YTD BIGINT NOT NULL,
  D_NAME CHAR(10) NOT NULL,
  D_STREET_1 CHAR(20) NOT NULL,
  D_STREET_2 CHAR(20) NOT NULL,
  D_CITY CHAR(20) NOT NULL,
  D_STATE CHAR(2) NOT NULL,
  D_ZIP CHAR(9) NOT NULL,
  D_ID SMALLINT NOT NULL,
  D_W_ID INTEGER NOT NULL
```

```
)
IN DIS_002
INDEX IN DIS_002
ORGANIZE BY KEY SEQUENCE (
  D_ID STARTING FROM 1 ENDING AT 10,
  D_W_ID STARTING FROM 801 ENDING AT 1600
)
ALLOW OVERFLOW;
```

```
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT3;
CREATE TABLE DISTRICT3
(
  D_NEXT_O_ID INTEGER NOT NULL,
  D_TAX INTEGER NOT NULL,
  D_YTD BIGINT NOT NULL,
  D_NAME CHAR(10) NOT NULL,
  D_STREET_1 CHAR(20) NOT NULL,
  D_STREET_2 CHAR(20) NOT NULL,
  D_CITY CHAR(20) NOT NULL,
  D_STATE CHAR(2) NOT NULL,
  D_ZIP CHAR(9) NOT NULL,
  D_ID SMALLINT NOT NULL,
  D_W_ID INTEGER NOT NULL
)
IN DIS_003
INDEX IN DIS_003
ORGANIZE BY KEY SEQUENCE (
  D_ID STARTING FROM 1 ENDING AT 10,
  D_W_ID STARTING FROM 1601 ENDING AT 2400
)
ALLOW OVERFLOW;
```

```
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT4;
CREATE TABLE DISTRICT4
(
  D_NEXT_O_ID INTEGER NOT NULL,
  D_TAX INTEGER NOT NULL,
  D_YTD BIGINT NOT NULL,
  D_NAME CHAR(10) NOT NULL,
  D_STREET_1 CHAR(20) NOT NULL,
  D_STREET_2 CHAR(20) NOT NULL,
  D_CITY CHAR(20) NOT NULL,
  D_STATE CHAR(2) NOT NULL,
  D_ZIP CHAR(9) NOT NULL,
  D_ID SMALLINT NOT NULL,
  D_W_ID INTEGER NOT NULL
)
IN DIS_004
INDEX IN DIS_004
ORGANIZE BY KEY SEQUENCE (
  D_ID STARTING FROM 1 ENDING AT 10,
  D_W_ID STARTING FROM 2401 ENDING AT 3200
)
ALLOW OVERFLOW;
```

```
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT5;
CREATE TABLE DISTRICT5
(
  D_NEXT_O_ID INTEGER NOT NULL,
  D_TAX INTEGER NOT NULL,
  D_YTD BIGINT NOT NULL,
  D_NAME CHAR(10) NOT NULL,
  D_STREET_1 CHAR(20) NOT NULL,
  D_STREET_2 CHAR(20) NOT NULL,
  D_CITY CHAR(20) NOT NULL,
```

```

D_STATE CHAR(2) NOT NULL,
D_ZIP CHAR(9) NOT NULL,
D_ID SMALLINT NOT NULL,
D_W_ID INTEGER NOT NULL
)
IN DIS_005
INDEX IN DIS_005
ORGANIZE BY KEY SEQUENCE (
D_ID STARTING FROM 1 ENDING AT 10,
D_W_ID STARTING FROM 3201 ENDING AT 4000
)
ALLOW OVERFLOW;

```

connect reset;

connect to TPCC in share mode;

DROP TABLE DISTRICT6;

CREATE TABLE DISTRICT6

```

(
D_NEXT_O_ID INTEGER NOT NULL,
D_TAX INTEGER NOT NULL,
D_YTD BIGINT NOT NULL,
D_NAME CHAR(10) NOT NULL,
D_STREET_1 CHAR(20) NOT NULL,
D_STREET_2 CHAR(20) NOT NULL,
D_CITY CHAR(20) NOT NULL,
D_STATE CHAR(2) NOT NULL,
D_ZIP CHAR(9) NOT NULL,
D_ID SMALLINT NOT NULL,
D_W_ID INTEGER NOT NULL
)
IN DIS_006
INDEX IN DIS_006
ORGANIZE BY KEY SEQUENCE (
D_ID STARTING FROM 1 ENDING AT 10,
D_W_ID STARTING FROM 4001 ENDING AT 4800
)
ALLOW OVERFLOW;

```

connect reset;

connect to TPCC in share mode;

DROP TABLE DISTRICT7;

CREATE TABLE DISTRICT7

```

(
D_NEXT_O_ID INTEGER NOT NULL,
D_TAX INTEGER NOT NULL,
D_YTD BIGINT NOT NULL,
D_NAME CHAR(10) NOT NULL,
D_STREET_1 CHAR(20) NOT NULL,
D_STREET_2 CHAR(20) NOT NULL,
D_CITY CHAR(20) NOT NULL,
D_STATE CHAR(2) NOT NULL,
D_ZIP CHAR(9) NOT NULL,
D_ID SMALLINT NOT NULL,
D_W_ID INTEGER NOT NULL
)
IN DIS_007
INDEX IN DIS_007
ORGANIZE BY KEY SEQUENCE (
D_ID STARTING FROM 1 ENDING AT 10,
D_W_ID STARTING FROM 4801 ENDING AT 5600
)
ALLOW OVERFLOW;

```

connect reset;

connect to TPCC in share mode;

DROP TABLE DISTRICT8;

CREATE TABLE DISTRICT8

```

(
D_NEXT_O_ID INTEGER NOT NULL,
D_TAX INTEGER NOT NULL,
D_YTD BIGINT NOT NULL,

```

```

D_NAME CHAR(10) NOT NULL,
D_STREET_1 CHAR(20) NOT NULL,
D_STREET_2 CHAR(20) NOT NULL,
D_CITY CHAR(20) NOT NULL,
D_STATE CHAR(2) NOT NULL,
D_ZIP CHAR(9) NOT NULL,
D_ID SMALLINT NOT NULL,
D_W_ID INTEGER NOT NULL
)
IN DIS_008
INDEX IN DIS_008
ORGANIZE BY KEY SEQUENCE (
D_ID STARTING FROM 1 ENDING AT 10,
D_W_ID STARTING FROM 5601 ENDING AT 6400
)
ALLOW OVERFLOW;

```

connect reset;

connect to TPCC in share mode;

DROP TABLE DISTRICT9;

CREATE TABLE DISTRICT9

```

(
D_NEXT_O_ID INTEGER NOT NULL,
D_TAX INTEGER NOT NULL,
D_YTD BIGINT NOT NULL,
D_NAME CHAR(10) NOT NULL,
D_STREET_1 CHAR(20) NOT NULL,
D_STREET_2 CHAR(20) NOT NULL,
D_CITY CHAR(20) NOT NULL,
D_STATE CHAR(2) NOT NULL,
D_ZIP CHAR(9) NOT NULL,
D_ID SMALLINT NOT NULL,
D_W_ID INTEGER NOT NULL
)
IN DIS_009
INDEX IN DIS_009
ORGANIZE BY KEY SEQUENCE (
D_ID STARTING FROM 1 ENDING AT 10,
D_W_ID STARTING FROM 6401 ENDING AT 7200
)
ALLOW OVERFLOW;

```

connect reset;

connect to TPCC in share mode;

DROP TABLE DISTRICT10;

CREATE TABLE DISTRICT10

```

(
D_NEXT_O_ID INTEGER NOT NULL,
D_TAX INTEGER NOT NULL,
D_YTD BIGINT NOT NULL,
D_NAME CHAR(10) NOT NULL,
D_STREET_1 CHAR(20) NOT NULL,
D_STREET_2 CHAR(20) NOT NULL,
D_CITY CHAR(20) NOT NULL,
D_STATE CHAR(2) NOT NULL,
D_ZIP CHAR(9) NOT NULL,
D_ID SMALLINT NOT NULL,
D_W_ID INTEGER NOT NULL
)
IN DIS_010
INDEX IN DIS_010
ORGANIZE BY KEY SEQUENCE (
D_ID STARTING FROM 1 ENDING AT 10,
D_W_ID STARTING FROM 7201 ENDING AT 8000
)
ALLOW OVERFLOW;

```

connect reset;

connect to TPCC in share mode;

DROP TABLE DISTRICT11;

CREATE TABLE DISTRICT11


```
(
D_NEXT_O_ID INTEGER NOT NULL,
D_TAX INTEGER NOT NULL,
D_YTD BIGINT NOT NULL,
D_NAME CHAR(10) NOT NULL,
D_STREET_1 CHAR(20) NOT NULL,
D_STREET_2 CHAR(20) NOT NULL,
D_CITY CHAR(20) NOT NULL,
D_STATE CHAR(2) NOT NULL,
D_ZIP CHAR(9) NOT NULL,
D_ID SMALLINT NOT NULL,
D_W_ID INTEGER NOT NULL
)
IN DIS_011
INDEX IN DIS_011
ORGANIZE BY KEY SEQUENCE (
D_ID STARTING FROM 1 ENDING AT 10,
D_W_ID STARTING FROM 8001 ENDING AT 8800
)
ALLOW OVERFLOW;
```

```
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT12;
CREATE TABLE DISTRICT12
```

```
(
D_NEXT_O_ID INTEGER NOT NULL,
D_TAX INTEGER NOT NULL,
D_YTD BIGINT NOT NULL,
D_NAME CHAR(10) NOT NULL,
D_STREET_1 CHAR(20) NOT NULL,
D_STREET_2 CHAR(20) NOT NULL,
D_CITY CHAR(20) NOT NULL,
D_STATE CHAR(2) NOT NULL,
D_ZIP CHAR(9) NOT NULL,
D_ID SMALLINT NOT NULL,
D_W_ID INTEGER NOT NULL
)
IN DIS_012
INDEX IN DIS_012
ORGANIZE BY KEY SEQUENCE (
D_ID STARTING FROM 1 ENDING AT 10,
D_W_ID STARTING FROM 8801 ENDING AT 9600
)
ALLOW OVERFLOW;
```

```
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT13;
CREATE TABLE DISTRICT13
```

```
(
D_NEXT_O_ID INTEGER NOT NULL,
D_TAX INTEGER NOT NULL,
D_YTD BIGINT NOT NULL,
D_NAME CHAR(10) NOT NULL,
D_STREET_1 CHAR(20) NOT NULL,
D_STREET_2 CHAR(20) NOT NULL,
D_CITY CHAR(20) NOT NULL,
D_STATE CHAR(2) NOT NULL,
D_ZIP CHAR(9) NOT NULL,
D_ID SMALLINT NOT NULL,
D_W_ID INTEGER NOT NULL
)
IN DIS_013
INDEX IN DIS_013
ORGANIZE BY KEY SEQUENCE (
D_ID STARTING FROM 1 ENDING AT 10,
D_W_ID STARTING FROM 9601 ENDING AT 10400
)
ALLOW OVERFLOW;
```

```
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT14;
CREATE TABLE DISTRICT14
```

```
(
D_NEXT_O_ID INTEGER NOT NULL,
D_TAX INTEGER NOT NULL,
D_YTD BIGINT NOT NULL,
D_NAME CHAR(10) NOT NULL,
D_STREET_1 CHAR(20) NOT NULL,
D_STREET_2 CHAR(20) NOT NULL,
D_CITY CHAR(20) NOT NULL,
D_STATE CHAR(2) NOT NULL,
D_ZIP CHAR(9) NOT NULL,
D_ID SMALLINT NOT NULL,
D_W_ID INTEGER NOT NULL
)
IN DIS_014
INDEX IN DIS_014
ORGANIZE BY KEY SEQUENCE (
D_ID STARTING FROM 1 ENDING AT 10,
D_W_ID STARTING FROM 10401 ENDING AT 11200
)
ALLOW OVERFLOW;
```

```
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT15;
CREATE TABLE DISTRICT15
```

```
(
D_NEXT_O_ID INTEGER NOT NULL,
D_TAX INTEGER NOT NULL,
D_YTD BIGINT NOT NULL,
D_NAME CHAR(10) NOT NULL,
D_STREET_1 CHAR(20) NOT NULL,
D_STREET_2 CHAR(20) NOT NULL,
D_CITY CHAR(20) NOT NULL,
D_STATE CHAR(2) NOT NULL,
D_ZIP CHAR(9) NOT NULL,
D_ID SMALLINT NOT NULL,
D_W_ID INTEGER NOT NULL
)
IN DIS_015
INDEX IN DIS_015
ORGANIZE BY KEY SEQUENCE (
D_ID STARTING FROM 1 ENDING AT 10,
D_W_ID STARTING FROM 11201 ENDING AT 12000
)
ALLOW OVERFLOW;
```

```
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT16;
CREATE TABLE DISTRICT16
```

```
(
D_NEXT_O_ID INTEGER NOT NULL,
D_TAX INTEGER NOT NULL,
D_YTD BIGINT NOT NULL,
D_NAME CHAR(10) NOT NULL,
D_STREET_1 CHAR(20) NOT NULL,
D_STREET_2 CHAR(20) NOT NULL,
D_CITY CHAR(20) NOT NULL,
D_STATE CHAR(2) NOT NULL,
D_ZIP CHAR(9) NOT NULL,
D_ID SMALLINT NOT NULL,
D_W_ID INTEGER NOT NULL
)
IN DIS_016
INDEX IN DIS_016
ORGANIZE BY KEY SEQUENCE (
```

```

        D_ID STARTING FROM 1 ENDING AT 10,
        D_W_ID STARTING FROM 12001 ENDING AT 12800
    )
    ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT17;
CREATE TABLE DISTRICT17
(
    D_NEXT_O_ID INTEGER    NOT NULL,
    D_TAX        INTEGER    NOT NULL,
    D_YTD        BIGINT     NOT NULL,
    D_NAME       CHAR(10)   NOT NULL,
    D_STREET_1   CHAR(20)   NOT NULL,
    D_STREET_2   CHAR(20)   NOT NULL,
    D_CITY       CHAR(20)   NOT NULL,
    D_STATE      CHAR(2)    NOT NULL,
    D_ZIP        CHAR(9)    NOT NULL,
    D_ID         SMALLINT   NOT NULL,
    D_W_ID       INTEGER    NOT NULL
)
IN DIS_017
INDEX IN DIS_017
ORGANIZE BY KEY SEQUENCE (
    D_ID STARTING FROM 1 ENDING AT 10,
    D_W_ID STARTING FROM 12801 ENDING AT 13600
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT18;
CREATE TABLE DISTRICT18
(
    D_NEXT_O_ID INTEGER    NOT NULL,
    D_TAX        INTEGER    NOT NULL,
    D_YTD        BIGINT     NOT NULL,
    D_NAME       CHAR(10)   NOT NULL,
    D_STREET_1   CHAR(20)   NOT NULL,
    D_STREET_2   CHAR(20)   NOT NULL,
    D_CITY       CHAR(20)   NOT NULL,
    D_STATE      CHAR(2)    NOT NULL,
    D_ZIP        CHAR(9)    NOT NULL,
    D_ID         SMALLINT   NOT NULL,
    D_W_ID       INTEGER    NOT NULL
)
IN DIS_018
INDEX IN DIS_018
ORGANIZE BY KEY SEQUENCE (
    D_ID STARTING FROM 1 ENDING AT 10,
    D_W_ID STARTING FROM 13601 ENDING AT 14400
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT19;
CREATE TABLE DISTRICT19
(
    D_NEXT_O_ID INTEGER    NOT NULL,
    D_TAX        INTEGER    NOT NULL,
    D_YTD        BIGINT     NOT NULL,
    D_NAME       CHAR(10)   NOT NULL,
    D_STREET_1   CHAR(20)   NOT NULL,
    D_STREET_2   CHAR(20)   NOT NULL,
    D_CITY       CHAR(20)   NOT NULL,
    D_STATE      CHAR(2)    NOT NULL,
    D_ZIP        CHAR(9)    NOT NULL,
    D_ID         SMALLINT   NOT NULL,
    D_W_ID       INTEGER    NOT NULL

```

```

    )
    IN DIS_019
    INDEX IN DIS_019
    ORGANIZE BY KEY SEQUENCE (
        D_ID STARTING FROM 1 ENDING AT 10,
        D_W_ID STARTING FROM 14401 ENDING AT 15200
    )
    ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT20;
CREATE TABLE DISTRICT20
(
    D_NEXT_O_ID INTEGER    NOT NULL,
    D_TAX        INTEGER    NOT NULL,
    D_YTD        BIGINT     NOT NULL,
    D_NAME       CHAR(10)   NOT NULL,
    D_STREET_1   CHAR(20)   NOT NULL,
    D_STREET_2   CHAR(20)   NOT NULL,
    D_CITY       CHAR(20)   NOT NULL,
    D_STATE      CHAR(2)    NOT NULL,
    D_ZIP        CHAR(9)    NOT NULL,
    D_ID         SMALLINT   NOT NULL,
    D_W_ID       INTEGER    NOT NULL
)
IN DIS_020
INDEX IN DIS_020
ORGANIZE BY KEY SEQUENCE (
    D_ID STARTING FROM 1 ENDING AT 10,
    D_W_ID STARTING FROM 15201 ENDING AT 16000
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT21;
CREATE TABLE DISTRICT21
(
    D_NEXT_O_ID INTEGER    NOT NULL,
    D_TAX        INTEGER    NOT NULL,
    D_YTD        BIGINT     NOT NULL,
    D_NAME       CHAR(10)   NOT NULL,
    D_STREET_1   CHAR(20)   NOT NULL,
    D_STREET_2   CHAR(20)   NOT NULL,
    D_CITY       CHAR(20)   NOT NULL,
    D_STATE      CHAR(2)    NOT NULL,
    D_ZIP        CHAR(9)    NOT NULL,
    D_ID         SMALLINT   NOT NULL,
    D_W_ID       INTEGER    NOT NULL
)
IN DIS_021
INDEX IN DIS_021
ORGANIZE BY KEY SEQUENCE (
    D_ID STARTING FROM 1 ENDING AT 10,
    D_W_ID STARTING FROM 16001 ENDING AT 16800
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE DISTRICT22;
CREATE TABLE DISTRICT22
(
    D_NEXT_O_ID INTEGER    NOT NULL,
    D_TAX        INTEGER    NOT NULL,
    D_YTD        BIGINT     NOT NULL,
    D_NAME       CHAR(10)   NOT NULL,
    D_STREET_1   CHAR(20)   NOT NULL,
    D_STREET_2   CHAR(20)   NOT NULL,
    D_CITY       CHAR(20)   NOT NULL,

```

```

D_STATE CHAR(2) NOT NULL,
D_ZIP CHAR(9) NOT NULL,
D_ID SMALLINT NOT NULL,
D_W_ID INTEGER NOT NULL
)
IN DIS_022
INDEX IN DIS_022
ORGANIZE BY KEY SEQUENCE (
D_ID STARTING FROM 1 ENDING AT 10,
D_W_ID STARTING FROM 16801 ENDING AT 17600
)
ALLOW OVERFLOW;

```

connect reset;

connect to TPCC in share mode;

DROP TABLE DISTRICT23;

CREATE TABLE DISTRICT23

```

(
D_NEXT_O_ID INTEGER NOT NULL,
D_TAX INTEGER NOT NULL,
D_YTD BIGINT NOT NULL,
D_NAME CHAR(10) NOT NULL,
D_STREET_1 CHAR(20) NOT NULL,
D_STREET_2 CHAR(20) NOT NULL,
D_CITY CHAR(20) NOT NULL,
D_STATE CHAR(2) NOT NULL,
D_ZIP CHAR(9) NOT NULL,
D_ID SMALLINT NOT NULL,
D_W_ID INTEGER NOT NULL
)

```

IN DIS_023

INDEX IN DIS_023

ORGANIZE BY KEY SEQUENCE (

D_ID STARTING FROM 1 ENDING AT 10,

D_W_ID STARTING FROM 17601 ENDING AT 18400

)

ALLOW OVERFLOW;

connect reset;

connect to TPCC in share mode;

DROP TABLE DISTRICT24;

CREATE TABLE DISTRICT24

```

(
D_NEXT_O_ID INTEGER NOT NULL,
D_TAX INTEGER NOT NULL,
D_YTD BIGINT NOT NULL,
D_NAME CHAR(10) NOT NULL,
D_STREET_1 CHAR(20) NOT NULL,
D_STREET_2 CHAR(20) NOT NULL,
D_CITY CHAR(20) NOT NULL,
D_STATE CHAR(2) NOT NULL,
D_ZIP CHAR(9) NOT NULL,
D_ID SMALLINT NOT NULL,
D_W_ID INTEGER NOT NULL
)

```

IN DIS_024

INDEX IN DIS_024

ORGANIZE BY KEY SEQUENCE (

D_ID STARTING FROM 1 ENDING AT 10,

D_W_ID STARTING FROM 18401 ENDING AT 19200

)

ALLOW OVERFLOW;

connect reset;

connect to TPCC in share mode;

DROP TABLE DISTRICT25;

CREATE TABLE DISTRICT25

```

(
D_NEXT_O_ID INTEGER NOT NULL,
D_TAX INTEGER NOT NULL,
D_YTD BIGINT NOT NULL,

```

```

D_NAME CHAR(10) NOT NULL,
D_STREET_1 CHAR(20) NOT NULL,
D_STREET_2 CHAR(20) NOT NULL,
D_CITY CHAR(20) NOT NULL,
D_STATE CHAR(2) NOT NULL,
D_ZIP CHAR(9) NOT NULL,
D_ID SMALLINT NOT NULL,
D_W_ID INTEGER NOT NULL
)

```

IN DIS_025

INDEX IN DIS_025

ORGANIZE BY KEY SEQUENCE (

D_ID STARTING FROM 1 ENDING AT 10,

D_W_ID STARTING FROM 19201 ENDING AT 20000

)

ALLOW OVERFLOW;

connect reset;

crtb_orders.ddl

connect to TPCC in share mode;

DROP TABLE ORDERS1;

CREATE TABLE ORDERS1

```

(
O_C_ID INTEGER NOT NULL,
O_ENTRY_D BIGINT NOT NULL,
O_CARRIER_ID SMALLINT NOT NULL,
O_OL_CNT SMALLINT NOT NULL,
O_ALL_LOCAL SMALLINT NOT NULL,
O_ID INTEGER NOT NULL,
O_W_ID INTEGER NOT NULL,
O_D_ID SMALLINT NOT NULL
)

```

IN ORD_001

INDEX IN ORDI_001

ORGANIZE BY KEY SEQUENCE (

O_ID STARTING FROM 0 ENDING AT 3675,

O_W_ID STARTING FROM 1 ENDING AT 800,

O_D_ID STARTING FROM 1 ENDING AT 10

)

ALLOW OVERFLOW;

connect reset;

connect to TPCC in share mode;

DROP TABLE ORDERS2;

CREATE TABLE ORDERS2

```

(
O_C_ID INTEGER NOT NULL,
O_ENTRY_D BIGINT NOT NULL,
O_CARRIER_ID SMALLINT NOT NULL,
O_OL_CNT SMALLINT NOT NULL,
O_ALL_LOCAL SMALLINT NOT NULL,
O_ID INTEGER NOT NULL,
O_W_ID INTEGER NOT NULL,
O_D_ID SMALLINT NOT NULL
)

```

IN ORD_002

INDEX IN ORDI_002

ORGANIZE BY KEY SEQUENCE (

O_ID STARTING FROM 0 ENDING AT 3675,

O_W_ID STARTING FROM 801 ENDING AT 1600,

O_D_ID STARTING FROM 1 ENDING AT 10

)

ALLOW OVERFLOW;

connect reset;

connect to TPCC in share mode;

DROP TABLE ORDERS3;

CREATE TABLE ORDERS3

```

(
  O_C_ID    INTEGER NOT NULL,
  O_ENTRY_D BIGINT  NOT NULL,
  O_CARRIER_ID SMALLINT NOT NULL,
  O_OL_CNT  SMALLINT NOT NULL,
  O_ALL_LOCAL SMALLINT NOT NULL,
  O_ID      INTEGER NOT NULL,
  O_W_ID    INTEGER NOT NULL,
  O_D_ID    SMALLINT NOT NULL
)
IN ORD_003
INDEX IN ORDI_003
ORGANIZE BY KEY SEQUENCE (
  O_ID STARTING FROM 0 ENDING AT 3675,
  O_W_ID STARTING FROM 1601 ENDING AT 2400,
  O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS4;
CREATE TABLE ORDERS4

```

```

(
  O_C_ID    INTEGER NOT NULL,
  O_ENTRY_D BIGINT  NOT NULL,
  O_CARRIER_ID SMALLINT NOT NULL,
  O_OL_CNT  SMALLINT NOT NULL,
  O_ALL_LOCAL SMALLINT NOT NULL,
  O_ID      INTEGER NOT NULL,
  O_W_ID    INTEGER NOT NULL,
  O_D_ID    SMALLINT NOT NULL
)
IN ORD_004
INDEX IN ORDI_004
ORGANIZE BY KEY SEQUENCE (
  O_ID STARTING FROM 0 ENDING AT 3675,
  O_W_ID STARTING FROM 2401 ENDING AT 3200,
  O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS5;
CREATE TABLE ORDERS5

```

```

(
  O_C_ID    INTEGER NOT NULL,
  O_ENTRY_D BIGINT  NOT NULL,
  O_CARRIER_ID SMALLINT NOT NULL,
  O_OL_CNT  SMALLINT NOT NULL,
  O_ALL_LOCAL SMALLINT NOT NULL,
  O_ID      INTEGER NOT NULL,
  O_W_ID    INTEGER NOT NULL,
  O_D_ID    SMALLINT NOT NULL
)
IN ORD_005
INDEX IN ORDI_005
ORGANIZE BY KEY SEQUENCE (
  O_ID STARTING FROM 0 ENDING AT 3675,
  O_W_ID STARTING FROM 3201 ENDING AT 4000,
  O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS6;
CREATE TABLE ORDERS6

```

```

(
  O_C_ID    INTEGER NOT NULL,

```

```

  O_ENTRY_D BIGINT  NOT NULL,
  O_CARRIER_ID SMALLINT NOT NULL,
  O_OL_CNT  SMALLINT NOT NULL,
  O_ALL_LOCAL SMALLINT NOT NULL,
  O_ID      INTEGER NOT NULL,
  O_W_ID    INTEGER NOT NULL,
  O_D_ID    SMALLINT NOT NULL
)
IN ORD_006
INDEX IN ORDI_006
ORGANIZE BY KEY SEQUENCE (
  O_ID STARTING FROM 0 ENDING AT 3675,
  O_W_ID STARTING FROM 4001 ENDING AT 4800,
  O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS7;
CREATE TABLE ORDERS7

```

```

(
  O_C_ID    INTEGER NOT NULL,
  O_ENTRY_D BIGINT  NOT NULL,
  O_CARRIER_ID SMALLINT NOT NULL,
  O_OL_CNT  SMALLINT NOT NULL,
  O_ALL_LOCAL SMALLINT NOT NULL,
  O_ID      INTEGER NOT NULL,
  O_W_ID    INTEGER NOT NULL,
  O_D_ID    SMALLINT NOT NULL
)
IN ORD_007
INDEX IN ORDI_007
ORGANIZE BY KEY SEQUENCE (
  O_ID STARTING FROM 0 ENDING AT 3675,
  O_W_ID STARTING FROM 4801 ENDING AT 5600,
  O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS8;
CREATE TABLE ORDERS8

```

```

(
  O_C_ID    INTEGER NOT NULL,
  O_ENTRY_D BIGINT  NOT NULL,
  O_CARRIER_ID SMALLINT NOT NULL,
  O_OL_CNT  SMALLINT NOT NULL,
  O_ALL_LOCAL SMALLINT NOT NULL,
  O_ID      INTEGER NOT NULL,
  O_W_ID    INTEGER NOT NULL,
  O_D_ID    SMALLINT NOT NULL
)
IN ORD_008
INDEX IN ORDI_008
ORGANIZE BY KEY SEQUENCE (
  O_ID STARTING FROM 0 ENDING AT 3675,
  O_W_ID STARTING FROM 5601 ENDING AT 6400,
  O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS9;
CREATE TABLE ORDERS9

```

```

(
  O_C_ID    INTEGER NOT NULL,
  O_ENTRY_D BIGINT  NOT NULL,
  O_CARRIER_ID SMALLINT NOT NULL,

```

```

O_OL_CNT SMALLINT NOT NULL,
O_ALL_LOCAL SMALLINT NOT NULL,
O_ID INTEGER NOT NULL,
O_W_ID INTEGER NOT NULL,
O_D_ID SMALLINT NOT NULL
)
IN ORD_009
INDEX IN ORDI_009
ORGANIZE BY KEY SEQUENCE (
O_ID STARTING FROM 0 ENDING AT 3675,
O_W_ID STARTING FROM 6401 ENDING AT 7200,
O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS10;
CREATE TABLE ORDERS10

```

```

(
O_C_ID INTEGER NOT NULL,
O_ENTRY_D BIGINT NOT NULL,
O_CARRIER_ID SMALLINT NOT NULL,
O_OL_CNT SMALLINT NOT NULL,
O_ALL_LOCAL SMALLINT NOT NULL,
O_ID INTEGER NOT NULL,
O_W_ID INTEGER NOT NULL,
O_D_ID SMALLINT NOT NULL
)
IN ORD_010
INDEX IN ORDI_010
ORGANIZE BY KEY SEQUENCE (
O_ID STARTING FROM 0 ENDING AT 3675,
O_W_ID STARTING FROM 7201 ENDING AT 8000,
O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS11;
CREATE TABLE ORDERS11

```

```

(
O_C_ID INTEGER NOT NULL,
O_ENTRY_D BIGINT NOT NULL,
O_CARRIER_ID SMALLINT NOT NULL,
O_OL_CNT SMALLINT NOT NULL,
O_ALL_LOCAL SMALLINT NOT NULL,
O_ID INTEGER NOT NULL,
O_W_ID INTEGER NOT NULL,
O_D_ID SMALLINT NOT NULL
)
IN ORD_011
INDEX IN ORDI_011
ORGANIZE BY KEY SEQUENCE (
O_ID STARTING FROM 0 ENDING AT 3675,
O_W_ID STARTING FROM 8001 ENDING AT 8800,
O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS12;
CREATE TABLE ORDERS12

```

```

(
O_C_ID INTEGER NOT NULL,
O_ENTRY_D BIGINT NOT NULL,
O_CARRIER_ID SMALLINT NOT NULL,
O_OL_CNT SMALLINT NOT NULL,
O_ALL_LOCAL SMALLINT NOT NULL,

```

```

O_ID INTEGER NOT NULL,
O_W_ID INTEGER NOT NULL,
O_D_ID SMALLINT NOT NULL
)
IN ORD_012
INDEX IN ORDI_012
ORGANIZE BY KEY SEQUENCE (
O_ID STARTING FROM 0 ENDING AT 3675,
O_W_ID STARTING FROM 8801 ENDING AT 9600,
O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS13;
CREATE TABLE ORDERS13

```

```

(
O_C_ID INTEGER NOT NULL,
O_ENTRY_D BIGINT NOT NULL,
O_CARRIER_ID SMALLINT NOT NULL,
O_OL_CNT SMALLINT NOT NULL,
O_ALL_LOCAL SMALLINT NOT NULL,
O_ID INTEGER NOT NULL,
O_W_ID INTEGER NOT NULL,
O_D_ID SMALLINT NOT NULL
)
IN ORD_013
INDEX IN ORDI_013
ORGANIZE BY KEY SEQUENCE (
O_ID STARTING FROM 0 ENDING AT 3675,
O_W_ID STARTING FROM 9601 ENDING AT 10400,
O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS14;
CREATE TABLE ORDERS14

```

```

(
O_C_ID INTEGER NOT NULL,
O_ENTRY_D BIGINT NOT NULL,
O_CARRIER_ID SMALLINT NOT NULL,
O_OL_CNT SMALLINT NOT NULL,
O_ALL_LOCAL SMALLINT NOT NULL,
O_ID INTEGER NOT NULL,
O_W_ID INTEGER NOT NULL,
O_D_ID SMALLINT NOT NULL
)
IN ORD_014
INDEX IN ORDI_014
ORGANIZE BY KEY SEQUENCE (
O_ID STARTING FROM 0 ENDING AT 3675,
O_W_ID STARTING FROM 10401 ENDING AT 11200,
O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS15;
CREATE TABLE ORDERS15

```

```

(
O_C_ID INTEGER NOT NULL,
O_ENTRY_D BIGINT NOT NULL,
O_CARRIER_ID SMALLINT NOT NULL,
O_OL_CNT SMALLINT NOT NULL,
O_ALL_LOCAL SMALLINT NOT NULL,
O_ID INTEGER NOT NULL,
O_W_ID INTEGER NOT NULL,

```

```

O_D_ID SMALLINT NOT NULL
)
IN ORD_015
INDEX IN ORDI_015
ORGANIZE BY KEY SEQUENCE (
O_ID STARTING FROM 0 ENDING AT 3675,
O_W_ID STARTING FROM 11201 ENDING AT 12000,
O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS16;
CREATE TABLE ORDERS16

```

```

(
O_C_ID INTEGER NOT NULL,
O_ENTRY_D BIGINT NOT NULL,
O_CARRIER_ID SMALLINT NOT NULL,
O_OL_CNT SMALLINT NOT NULL,
O_ALL_LOCAL SMALLINT NOT NULL,
O_ID INTEGER NOT NULL,
O_W_ID INTEGER NOT NULL,
O_D_ID SMALLINT NOT NULL
)
IN ORD_016
INDEX IN ORDI_016
ORGANIZE BY KEY SEQUENCE (
O_ID STARTING FROM 0 ENDING AT 3675,
O_W_ID STARTING FROM 12001 ENDING AT 12800,
O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS17;
CREATE TABLE ORDERS17

```

```

(
O_C_ID INTEGER NOT NULL,
O_ENTRY_D BIGINT NOT NULL,
O_CARRIER_ID SMALLINT NOT NULL,
O_OL_CNT SMALLINT NOT NULL,
O_ALL_LOCAL SMALLINT NOT NULL,
O_ID INTEGER NOT NULL,
O_W_ID INTEGER NOT NULL,
O_D_ID SMALLINT NOT NULL
)
IN ORD_017
INDEX IN ORDI_017
ORGANIZE BY KEY SEQUENCE (
O_ID STARTING FROM 0 ENDING AT 3675,
O_W_ID STARTING FROM 12801 ENDING AT 13600,
O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS18;
CREATE TABLE ORDERS18

```

```

(
O_C_ID INTEGER NOT NULL,
O_ENTRY_D BIGINT NOT NULL,
O_CARRIER_ID SMALLINT NOT NULL,
O_OL_CNT SMALLINT NOT NULL,
O_ALL_LOCAL SMALLINT NOT NULL,
O_ID INTEGER NOT NULL,
O_W_ID INTEGER NOT NULL,
O_D_ID SMALLINT NOT NULL
)

```

```

IN ORD_018
INDEX IN ORDI_018
ORGANIZE BY KEY SEQUENCE (
O_ID STARTING FROM 0 ENDING AT 3675,
O_W_ID STARTING FROM 13601 ENDING AT 14400,
O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS19;
CREATE TABLE ORDERS19

```

```

(
O_C_ID INTEGER NOT NULL,
O_ENTRY_D BIGINT NOT NULL,
O_CARRIER_ID SMALLINT NOT NULL,
O_OL_CNT SMALLINT NOT NULL,
O_ALL_LOCAL SMALLINT NOT NULL,
O_ID INTEGER NOT NULL,
O_W_ID INTEGER NOT NULL,
O_D_ID SMALLINT NOT NULL
)
IN ORD_019
INDEX IN ORDI_019
ORGANIZE BY KEY SEQUENCE (
O_ID STARTING FROM 0 ENDING AT 3675,
O_W_ID STARTING FROM 14401 ENDING AT 15200,
O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS20;
CREATE TABLE ORDERS20

```

```

(
O_C_ID INTEGER NOT NULL,
O_ENTRY_D BIGINT NOT NULL,
O_CARRIER_ID SMALLINT NOT NULL,
O_OL_CNT SMALLINT NOT NULL,
O_ALL_LOCAL SMALLINT NOT NULL,
O_ID INTEGER NOT NULL,
O_W_ID INTEGER NOT NULL,
O_D_ID SMALLINT NOT NULL
)
IN ORD_020
INDEX IN ORDI_020
ORGANIZE BY KEY SEQUENCE (
O_ID STARTING FROM 0 ENDING AT 3675,
O_W_ID STARTING FROM 15201 ENDING AT 16000,
O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS21;
CREATE TABLE ORDERS21

```

```

(
O_C_ID INTEGER NOT NULL,
O_ENTRY_D BIGINT NOT NULL,
O_CARRIER_ID SMALLINT NOT NULL,
O_OL_CNT SMALLINT NOT NULL,
O_ALL_LOCAL SMALLINT NOT NULL,
O_ID INTEGER NOT NULL,
O_W_ID INTEGER NOT NULL,
O_D_ID SMALLINT NOT NULL
)
IN ORD_021
INDEX IN ORDI_021

```

```

ORGANIZE BY KEY SEQUENCE (
  O_ID STARTING FROM 0 ENDING AT 3675,
  O_W_ID STARTING FROM 16001 ENDING AT 16800,
  O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS22;
CREATE TABLE ORDERS22
(
  O_C_ID INTEGER NOT NULL,
  O_ENTRY_D BIGINT NOT NULL,
  O_CARRIER_ID SMALLINT NOT NULL,
  O_OL_CNT SMALLINT NOT NULL,
  O_ALL_LOCAL SMALLINT NOT NULL,
  O_ID INTEGER NOT NULL,
  O_W_ID INTEGER NOT NULL,
  O_D_ID SMALLINT NOT NULL
)
IN ORD_022
INDEX IN ORDI_022
ORGANIZE BY KEY SEQUENCE (
  O_ID STARTING FROM 0 ENDING AT 3675,
  O_W_ID STARTING FROM 16801 ENDING AT 17600,
  O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS23;
CREATE TABLE ORDERS23
(
  O_C_ID INTEGER NOT NULL,
  O_ENTRY_D BIGINT NOT NULL,
  O_CARRIER_ID SMALLINT NOT NULL,
  O_OL_CNT SMALLINT NOT NULL,
  O_ALL_LOCAL SMALLINT NOT NULL,
  O_ID INTEGER NOT NULL,
  O_W_ID INTEGER NOT NULL,
  O_D_ID SMALLINT NOT NULL
)
IN ORD_023
INDEX IN ORDI_023
ORGANIZE BY KEY SEQUENCE (
  O_ID STARTING FROM 0 ENDING AT 3675,
  O_W_ID STARTING FROM 17601 ENDING AT 18400,
  O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS24;
CREATE TABLE ORDERS24
(
  O_C_ID INTEGER NOT NULL,
  O_ENTRY_D BIGINT NOT NULL,
  O_CARRIER_ID SMALLINT NOT NULL,
  O_OL_CNT SMALLINT NOT NULL,
  O_ALL_LOCAL SMALLINT NOT NULL,
  O_ID INTEGER NOT NULL,
  O_W_ID INTEGER NOT NULL,
  O_D_ID SMALLINT NOT NULL
)
IN ORD_024
INDEX IN ORDI_024
ORGANIZE BY KEY SEQUENCE (
  O_ID STARTING FROM 0 ENDING AT 3675,

```

```

  O_W_ID STARTING FROM 18401 ENDING AT 19200,
  O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDERS25;
CREATE TABLE ORDERS25
(
  O_C_ID INTEGER NOT NULL,
  O_ENTRY_D BIGINT NOT NULL,
  O_CARRIER_ID SMALLINT NOT NULL,
  O_OL_CNT SMALLINT NOT NULL,
  O_ALL_LOCAL SMALLINT NOT NULL,
  O_ID INTEGER NOT NULL,
  O_W_ID INTEGER NOT NULL,
  O_D_ID SMALLINT NOT NULL
)
IN ORD_025
INDEX IN ORDI_025
ORGANIZE BY KEY SEQUENCE (
  O_ID STARTING FROM 0 ENDING AT 3675,
  O_W_ID STARTING FROM 19201 ENDING AT 20000,
  O_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;
connect reset;

```

crtb_order_line.ddl

```

connect to TPCC in share mode;
DROP TABLE ORDER_LINE1;
CREATE TABLE ORDER_LINE1
(
  OL_DELIVERY_D BIGINT NOT NULL,
  OL_AMOUNT INTEGER NOT NULL,
  OL_I_ID INTEGER NOT NULL,
  OL_SUPPLY_W_ID INTEGER NOT NULL,
  OL_QUANTITY SMALLINT NOT NULL,
  OL_DIST_INFO CHAR(24) NOT NULL,
  OL_O_ID INTEGER NOT NULL,
  OL_D_ID SMALLINT NOT NULL,
  OL_W_ID INTEGER NOT NULL,
  OL_NUMBER SMALLINT NOT NULL
)
IN OLN_001
INDEX IN OLN_001
ORGANIZE BY KEY SEQUENCE (
  OL_W_ID STARTING FROM 1 ENDING AT 800,
  OL_D_ID STARTING FROM 1 ENDING AT 10,
  OL_O_ID STARTING FROM 0 ENDING AT 3675,
  OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE2;
CREATE TABLE ORDER_LINE2
(
  OL_DELIVERY_D BIGINT NOT NULL,
  OL_AMOUNT INTEGER NOT NULL,
  OL_I_ID INTEGER NOT NULL,
  OL_SUPPLY_W_ID INTEGER NOT NULL,
  OL_QUANTITY SMALLINT NOT NULL,
  OL_DIST_INFO CHAR(24) NOT NULL,
  OL_O_ID INTEGER NOT NULL,
  OL_D_ID SMALLINT NOT NULL,

```

```

OL_W_ID    INTEGER NOT NULL,
OL_NUMBER  SMALLINT NOT NULL
)
IN OLN_002
INDEX IN OLN_002
ORGANIZE BY KEY SEQUENCE (
  OL_W_ID STARTING FROM 801 ENDING AT 1600,
  OL_D_ID STARTING FROM 1 ENDING AT 10,
  OL_O_ID STARTING FROM 0 ENDING AT 3675,
  OL_NUMBER STARTING FROM 1 ENDING AT 15
)
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE3;
CREATE TABLE ORDER_LINE3
(
  OL_DELIVERY_D BIGINT NOT NULL,
  OL_AMOUNT     INTEGER NOT NULL,
  OL_I_ID       INTEGER NOT NULL,
  OL_SUPPLY_W_ID INTEGER NOT NULL,
  OL_QUANTITY   SMALLINT NOT NULL,
  OL_DIST_INFO  CHAR(24) NOT NULL,
  OL_O_ID       INTEGER NOT NULL,
  OL_D_ID       SMALLINT NOT NULL,
  OL_W_ID       INTEGER NOT NULL,
  OL_NUMBER     SMALLINT NOT NULL
)
IN OLN_003
INDEX IN OLN_003
ORGANIZE BY KEY SEQUENCE (
  OL_W_ID STARTING FROM 1601 ENDING AT 2400,
  OL_D_ID STARTING FROM 1 ENDING AT 10,
  OL_O_ID STARTING FROM 0 ENDING AT 3675,
  OL_NUMBER STARTING FROM 1 ENDING AT 15
)
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE4;
CREATE TABLE ORDER_LINE4
(
  OL_DELIVERY_D BIGINT NOT NULL,
  OL_AMOUNT     INTEGER NOT NULL,
  OL_I_ID       INTEGER NOT NULL,
  OL_SUPPLY_W_ID INTEGER NOT NULL,
  OL_QUANTITY   SMALLINT NOT NULL,
  OL_DIST_INFO  CHAR(24) NOT NULL,
  OL_O_ID       INTEGER NOT NULL,
  OL_D_ID       SMALLINT NOT NULL,
  OL_W_ID       INTEGER NOT NULL,
  OL_NUMBER     SMALLINT NOT NULL
)
IN OLN_004
INDEX IN OLN_004
ORGANIZE BY KEY SEQUENCE (
  OL_W_ID STARTING FROM 2401 ENDING AT 3200,
  OL_D_ID STARTING FROM 1 ENDING AT 10,
  OL_O_ID STARTING FROM 0 ENDING AT 3675,
  OL_NUMBER STARTING FROM 1 ENDING AT 15
)
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE5;
CREATE TABLE ORDER_LINE5
(
  OL_DELIVERY_D BIGINT NOT NULL,

```

```

OL_AMOUNT   INTEGER NOT NULL,
OL_I_ID     INTEGER NOT NULL,
OL_SUPPLY_W_ID INTEGER NOT NULL,
OL_QUANTITY SMALLINT NOT NULL,
OL_DIST_INFO CHAR(24) NOT NULL,
OL_O_ID     INTEGER NOT NULL,
OL_D_ID     SMALLINT NOT NULL,
OL_W_ID     INTEGER NOT NULL,
OL_NUMBER   SMALLINT NOT NULL
)
)
IN OLN_005
INDEX IN OLN_005
ORGANIZE BY KEY SEQUENCE (
  OL_W_ID STARTING FROM 3201 ENDING AT 4000,
  OL_D_ID STARTING FROM 1 ENDING AT 10,
  OL_O_ID STARTING FROM 0 ENDING AT 3675,
  OL_NUMBER STARTING FROM 1 ENDING AT 15
)
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE6;
CREATE TABLE ORDER_LINE6
(
  OL_DELIVERY_D BIGINT NOT NULL,
  OL_AMOUNT     INTEGER NOT NULL,
  OL_I_ID       INTEGER NOT NULL,
  OL_SUPPLY_W_ID INTEGER NOT NULL,
  OL_QUANTITY   SMALLINT NOT NULL,
  OL_DIST_INFO  CHAR(24) NOT NULL,
  OL_O_ID       INTEGER NOT NULL,
  OL_D_ID       SMALLINT NOT NULL,
  OL_W_ID       INTEGER NOT NULL,
  OL_NUMBER     SMALLINT NOT NULL
)
)
IN OLN_006
INDEX IN OLN_006
ORGANIZE BY KEY SEQUENCE (
  OL_W_ID STARTING FROM 4001 ENDING AT 4800,
  OL_D_ID STARTING FROM 1 ENDING AT 10,
  OL_O_ID STARTING FROM 0 ENDING AT 3675,
  OL_NUMBER STARTING FROM 1 ENDING AT 15
)
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE7;
CREATE TABLE ORDER_LINE7
(
  OL_DELIVERY_D BIGINT NOT NULL,
  OL_AMOUNT     INTEGER NOT NULL,
  OL_I_ID       INTEGER NOT NULL,
  OL_SUPPLY_W_ID INTEGER NOT NULL,
  OL_QUANTITY   SMALLINT NOT NULL,
  OL_DIST_INFO  CHAR(24) NOT NULL,
  OL_O_ID       INTEGER NOT NULL,
  OL_D_ID       SMALLINT NOT NULL,
  OL_W_ID       INTEGER NOT NULL,
  OL_NUMBER     SMALLINT NOT NULL
)
)
IN OLN_007
INDEX IN OLN_007
ORGANIZE BY KEY SEQUENCE (
  OL_W_ID STARTING FROM 4801 ENDING AT 5600,
  OL_D_ID STARTING FROM 1 ENDING AT 10,
  OL_O_ID STARTING FROM 0 ENDING AT 3675,
  OL_NUMBER STARTING FROM 1 ENDING AT 15
)
)

```



```

ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE8;
CREATE TABLE ORDER_LINE8
(
  OL_DELIVERY_D BIGINT NOT NULL,
  OL_AMOUNT INTEGER NOT NULL,
  OL_I_ID INTEGER NOT NULL,
  OL_SUPPLY_W_ID INTEGER NOT NULL,
  OL_QUANTITY SMALLINT NOT NULL,
  OL_DIST_INFO CHAR(24) NOT NULL,
  OL_O_ID INTEGER NOT NULL,
  OL_D_ID SMALLINT NOT NULL,
  OL_W_ID INTEGER NOT NULL,
  OL_NUMBER SMALLINT NOT NULL
)
IN OLN_008
INDEX IN OLN_008
ORGANIZE BY KEY SEQUENCE (
  OL_W_ID STARTING FROM 5601 ENDING AT 6400,
  OL_D_ID STARTING FROM 1 ENDING AT 10,
  OL_O_ID STARTING FROM 0 ENDING AT 3675,
  OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE9;
CREATE TABLE ORDER_LINE9
(
  OL_DELIVERY_D BIGINT NOT NULL,
  OL_AMOUNT INTEGER NOT NULL,
  OL_I_ID INTEGER NOT NULL,
  OL_SUPPLY_W_ID INTEGER NOT NULL,
  OL_QUANTITY SMALLINT NOT NULL,
  OL_DIST_INFO CHAR(24) NOT NULL,
  OL_O_ID INTEGER NOT NULL,
  OL_D_ID SMALLINT NOT NULL,
  OL_W_ID INTEGER NOT NULL,
  OL_NUMBER SMALLINT NOT NULL
)
IN OLN_009
INDEX IN OLN_009
ORGANIZE BY KEY SEQUENCE (
  OL_W_ID STARTING FROM 6401 ENDING AT 7200,
  OL_D_ID STARTING FROM 1 ENDING AT 10,
  OL_O_ID STARTING FROM 0 ENDING AT 3675,
  OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE10;
CREATE TABLE ORDER_LINE10
(
  OL_DELIVERY_D BIGINT NOT NULL,
  OL_AMOUNT INTEGER NOT NULL,
  OL_I_ID INTEGER NOT NULL,
  OL_SUPPLY_W_ID INTEGER NOT NULL,
  OL_QUANTITY SMALLINT NOT NULL,
  OL_DIST_INFO CHAR(24) NOT NULL,
  OL_O_ID INTEGER NOT NULL,
  OL_D_ID SMALLINT NOT NULL,
  OL_W_ID INTEGER NOT NULL,
  OL_NUMBER SMALLINT NOT NULL
)
IN OLN_010

```

```

INDEX IN OLN_010
ORGANIZE BY KEY SEQUENCE (
  OL_W_ID STARTING FROM 7201 ENDING AT 8000,
  OL_D_ID STARTING FROM 1 ENDING AT 10,
  OL_O_ID STARTING FROM 0 ENDING AT 3675,
  OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE11;
CREATE TABLE ORDER_LINE11
(
  OL_DELIVERY_D BIGINT NOT NULL,
  OL_AMOUNT INTEGER NOT NULL,
  OL_I_ID INTEGER NOT NULL,
  OL_SUPPLY_W_ID INTEGER NOT NULL,
  OL_QUANTITY SMALLINT NOT NULL,
  OL_DIST_INFO CHAR(24) NOT NULL,
  OL_O_ID INTEGER NOT NULL,
  OL_D_ID SMALLINT NOT NULL,
  OL_W_ID INTEGER NOT NULL,
  OL_NUMBER SMALLINT NOT NULL
)
IN OLN_011
INDEX IN OLN_011
ORGANIZE BY KEY SEQUENCE (
  OL_W_ID STARTING FROM 8001 ENDING AT 8800,
  OL_D_ID STARTING FROM 1 ENDING AT 10,
  OL_O_ID STARTING FROM 0 ENDING AT 3675,
  OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE12;
CREATE TABLE ORDER_LINE12
(
  OL_DELIVERY_D BIGINT NOT NULL,
  OL_AMOUNT INTEGER NOT NULL,
  OL_I_ID INTEGER NOT NULL,
  OL_SUPPLY_W_ID INTEGER NOT NULL,
  OL_QUANTITY SMALLINT NOT NULL,
  OL_DIST_INFO CHAR(24) NOT NULL,
  OL_O_ID INTEGER NOT NULL,
  OL_D_ID SMALLINT NOT NULL,
  OL_W_ID INTEGER NOT NULL,
  OL_NUMBER SMALLINT NOT NULL
)
IN OLN_012
INDEX IN OLN_012
ORGANIZE BY KEY SEQUENCE (
  OL_W_ID STARTING FROM 8801 ENDING AT 9600,
  OL_D_ID STARTING FROM 1 ENDING AT 10,
  OL_O_ID STARTING FROM 0 ENDING AT 3675,
  OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE13;
CREATE TABLE ORDER_LINE13
(
  OL_DELIVERY_D BIGINT NOT NULL,
  OL_AMOUNT INTEGER NOT NULL,
  OL_I_ID INTEGER NOT NULL,
  OL_SUPPLY_W_ID INTEGER NOT NULL,
  OL_QUANTITY SMALLINT NOT NULL,

```

```

OL_DIST_INFO CHAR(24) NOT NULL,
OL_O_ID INTEGER NOT NULL,
OL_D_ID SMALLINT NOT NULL,
OL_W_ID INTEGER NOT NULL,
OL_NUMBER SMALLINT NOT NULL
)
IN OLN_013
INDEX IN OLN_013
ORGANIZE BY KEY SEQUENCE (
OL_W_ID STARTING FROM 9601 ENDING AT 10400,
OL_D_ID STARTING FROM 1 ENDING AT 10,
OL_O_ID STARTING FROM 0 ENDING AT 3675,
OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE14;
CREATE TABLE ORDER_LINE14
(
OL_DELIVERY_D BIGINT NOT NULL,
OL_AMOUNT INTEGER NOT NULL,
OL_I_ID INTEGER NOT NULL,
OL_SUPPLY_W_ID INTEGER NOT NULL,
OL_QUANTITY SMALLINT NOT NULL,
OL_DIST_INFO CHAR(24) NOT NULL,
OL_O_ID INTEGER NOT NULL,
OL_D_ID SMALLINT NOT NULL,
OL_W_ID INTEGER NOT NULL,
OL_NUMBER SMALLINT NOT NULL
)
IN OLN_014
INDEX IN OLN_014
ORGANIZE BY KEY SEQUENCE (
OL_W_ID STARTING FROM 10401 ENDING AT 11200,
OL_D_ID STARTING FROM 1 ENDING AT 10,
OL_O_ID STARTING FROM 0 ENDING AT 3675,
OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE15;
CREATE TABLE ORDER_LINE15
(
OL_DELIVERY_D BIGINT NOT NULL,
OL_AMOUNT INTEGER NOT NULL,
OL_I_ID INTEGER NOT NULL,
OL_SUPPLY_W_ID INTEGER NOT NULL,
OL_QUANTITY SMALLINT NOT NULL,
OL_DIST_INFO CHAR(24) NOT NULL,
OL_O_ID INTEGER NOT NULL,
OL_D_ID SMALLINT NOT NULL,
OL_W_ID INTEGER NOT NULL,
OL_NUMBER SMALLINT NOT NULL
)
IN OLN_015
INDEX IN OLN_015
ORGANIZE BY KEY SEQUENCE (
OL_W_ID STARTING FROM 11201 ENDING AT 12000,
OL_D_ID STARTING FROM 1 ENDING AT 10,
OL_O_ID STARTING FROM 0 ENDING AT 3675,
OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE16;

```

```

CREATE TABLE ORDER_LINE16
(
OL_DELIVERY_D BIGINT NOT NULL,
OL_AMOUNT INTEGER NOT NULL,
OL_I_ID INTEGER NOT NULL,
OL_SUPPLY_W_ID INTEGER NOT NULL,
OL_QUANTITY SMALLINT NOT NULL,
OL_DIST_INFO CHAR(24) NOT NULL,
OL_O_ID INTEGER NOT NULL,
OL_D_ID SMALLINT NOT NULL,
OL_W_ID INTEGER NOT NULL,
OL_NUMBER SMALLINT NOT NULL
)
IN OLN_016
INDEX IN OLN_016
ORGANIZE BY KEY SEQUENCE (
OL_W_ID STARTING FROM 12001 ENDING AT 12800,
OL_D_ID STARTING FROM 1 ENDING AT 10,
OL_O_ID STARTING FROM 0 ENDING AT 3675,
OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE17;
CREATE TABLE ORDER_LINE17
(
OL_DELIVERY_D BIGINT NOT NULL,
OL_AMOUNT INTEGER NOT NULL,
OL_I_ID INTEGER NOT NULL,
OL_SUPPLY_W_ID INTEGER NOT NULL,
OL_QUANTITY SMALLINT NOT NULL,
OL_DIST_INFO CHAR(24) NOT NULL,
OL_O_ID INTEGER NOT NULL,
OL_D_ID SMALLINT NOT NULL,
OL_W_ID INTEGER NOT NULL,
OL_NUMBER SMALLINT NOT NULL
)
IN OLN_017
INDEX IN OLN_017
ORGANIZE BY KEY SEQUENCE (
OL_W_ID STARTING FROM 12801 ENDING AT 13600,
OL_D_ID STARTING FROM 1 ENDING AT 10,
OL_O_ID STARTING FROM 0 ENDING AT 3675,
OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE18;
CREATE TABLE ORDER_LINE18
(
OL_DELIVERY_D BIGINT NOT NULL,
OL_AMOUNT INTEGER NOT NULL,
OL_I_ID INTEGER NOT NULL,
OL_SUPPLY_W_ID INTEGER NOT NULL,
OL_QUANTITY SMALLINT NOT NULL,
OL_DIST_INFO CHAR(24) NOT NULL,
OL_O_ID INTEGER NOT NULL,
OL_D_ID SMALLINT NOT NULL,
OL_W_ID INTEGER NOT NULL,
OL_NUMBER SMALLINT NOT NULL
)
IN OLN_018
INDEX IN OLN_018
ORGANIZE BY KEY SEQUENCE (
OL_W_ID STARTING FROM 13601 ENDING AT 14400,
OL_D_ID STARTING FROM 1 ENDING AT 10,

```

```

        OL_O_ID STARTING FROM 0 ENDING AT 3675,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
    )
    ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE19;
CREATE TABLE ORDER_LINE19
(
    OL_DELIVERY_D BIGINT NOT NULL,
    OL_AMOUNT INTEGER NOT NULL,
    OL_I_ID INTEGER NOT NULL,
    OL_SUPPLY_W_ID INTEGER NOT NULL,
    OL_QUANTITY SMALLINT NOT NULL,
    OL_DIST_INFO CHAR(24) NOT NULL,
    OL_O_ID INTEGER NOT NULL,
    OL_D_ID SMALLINT NOT NULL,
    OL_W_ID INTEGER NOT NULL,
    OL_NUMBER SMALLINT NOT NULL
)
IN OLN_019
INDEX IN OLN_019
ORGANIZE BY KEY SEQUENCE (
    OL_W_ID STARTING FROM 14401 ENDING AT 15200,
    OL_D_ID STARTING FROM 1 ENDING AT 10,
    OL_O_ID STARTING FROM 0 ENDING AT 3675,
    OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE20;
CREATE TABLE ORDER_LINE20
(
    OL_DELIVERY_D BIGINT NOT NULL,
    OL_AMOUNT INTEGER NOT NULL,
    OL_I_ID INTEGER NOT NULL,
    OL_SUPPLY_W_ID INTEGER NOT NULL,
    OL_QUANTITY SMALLINT NOT NULL,
    OL_DIST_INFO CHAR(24) NOT NULL,
    OL_O_ID INTEGER NOT NULL,
    OL_D_ID SMALLINT NOT NULL,
    OL_W_ID INTEGER NOT NULL,
    OL_NUMBER SMALLINT NOT NULL
)
IN OLN_020
INDEX IN OLN_020
ORGANIZE BY KEY SEQUENCE (
    OL_W_ID STARTING FROM 15201 ENDING AT 16000,
    OL_D_ID STARTING FROM 1 ENDING AT 10,
    OL_O_ID STARTING FROM 0 ENDING AT 3675,
    OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE21;
CREATE TABLE ORDER_LINE21
(
    OL_DELIVERY_D BIGINT NOT NULL,
    OL_AMOUNT INTEGER NOT NULL,
    OL_I_ID INTEGER NOT NULL,
    OL_SUPPLY_W_ID INTEGER NOT NULL,
    OL_QUANTITY SMALLINT NOT NULL,
    OL_DIST_INFO CHAR(24) NOT NULL,
    OL_O_ID INTEGER NOT NULL,
    OL_D_ID SMALLINT NOT NULL,
    OL_W_ID INTEGER NOT NULL,

```

```

        OL_NUMBER SMALLINT NOT NULL
    )
    IN OLN_021
    INDEX IN OLN_021
    ORGANIZE BY KEY SEQUENCE (
        OL_W_ID STARTING FROM 16001 ENDING AT 16800,
        OL_D_ID STARTING FROM 1 ENDING AT 10,
        OL_O_ID STARTING FROM 0 ENDING AT 3675,
        OL_NUMBER STARTING FROM 1 ENDING AT 15
    )
    ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE22;
CREATE TABLE ORDER_LINE22
(
    OL_DELIVERY_D BIGINT NOT NULL,
    OL_AMOUNT INTEGER NOT NULL,
    OL_I_ID INTEGER NOT NULL,
    OL_SUPPLY_W_ID INTEGER NOT NULL,
    OL_QUANTITY SMALLINT NOT NULL,
    OL_DIST_INFO CHAR(24) NOT NULL,
    OL_O_ID INTEGER NOT NULL,
    OL_D_ID SMALLINT NOT NULL,
    OL_W_ID INTEGER NOT NULL,
    OL_NUMBER SMALLINT NOT NULL
)
IN OLN_022
INDEX IN OLN_022
ORGANIZE BY KEY SEQUENCE (
    OL_W_ID STARTING FROM 16801 ENDING AT 17600,
    OL_D_ID STARTING FROM 1 ENDING AT 10,
    OL_O_ID STARTING FROM 0 ENDING AT 3675,
    OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE23;
CREATE TABLE ORDER_LINE23
(
    OL_DELIVERY_D BIGINT NOT NULL,
    OL_AMOUNT INTEGER NOT NULL,
    OL_I_ID INTEGER NOT NULL,
    OL_SUPPLY_W_ID INTEGER NOT NULL,
    OL_QUANTITY SMALLINT NOT NULL,
    OL_DIST_INFO CHAR(24) NOT NULL,
    OL_O_ID INTEGER NOT NULL,
    OL_D_ID SMALLINT NOT NULL,
    OL_W_ID INTEGER NOT NULL,
    OL_NUMBER SMALLINT NOT NULL
)
IN OLN_023
INDEX IN OLN_023
ORGANIZE BY KEY SEQUENCE (
    OL_W_ID STARTING FROM 17601 ENDING AT 18400,
    OL_D_ID STARTING FROM 1 ENDING AT 10,
    OL_O_ID STARTING FROM 0 ENDING AT 3675,
    OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE24;
CREATE TABLE ORDER_LINE24
(
    OL_DELIVERY_D BIGINT NOT NULL,
    OL_AMOUNT INTEGER NOT NULL,

```

```

OL_I_ID    INTEGER NOT NULL,
OL_SUPPLY_W_ID INTEGER NOT NULL,
OL_QUANTITY SMALLINT NOT NULL,
OL_DIST_INFO CHAR(24) NOT NULL,
OL_O_ID    INTEGER NOT NULL,
OL_D_ID    SMALLINT NOT NULL,
OL_W_ID    INTEGER NOT NULL,
OL_NUMBER  SMALLINT NOT NULL
)
IN OLN_024
INDEX IN OLN_024
ORGANIZE BY KEY SEQUENCE (
  OL_W_ID STARTING FROM 18401 ENDING AT 19200,
  OL_D_ID STARTING FROM 1 ENDING AT 10,
  OL_O_ID STARTING FROM 0 ENDING AT 3675,
  OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE ORDER_LINE25;
CREATE TABLE ORDER_LINE25
(
  OL_DELIVERY_D BIGINT NOT NULL,
  OL_AMOUNT     INTEGER NOT NULL,
  OL_I_ID       INTEGER NOT NULL,
  OL_SUPPLY_W_ID INTEGER NOT NULL,
  OL_QUANTITY   SMALLINT NOT NULL,
  OL_DIST_INFO  CHAR(24) NOT NULL,
  OL_O_ID       INTEGER NOT NULL,
  OL_D_ID       SMALLINT NOT NULL,
  OL_W_ID       INTEGER NOT NULL,
  OL_NUMBER     SMALLINT NOT NULL
)
IN OLN_025
INDEX IN OLN_025
ORGANIZE BY KEY SEQUENCE (
  OL_W_ID STARTING FROM 19201 ENDING AT 20000,
  OL_D_ID STARTING FROM 1 ENDING AT 10,
  OL_O_ID STARTING FROM 0 ENDING AT 3675,
  OL_NUMBER STARTING FROM 1 ENDING AT 15
)
ALLOW OVERFLOW;
connect reset;

```

crtb_new_ordera.ddl

```

connect to TPCC in share mode;
DROP TABLE NEW_ORDERA1;
CREATE TABLE NEW_ORDERA1
(
  NO_O_ID    INTEGER NOT NULL,
  NO_D_ID    SMALLINT NOT NULL,
  NO_W_ID    INTEGER NOT NULL
)
IN NEWA_001
INDEX IN NEWA_001
ORGANIZE BY KEY SEQUENCE (
  NO_W_ID STARTING FROM 1 ENDING AT 800,
  NO_D_ID STARTING FROM 1 ENDING AT 10,
  NO_O_ID STARTING FROM 1900 ENDING AT 3675
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA2;
CREATE TABLE NEW_ORDERA2

```

```

(
  NO_O_ID    INTEGER NOT NULL,
  NO_D_ID    SMALLINT NOT NULL,
  NO_W_ID    INTEGER NOT NULL
)
IN NEWA_002
INDEX IN NEWA_002
ORGANIZE BY KEY SEQUENCE (
  NO_W_ID STARTING FROM 801 ENDING AT 1600,
  NO_D_ID STARTING FROM 1 ENDING AT 10,
  NO_O_ID STARTING FROM 1900 ENDING AT 3675
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA3;
CREATE TABLE NEW_ORDERA3
(
  NO_O_ID    INTEGER NOT NULL,
  NO_D_ID    SMALLINT NOT NULL,
  NO_W_ID    INTEGER NOT NULL
)
IN NEWA_003
INDEX IN NEWA_003
ORGANIZE BY KEY SEQUENCE (
  NO_W_ID STARTING FROM 1601 ENDING AT 2400,
  NO_D_ID STARTING FROM 1 ENDING AT 10,
  NO_O_ID STARTING FROM 1900 ENDING AT 3675
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA4;
CREATE TABLE NEW_ORDERA4
(
  NO_O_ID    INTEGER NOT NULL,
  NO_D_ID    SMALLINT NOT NULL,
  NO_W_ID    INTEGER NOT NULL
)
IN NEWA_004
INDEX IN NEWA_004
ORGANIZE BY KEY SEQUENCE (
  NO_W_ID STARTING FROM 2401 ENDING AT 3200,
  NO_D_ID STARTING FROM 1 ENDING AT 10,
  NO_O_ID STARTING FROM 1900 ENDING AT 3675
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA5;
CREATE TABLE NEW_ORDERA5
(
  NO_O_ID    INTEGER NOT NULL,
  NO_D_ID    SMALLINT NOT NULL,
  NO_W_ID    INTEGER NOT NULL
)
IN NEWA_005
INDEX IN NEWA_005
ORGANIZE BY KEY SEQUENCE (
  NO_W_ID STARTING FROM 3201 ENDING AT 4000,
  NO_D_ID STARTING FROM 1 ENDING AT 10,
  NO_O_ID STARTING FROM 1900 ENDING AT 3675
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA6;
CREATE TABLE NEW_ORDERA6

```

```

(
NO_O_ID    INTEGER    NOT NULL,
NO_D_ID    SMALLINT  NOT NULL,
NO_W_ID    INTEGER    NOT NULL
)
)
IN NEWA_006
INDEX IN NEWA_006
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 4001 ENDING AT 4800,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 1900 ENDING AT 3675
)
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA7;
CREATE TABLE NEW_ORDERA7

```

```

(
NO_O_ID    INTEGER    NOT NULL,
NO_D_ID    SMALLINT  NOT NULL,
NO_W_ID    INTEGER    NOT NULL
)
)
IN NEWA_007
INDEX IN NEWA_007
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 4801 ENDING AT 5600,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 1900 ENDING AT 3675
)
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA8;
CREATE TABLE NEW_ORDERA8

```

```

(
NO_O_ID    INTEGER    NOT NULL,
NO_D_ID    SMALLINT  NOT NULL,
NO_W_ID    INTEGER    NOT NULL
)
)
IN NEWA_008
INDEX IN NEWA_008
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 5601 ENDING AT 6400,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 1900 ENDING AT 3675
)
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA9;
CREATE TABLE NEW_ORDERA9

```

```

(
NO_O_ID    INTEGER    NOT NULL,
NO_D_ID    SMALLINT  NOT NULL,
NO_W_ID    INTEGER    NOT NULL
)
)
IN NEWA_009
INDEX IN NEWA_009
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 6401 ENDING AT 7200,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 1900 ENDING AT 3675
)
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA10;
CREATE TABLE NEW_ORDERA10

```

```

(
NO_O_ID    INTEGER    NOT NULL,
NO_D_ID    SMALLINT  NOT NULL,
NO_W_ID    INTEGER    NOT NULL
)
)
IN NEWA_010
INDEX IN NEWA_010
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 7201 ENDING AT 8000,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 1900 ENDING AT 3675
)
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA11;
CREATE TABLE NEW_ORDERA11

```

```

(
NO_O_ID    INTEGER    NOT NULL,
NO_D_ID    SMALLINT  NOT NULL,
NO_W_ID    INTEGER    NOT NULL
)
)
IN NEWA_011
INDEX IN NEWA_011
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 8001 ENDING AT 8800,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 1900 ENDING AT 3675
)
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA12;
CREATE TABLE NEW_ORDERA12

```

```

(
NO_O_ID    INTEGER    NOT NULL,
NO_D_ID    SMALLINT  NOT NULL,
NO_W_ID    INTEGER    NOT NULL
)
)
IN NEWA_012
INDEX IN NEWA_012
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 8801 ENDING AT 9600,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 1900 ENDING AT 3675
)
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA13;
CREATE TABLE NEW_ORDERA13

```

```

(
NO_O_ID    INTEGER    NOT NULL,
NO_D_ID    SMALLINT  NOT NULL,
NO_W_ID    INTEGER    NOT NULL
)
)
IN NEWA_013
INDEX IN NEWA_013
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 9601 ENDING AT 10400,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 1900 ENDING AT 3675
)
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA14;
CREATE TABLE NEW_ORDERA14

```

```

(
NO_O_ID    INTEGER    NOT NULL,
NO_D_ID    SMALLINT   NOT NULL,
NO_W_ID    INTEGER    NOT NULL
)
)
IN NEWA_014
INDEX IN NEWA_014
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 10401 ENDING AT 11200,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 1900 ENDING AT 3675
)
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA15;
CREATE TABLE NEW_ORDERA15

```

```

(
NO_O_ID    INTEGER    NOT NULL,
NO_D_ID    SMALLINT   NOT NULL,
NO_W_ID    INTEGER    NOT NULL
)
)
IN NEWA_015
INDEX IN NEWA_015
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 11201 ENDING AT 12000,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 1900 ENDING AT 3675
)
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA16;
CREATE TABLE NEW_ORDERA16

```

```

(
NO_O_ID    INTEGER    NOT NULL,
NO_D_ID    SMALLINT   NOT NULL,
NO_W_ID    INTEGER    NOT NULL
)
)
IN NEWA_016
INDEX IN NEWA_016
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 12001 ENDING AT 12800,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 1900 ENDING AT 3675
)
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA17;
CREATE TABLE NEW_ORDERA17

```

```

(
NO_O_ID    INTEGER    NOT NULL,
NO_D_ID    SMALLINT   NOT NULL,
NO_W_ID    INTEGER    NOT NULL
)
)
IN NEWA_017
INDEX IN NEWA_017
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 12801 ENDING AT 13600,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 1900 ENDING AT 3675
)
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA18;
CREATE TABLE NEW_ORDERA18

```

```

(
NO_O_ID    INTEGER    NOT NULL,
NO_D_ID    SMALLINT   NOT NULL,
NO_W_ID    INTEGER    NOT NULL
)
)
IN NEWA_018
INDEX IN NEWA_018
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 13601 ENDING AT 14400,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 1900 ENDING AT 3675
)
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA19;
CREATE TABLE NEW_ORDERA19

```

```

(
NO_O_ID    INTEGER    NOT NULL,
NO_D_ID    SMALLINT   NOT NULL,
NO_W_ID    INTEGER    NOT NULL
)
)
IN NEWA_019
INDEX IN NEWA_019
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 14401 ENDING AT 15200,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 1900 ENDING AT 3675
)
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA20;
CREATE TABLE NEW_ORDERA20

```

```

(
NO_O_ID    INTEGER    NOT NULL,
NO_D_ID    SMALLINT   NOT NULL,
NO_W_ID    INTEGER    NOT NULL
)
)
IN NEWA_020
INDEX IN NEWA_020
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 15201 ENDING AT 16000,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 1900 ENDING AT 3675
)
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA21;
CREATE TABLE NEW_ORDERA21

```

```

(
NO_O_ID    INTEGER    NOT NULL,
NO_D_ID    SMALLINT   NOT NULL,
NO_W_ID    INTEGER    NOT NULL
)
)
IN NEWA_021
INDEX IN NEWA_021
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 16001 ENDING AT 16800,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 1900 ENDING AT 3675
)
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA22;
CREATE TABLE NEW_ORDERA22

```

```

(
NO_O_ID    INTEGER    NOT NULL,
NO_D_ID    SMALLINT   NOT NULL,
NO_W_ID    INTEGER    NOT NULL
)
IN NEWA_022
INDEX IN NEWA_022
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 16801 ENDING AT 17600,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 1900 ENDING AT 3675
)
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA23;
CREATE TABLE NEW_ORDERA23
(
NO_O_ID    INTEGER    NOT NULL,
NO_D_ID    SMALLINT   NOT NULL,
NO_W_ID    INTEGER    NOT NULL
)
IN NEWA_023
INDEX IN NEWA_023
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 17601 ENDING AT 18400,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 1900 ENDING AT 3675
)
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA24;
CREATE TABLE NEW_ORDERA24
(
NO_O_ID    INTEGER    NOT NULL,
NO_D_ID    SMALLINT   NOT NULL,
NO_W_ID    INTEGER    NOT NULL
)
IN NEWA_024
INDEX IN NEWA_024
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 18401 ENDING AT 19200,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 1900 ENDING AT 3675
)
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERA25;
CREATE TABLE NEW_ORDERA25
(
NO_O_ID    INTEGER    NOT NULL,
NO_D_ID    SMALLINT   NOT NULL,
NO_W_ID    INTEGER    NOT NULL
)
IN NEWA_025
INDEX IN NEWA_025
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 19201 ENDING AT 20000,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 1900 ENDING AT 3675
)
)
ALLOW OVERFLOW;
connect reset;

```

crtb_new_orderb.ddl

```

connect to TPCC in share mode;
DROP TABLE NEW_ORDERB1;
CREATE TABLE NEW_ORDERB1
(
NO_O_ID    INTEGER    NOT NULL,
NO_D_ID    SMALLINT   NOT NULL,
NO_W_ID    INTEGER    NOT NULL
)
IN NEWB_001
INDEX IN NEWB_001
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 1 ENDING AT 800,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 3676 ENDING AT 5451
)
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB2;
CREATE TABLE NEW_ORDERB2
(
NO_O_ID    INTEGER    NOT NULL,
NO_D_ID    SMALLINT   NOT NULL,
NO_W_ID    INTEGER    NOT NULL
)
IN NEWB_002
INDEX IN NEWB_002
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 801 ENDING AT 1600,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 3676 ENDING AT 5451
)
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB3;
CREATE TABLE NEW_ORDERB3
(
NO_O_ID    INTEGER    NOT NULL,
NO_D_ID    SMALLINT   NOT NULL,
NO_W_ID    INTEGER    NOT NULL
)
IN NEWB_003
INDEX IN NEWB_003
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 1601 ENDING AT 2400,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 3676 ENDING AT 5451
)
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB4;
CREATE TABLE NEW_ORDERB4
(
NO_O_ID    INTEGER    NOT NULL,
NO_D_ID    SMALLINT   NOT NULL,
NO_W_ID    INTEGER    NOT NULL
)
IN NEWB_004
INDEX IN NEWB_004
ORGANIZE BY KEY SEQUENCE (
NO_W_ID STARTING FROM 2401 ENDING AT 3200,
NO_D_ID STARTING FROM 1 ENDING AT 10,
NO_O_ID STARTING FROM 3676 ENDING AT 5451
)
)
ALLOW OVERFLOW;
connect reset;

```

```

connect to TPCC in share mode;
DROP TABLE NEW_ORDERB5;
CREATE TABLE NEW_ORDERB5
(
  NO_O_ID    INTEGER NOT NULL,
  NO_D_ID    SMALLINT NOT NULL,
  NO_W_ID    INTEGER NOT NULL
)
IN NEWB_005
INDEX IN NEWB_005
ORGANIZE BY KEY SEQUENCE (
  NO_W_ID STARTING FROM 3201 ENDING AT 4000,
  NO_D_ID STARTING FROM 1 ENDING AT 10,
  NO_O_ID STARTING FROM 3676 ENDING AT 5451
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB6;
CREATE TABLE NEW_ORDERB6

```

```

(
  NO_O_ID    INTEGER NOT NULL,
  NO_D_ID    SMALLINT NOT NULL,
  NO_W_ID    INTEGER NOT NULL
)
IN NEWB_006
INDEX IN NEWB_006
ORGANIZE BY KEY SEQUENCE (
  NO_W_ID STARTING FROM 4001 ENDING AT 4800,
  NO_D_ID STARTING FROM 1 ENDING AT 10,
  NO_O_ID STARTING FROM 3676 ENDING AT 5451
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB7;
CREATE TABLE NEW_ORDERB7

```

```

(
  NO_O_ID    INTEGER NOT NULL,
  NO_D_ID    SMALLINT NOT NULL,
  NO_W_ID    INTEGER NOT NULL
)
IN NEWB_007
INDEX IN NEWB_007
ORGANIZE BY KEY SEQUENCE (
  NO_W_ID STARTING FROM 4801 ENDING AT 5600,
  NO_D_ID STARTING FROM 1 ENDING AT 10,
  NO_O_ID STARTING FROM 3676 ENDING AT 5451
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB8;
CREATE TABLE NEW_ORDERB8

```

```

(
  NO_O_ID    INTEGER NOT NULL,
  NO_D_ID    SMALLINT NOT NULL,
  NO_W_ID    INTEGER NOT NULL
)
IN NEWB_008
INDEX IN NEWB_008
ORGANIZE BY KEY SEQUENCE (
  NO_W_ID STARTING FROM 5601 ENDING AT 6400,
  NO_D_ID STARTING FROM 1 ENDING AT 10,
  NO_O_ID STARTING FROM 3676 ENDING AT 5451
)
ALLOW OVERFLOW;

```

```

connect reset;

```

```

connect to TPCC in share mode;
DROP TABLE NEW_ORDERB9;
CREATE TABLE NEW_ORDERB9

```

```

(
  NO_O_ID    INTEGER NOT NULL,
  NO_D_ID    SMALLINT NOT NULL,
  NO_W_ID    INTEGER NOT NULL
)
IN NEWB_009
INDEX IN NEWB_009
ORGANIZE BY KEY SEQUENCE (
  NO_W_ID STARTING FROM 6401 ENDING AT 7200,
  NO_D_ID STARTING FROM 1 ENDING AT 10,
  NO_O_ID STARTING FROM 3676 ENDING AT 5451
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB10;
CREATE TABLE NEW_ORDERB10

```

```

(
  NO_O_ID    INTEGER NOT NULL,
  NO_D_ID    SMALLINT NOT NULL,
  NO_W_ID    INTEGER NOT NULL
)
IN NEWB_010
INDEX IN NEWB_010
ORGANIZE BY KEY SEQUENCE (
  NO_W_ID STARTING FROM 7201 ENDING AT 8000,
  NO_D_ID STARTING FROM 1 ENDING AT 10,
  NO_O_ID STARTING FROM 3676 ENDING AT 5451
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB11;
CREATE TABLE NEW_ORDERB11

```

```

(
  NO_O_ID    INTEGER NOT NULL,
  NO_D_ID    SMALLINT NOT NULL,
  NO_W_ID    INTEGER NOT NULL
)
IN NEWB_011
INDEX IN NEWB_011
ORGANIZE BY KEY SEQUENCE (
  NO_W_ID STARTING FROM 8001 ENDING AT 8800,
  NO_D_ID STARTING FROM 1 ENDING AT 10,
  NO_O_ID STARTING FROM 3676 ENDING AT 5451
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB12;
CREATE TABLE NEW_ORDERB12

```

```

(
  NO_O_ID    INTEGER NOT NULL,
  NO_D_ID    SMALLINT NOT NULL,
  NO_W_ID    INTEGER NOT NULL
)
IN NEWB_012
INDEX IN NEWB_012
ORGANIZE BY KEY SEQUENCE (
  NO_W_ID STARTING FROM 8801 ENDING AT 9600,
  NO_D_ID STARTING FROM 1 ENDING AT 10,
  NO_O_ID STARTING FROM 3676 ENDING AT 5451
)
ALLOW OVERFLOW;

```

```

connect reset;

```



```

connect to TPCC in share mode;
DROP TABLE NEW_ORDERB13;
CREATE TABLE NEW_ORDERB13
(
  NO_O_ID    INTEGER NOT NULL,
  NO_D_ID    SMALLINT NOT NULL,
  NO_W_ID    INTEGER NOT NULL
)
IN NEWB_013
INDEX IN NEWB_013
ORGANIZE BY KEY SEQUENCE (
  NO_W_ID STARTING FROM 9601 ENDING AT 10400,
  NO_D_ID STARTING FROM 1 ENDING AT 10,
  NO_O_ID STARTING FROM 3676 ENDING AT 5451
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB14;
CREATE TABLE NEW_ORDERB14
(
  NO_O_ID    INTEGER NOT NULL,
  NO_D_ID    SMALLINT NOT NULL,
  NO_W_ID    INTEGER NOT NULL
)
IN NEWB_014
INDEX IN NEWB_014
ORGANIZE BY KEY SEQUENCE (
  NO_W_ID STARTING FROM 10401 ENDING AT 11200,
  NO_D_ID STARTING FROM 1 ENDING AT 10,
  NO_O_ID STARTING FROM 3676 ENDING AT 5451
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB15;
CREATE TABLE NEW_ORDERB15
(
  NO_O_ID    INTEGER NOT NULL,
  NO_D_ID    SMALLINT NOT NULL,
  NO_W_ID    INTEGER NOT NULL
)
IN NEWB_015
INDEX IN NEWB_015
ORGANIZE BY KEY SEQUENCE (
  NO_W_ID STARTING FROM 11201 ENDING AT 12000,
  NO_D_ID STARTING FROM 1 ENDING AT 10,
  NO_O_ID STARTING FROM 3676 ENDING AT 5451
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB16;
CREATE TABLE NEW_ORDERB16
(
  NO_O_ID    INTEGER NOT NULL,
  NO_D_ID    SMALLINT NOT NULL,
  NO_W_ID    INTEGER NOT NULL
)
IN NEWB_016
INDEX IN NEWB_016
ORGANIZE BY KEY SEQUENCE (
  NO_W_ID STARTING FROM 12001 ENDING AT 12800,
  NO_D_ID STARTING FROM 1 ENDING AT 10,
  NO_O_ID STARTING FROM 3676 ENDING AT 5451
)
ALLOW OVERFLOW;

```

connect reset;

```

connect to TPCC in share mode;
DROP TABLE NEW_ORDERB17;
CREATE TABLE NEW_ORDERB17
(
  NO_O_ID    INTEGER NOT NULL,
  NO_D_ID    SMALLINT NOT NULL,
  NO_W_ID    INTEGER NOT NULL
)
IN NEWB_017
INDEX IN NEWB_017
ORGANIZE BY KEY SEQUENCE (
  NO_W_ID STARTING FROM 12801 ENDING AT 13600,
  NO_D_ID STARTING FROM 1 ENDING AT 10,
  NO_O_ID STARTING FROM 3676 ENDING AT 5451
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB18;
CREATE TABLE NEW_ORDERB18
(
  NO_O_ID    INTEGER NOT NULL,
  NO_D_ID    SMALLINT NOT NULL,
  NO_W_ID    INTEGER NOT NULL
)
IN NEWB_018
INDEX IN NEWB_018
ORGANIZE BY KEY SEQUENCE (
  NO_W_ID STARTING FROM 13601 ENDING AT 14400,
  NO_D_ID STARTING FROM 1 ENDING AT 10,
  NO_O_ID STARTING FROM 3676 ENDING AT 5451
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB19;
CREATE TABLE NEW_ORDERB19
(
  NO_O_ID    INTEGER NOT NULL,
  NO_D_ID    SMALLINT NOT NULL,
  NO_W_ID    INTEGER NOT NULL
)
IN NEWB_019
INDEX IN NEWB_019
ORGANIZE BY KEY SEQUENCE (
  NO_W_ID STARTING FROM 14401 ENDING AT 15200,
  NO_D_ID STARTING FROM 1 ENDING AT 10,
  NO_O_ID STARTING FROM 3676 ENDING AT 5451
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB20;
CREATE TABLE NEW_ORDERB20
(
  NO_O_ID    INTEGER NOT NULL,
  NO_D_ID    SMALLINT NOT NULL,
  NO_W_ID    INTEGER NOT NULL
)
IN NEWB_020
INDEX IN NEWB_020
ORGANIZE BY KEY SEQUENCE (
  NO_W_ID STARTING FROM 15201 ENDING AT 16000,
  NO_D_ID STARTING FROM 1 ENDING AT 10,
  NO_O_ID STARTING FROM 3676 ENDING AT 5451
)
ALLOW OVERFLOW;

```

connect reset;

```

connect to TPCC in share mode;
DROP TABLE NEW_ORDERB21;
CREATE TABLE NEW_ORDERB21
(
  NO_O_ID    INTEGER NOT NULL,
  NO_D_ID    SMALLINT NOT NULL,
  NO_W_ID    INTEGER NOT NULL
)
IN NEWB_021
INDEX IN NEWB_021
ORGANIZE BY KEY SEQUENCE (
  NO_W_ID STARTING FROM 16001 ENDING AT 16800,
  NO_D_ID STARTING FROM 1 ENDING AT 10,
  NO_O_ID STARTING FROM 3676 ENDING AT 5451
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB22;
CREATE TABLE NEW_ORDERB22

```

```

(
  NO_O_ID    INTEGER NOT NULL,
  NO_D_ID    SMALLINT NOT NULL,
  NO_W_ID    INTEGER NOT NULL
)
IN NEWB_022
INDEX IN NEWB_022
ORGANIZE BY KEY SEQUENCE (
  NO_W_ID STARTING FROM 16801 ENDING AT 17600,
  NO_D_ID STARTING FROM 1 ENDING AT 10,
  NO_O_ID STARTING FROM 3676 ENDING AT 5451
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB23;
CREATE TABLE NEW_ORDERB23

```

```

(
  NO_O_ID    INTEGER NOT NULL,
  NO_D_ID    SMALLINT NOT NULL,
  NO_W_ID    INTEGER NOT NULL
)
IN NEWB_023
INDEX IN NEWB_023
ORGANIZE BY KEY SEQUENCE (
  NO_W_ID STARTING FROM 17601 ENDING AT 18400,
  NO_D_ID STARTING FROM 1 ENDING AT 10,
  NO_O_ID STARTING FROM 3676 ENDING AT 5451
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE NEW_ORDERB24;
CREATE TABLE NEW_ORDERB24

```

```

(
  NO_O_ID    INTEGER NOT NULL,
  NO_D_ID    SMALLINT NOT NULL,
  NO_W_ID    INTEGER NOT NULL
)
IN NEWB_024
INDEX IN NEWB_024
ORGANIZE BY KEY SEQUENCE (
  NO_W_ID STARTING FROM 18401 ENDING AT 19200,
  NO_D_ID STARTING FROM 1 ENDING AT 10,
  NO_O_ID STARTING FROM 3676 ENDING AT 5451
)
ALLOW OVERFLOW;

```

```

connect reset;

```

```

connect to TPCC in share mode;
DROP TABLE NEW_ORDERB25;
CREATE TABLE NEW_ORDERB25
(
  NO_O_ID    INTEGER NOT NULL,
  NO_D_ID    SMALLINT NOT NULL,
  NO_W_ID    INTEGER NOT NULL
)
IN NEWB_025
INDEX IN NEWB_025
ORGANIZE BY KEY SEQUENCE (
  NO_W_ID STARTING FROM 19201 ENDING AT 20000,
  NO_D_ID STARTING FROM 1 ENDING AT 10,
  NO_O_ID STARTING FROM 3676 ENDING AT 5451
)
ALLOW OVERFLOW;

```

```

connect reset;

```

crtb_stock.ddl

```

connect to TPCC in share mode;
DROP TABLE STOCK1;
CREATE TABLE STOCK1

```

```

(
  S_REMOTE_CNT INTEGER NOT NULL,
  S_QUANTITY   INTEGER NOT NULL,
  S_ORDER_CNT  INTEGER NOT NULL,
  S_YTD        INTEGER NOT NULL,
  S_DATA       VARCHAR(50) NOT NULL,
  S_DIST_01    CHAR(24) NOT NULL,
  S_DIST_02    CHAR(24) NOT NULL,
  S_DIST_03    CHAR(24) NOT NULL,
  S_DIST_04    CHAR(24) NOT NULL,
  S_DIST_05    CHAR(24) NOT NULL,
  S_DIST_06    CHAR(24) NOT NULL,
  S_DIST_07    CHAR(24) NOT NULL,
  S_DIST_08    CHAR(24) NOT NULL,
  S_DIST_09    CHAR(24) NOT NULL,
  S_DIST_10    CHAR(24) NOT NULL,
  S_I_ID       INTEGER NOT NULL,
  S_W_ID       INTEGER NOT NULL
)
IN STK_001
INDEX IN STK_001
ORGANIZE BY KEY SEQUENCE (
  S_I_ID STARTING FROM 1 ENDING AT 10000,
  S_W_ID STARTING FROM 1 ENDING AT 800
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK2;
CREATE TABLE STOCK2

```

```

(
  S_REMOTE_CNT INTEGER NOT NULL,
  S_QUANTITY   INTEGER NOT NULL,
  S_ORDER_CNT  INTEGER NOT NULL,
  S_YTD        INTEGER NOT NULL,
  S_DATA       VARCHAR(50) NOT NULL,
  S_DIST_01    CHAR(24) NOT NULL,
  S_DIST_02    CHAR(24) NOT NULL,
  S_DIST_03    CHAR(24) NOT NULL,
  S_DIST_04    CHAR(24) NOT NULL,
  S_DIST_05    CHAR(24) NOT NULL,
  S_DIST_06    CHAR(24) NOT NULL,
  S_DIST_07    CHAR(24) NOT NULL,
  S_DIST_08    CHAR(24) NOT NULL,

```

```

S_DIST_09 CHAR(24) NOT NULL,
S_DIST_10 CHAR(24) NOT NULL,
S_I_ID INTEGER NOT NULL,
S_W_ID INTEGER NOT NULL
)
IN STK_002
INDEX IN STK_002
ORGANIZE BY KEY SEQUENCE (
S_I_ID STARTING FROM 1 ENDING AT 10000,
S_W_ID STARTING FROM 801 ENDING AT 1600
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK3;
CREATE TABLE STOCK3
(
S_REMOTE_CNT INTEGER NOT NULL,
S_QUANTITY INTEGER NOT NULL,
S_ORDER_CNT INTEGER NOT NULL,
S_YTD INTEGER NOT NULL,
S_DATA VARCHAR(50) NOT NULL,
S_DIST_01 CHAR(24) NOT NULL,
S_DIST_02 CHAR(24) NOT NULL,
S_DIST_03 CHAR(24) NOT NULL,
S_DIST_04 CHAR(24) NOT NULL,
S_DIST_05 CHAR(24) NOT NULL,
S_DIST_06 CHAR(24) NOT NULL,
S_DIST_07 CHAR(24) NOT NULL,
S_DIST_08 CHAR(24) NOT NULL,
S_DIST_09 CHAR(24) NOT NULL,
S_DIST_10 CHAR(24) NOT NULL,
S_I_ID INTEGER NOT NULL,
S_W_ID INTEGER NOT NULL
)
IN STK_003
INDEX IN STK_003
ORGANIZE BY KEY SEQUENCE (
S_I_ID STARTING FROM 1 ENDING AT 10000,
S_W_ID STARTING FROM 1601 ENDING AT 2400
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK4;
CREATE TABLE STOCK4
(
S_REMOTE_CNT INTEGER NOT NULL,
S_QUANTITY INTEGER NOT NULL,
S_ORDER_CNT INTEGER NOT NULL,
S_YTD INTEGER NOT NULL,
S_DATA VARCHAR(50) NOT NULL,
S_DIST_01 CHAR(24) NOT NULL,
S_DIST_02 CHAR(24) NOT NULL,
S_DIST_03 CHAR(24) NOT NULL,
S_DIST_04 CHAR(24) NOT NULL,
S_DIST_05 CHAR(24) NOT NULL,
S_DIST_06 CHAR(24) NOT NULL,
S_DIST_07 CHAR(24) NOT NULL,
S_DIST_08 CHAR(24) NOT NULL,
S_DIST_09 CHAR(24) NOT NULL,
S_DIST_10 CHAR(24) NOT NULL,
S_I_ID INTEGER NOT NULL,
S_W_ID INTEGER NOT NULL
)
IN STK_004
INDEX IN STK_004
ORGANIZE BY KEY SEQUENCE (

```

```

S_I_ID STARTING FROM 1 ENDING AT 10000,
S_W_ID STARTING FROM 2401 ENDING AT 3200
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK5;
CREATE TABLE STOCK5
(
S_REMOTE_CNT INTEGER NOT NULL,
S_QUANTITY INTEGER NOT NULL,
S_ORDER_CNT INTEGER NOT NULL,
S_YTD INTEGER NOT NULL,
S_DATA VARCHAR(50) NOT NULL,
S_DIST_01 CHAR(24) NOT NULL,
S_DIST_02 CHAR(24) NOT NULL,
S_DIST_03 CHAR(24) NOT NULL,
S_DIST_04 CHAR(24) NOT NULL,
S_DIST_05 CHAR(24) NOT NULL,
S_DIST_06 CHAR(24) NOT NULL,
S_DIST_07 CHAR(24) NOT NULL,
S_DIST_08 CHAR(24) NOT NULL,
S_DIST_09 CHAR(24) NOT NULL,
S_DIST_10 CHAR(24) NOT NULL,
S_I_ID INTEGER NOT NULL,
S_W_ID INTEGER NOT NULL
)
IN STK_005
INDEX IN STK_005
ORGANIZE BY KEY SEQUENCE (
S_I_ID STARTING FROM 1 ENDING AT 10000,
S_W_ID STARTING FROM 3201 ENDING AT 4000
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK6;
CREATE TABLE STOCK6
(
S_REMOTE_CNT INTEGER NOT NULL,
S_QUANTITY INTEGER NOT NULL,
S_ORDER_CNT INTEGER NOT NULL,
S_YTD INTEGER NOT NULL,
S_DATA VARCHAR(50) NOT NULL,
S_DIST_01 CHAR(24) NOT NULL,
S_DIST_02 CHAR(24) NOT NULL,
S_DIST_03 CHAR(24) NOT NULL,
S_DIST_04 CHAR(24) NOT NULL,
S_DIST_05 CHAR(24) NOT NULL,
S_DIST_06 CHAR(24) NOT NULL,
S_DIST_07 CHAR(24) NOT NULL,
S_DIST_08 CHAR(24) NOT NULL,
S_DIST_09 CHAR(24) NOT NULL,
S_DIST_10 CHAR(24) NOT NULL,
S_I_ID INTEGER NOT NULL,
S_W_ID INTEGER NOT NULL
)
IN STK_006
INDEX IN STK_006
ORGANIZE BY KEY SEQUENCE (
S_I_ID STARTING FROM 1 ENDING AT 10000,
S_W_ID STARTING FROM 4001 ENDING AT 4800
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK7;
CREATE TABLE STOCK7

```

```

(
S_REMOTE_CNT INTEGER NOT NULL,
S_QUANTITY INTEGER NOT NULL,
S_ORDER_CNT INTEGER NOT NULL,
S_YTD INTEGER NOT NULL,
S_DATA VARCHAR(50) NOT NULL,
S_DIST_01 CHAR(24) NOT NULL,
S_DIST_02 CHAR(24) NOT NULL,
S_DIST_03 CHAR(24) NOT NULL,
S_DIST_04 CHAR(24) NOT NULL,
S_DIST_05 CHAR(24) NOT NULL,
S_DIST_06 CHAR(24) NOT NULL,
S_DIST_07 CHAR(24) NOT NULL,
S_DIST_08 CHAR(24) NOT NULL,
S_DIST_09 CHAR(24) NOT NULL,
S_DIST_10 CHAR(24) NOT NULL,
S_I_ID INTEGER NOT NULL,
S_W_ID INTEGER NOT NULL
)
IN STK_007
INDEX IN STK_007
ORGANIZE BY KEY SEQUENCE (
S_I_ID STARTING FROM 1 ENDING AT 10000,
S_W_ID STARTING FROM 4801 ENDING AT 5600
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK8;
CREATE TABLE STOCK8
(
S_REMOTE_CNT INTEGER NOT NULL,
S_QUANTITY INTEGER NOT NULL,
S_ORDER_CNT INTEGER NOT NULL,
S_YTD INTEGER NOT NULL,
S_DATA VARCHAR(50) NOT NULL,
S_DIST_01 CHAR(24) NOT NULL,
S_DIST_02 CHAR(24) NOT NULL,
S_DIST_03 CHAR(24) NOT NULL,
S_DIST_04 CHAR(24) NOT NULL,
S_DIST_05 CHAR(24) NOT NULL,
S_DIST_06 CHAR(24) NOT NULL,
S_DIST_07 CHAR(24) NOT NULL,
S_DIST_08 CHAR(24) NOT NULL,
S_DIST_09 CHAR(24) NOT NULL,
S_DIST_10 CHAR(24) NOT NULL,
S_I_ID INTEGER NOT NULL,
S_W_ID INTEGER NOT NULL
)
IN STK_008
INDEX IN STK_008
ORGANIZE BY KEY SEQUENCE (
S_I_ID STARTING FROM 1 ENDING AT 10000,
S_W_ID STARTING FROM 5601 ENDING AT 6400
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK9;
CREATE TABLE STOCK9
(
S_REMOTE_CNT INTEGER NOT NULL,
S_QUANTITY INTEGER NOT NULL,
S_ORDER_CNT INTEGER NOT NULL,
S_YTD INTEGER NOT NULL,
S_DATA VARCHAR(50) NOT NULL,
S_DIST_01 CHAR(24) NOT NULL,
S_DIST_02 CHAR(24) NOT NULL,

```

```

S_DIST_03 CHAR(24) NOT NULL,
S_DIST_04 CHAR(24) NOT NULL,
S_DIST_05 CHAR(24) NOT NULL,
S_DIST_06 CHAR(24) NOT NULL,
S_DIST_07 CHAR(24) NOT NULL,
S_DIST_08 CHAR(24) NOT NULL,
S_DIST_09 CHAR(24) NOT NULL,
S_DIST_10 CHAR(24) NOT NULL,
S_I_ID INTEGER NOT NULL,
S_W_ID INTEGER NOT NULL
)
IN STK_009
INDEX IN STK_009
ORGANIZE BY KEY SEQUENCE (
S_I_ID STARTING FROM 1 ENDING AT 10000,
S_W_ID STARTING FROM 6401 ENDING AT 7200
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK10;
CREATE TABLE STOCK10
(
S_REMOTE_CNT INTEGER NOT NULL,
S_QUANTITY INTEGER NOT NULL,
S_ORDER_CNT INTEGER NOT NULL,
S_YTD INTEGER NOT NULL,
S_DATA VARCHAR(50) NOT NULL,
S_DIST_01 CHAR(24) NOT NULL,
S_DIST_02 CHAR(24) NOT NULL,
S_DIST_03 CHAR(24) NOT NULL,
S_DIST_04 CHAR(24) NOT NULL,
S_DIST_05 CHAR(24) NOT NULL,
S_DIST_06 CHAR(24) NOT NULL,
S_DIST_07 CHAR(24) NOT NULL,
S_DIST_08 CHAR(24) NOT NULL,
S_DIST_09 CHAR(24) NOT NULL,
S_DIST_10 CHAR(24) NOT NULL,
S_I_ID INTEGER NOT NULL,
S_W_ID INTEGER NOT NULL
)
IN STK_010
INDEX IN STK_010
ORGANIZE BY KEY SEQUENCE (
S_I_ID STARTING FROM 1 ENDING AT 10000,
S_W_ID STARTING FROM 7201 ENDING AT 8000
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK11;
CREATE TABLE STOCK11
(
S_REMOTE_CNT INTEGER NOT NULL,
S_QUANTITY INTEGER NOT NULL,
S_ORDER_CNT INTEGER NOT NULL,
S_YTD INTEGER NOT NULL,
S_DATA VARCHAR(50) NOT NULL,
S_DIST_01 CHAR(24) NOT NULL,
S_DIST_02 CHAR(24) NOT NULL,
S_DIST_03 CHAR(24) NOT NULL,
S_DIST_04 CHAR(24) NOT NULL,
S_DIST_05 CHAR(24) NOT NULL,
S_DIST_06 CHAR(24) NOT NULL,
S_DIST_07 CHAR(24) NOT NULL,
S_DIST_08 CHAR(24) NOT NULL,
S_DIST_09 CHAR(24) NOT NULL,
S_DIST_10 CHAR(24) NOT NULL,

```

```

S_I_ID    INTEGER NOT NULL,
S_W_ID    INTEGER NOT NULL
)
IN STK_011
INDEX IN STK_011
ORGANIZE BY KEY SEQUENCE (
S_I_ID STARTING FROM 1 ENDING AT 10000,
S_W_ID STARTING FROM 8001 ENDING AT 8800
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK12;
CREATE TABLE STOCK12
(
S_REMOTE_CNT INTEGER NOT NULL,
S_QUANTITY  INTEGER NOT NULL,
S_ORDER_CNT  INTEGER NOT NULL,
S_YTD       INTEGER NOT NULL,
S_DATA      VARCHAR(50) NOT NULL,
S_DIST_01   CHAR(24) NOT NULL,
S_DIST_02   CHAR(24) NOT NULL,
S_DIST_03   CHAR(24) NOT NULL,
S_DIST_04   CHAR(24) NOT NULL,
S_DIST_05   CHAR(24) NOT NULL,
S_DIST_06   CHAR(24) NOT NULL,
S_DIST_07   CHAR(24) NOT NULL,
S_DIST_08   CHAR(24) NOT NULL,
S_DIST_09   CHAR(24) NOT NULL,
S_DIST_10   CHAR(24) NOT NULL,
S_I_ID      INTEGER NOT NULL,
S_W_ID      INTEGER NOT NULL
)
IN STK_012
INDEX IN STK_012
ORGANIZE BY KEY SEQUENCE (
S_I_ID STARTING FROM 1 ENDING AT 10000,
S_W_ID STARTING FROM 8801 ENDING AT 9600
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK13;
CREATE TABLE STOCK13
(
S_REMOTE_CNT INTEGER NOT NULL,
S_QUANTITY  INTEGER NOT NULL,
S_ORDER_CNT  INTEGER NOT NULL,
S_YTD       INTEGER NOT NULL,
S_DATA      VARCHAR(50) NOT NULL,
S_DIST_01   CHAR(24) NOT NULL,
S_DIST_02   CHAR(24) NOT NULL,
S_DIST_03   CHAR(24) NOT NULL,
S_DIST_04   CHAR(24) NOT NULL,
S_DIST_05   CHAR(24) NOT NULL,
S_DIST_06   CHAR(24) NOT NULL,
S_DIST_07   CHAR(24) NOT NULL,
S_DIST_08   CHAR(24) NOT NULL,
S_DIST_09   CHAR(24) NOT NULL,
S_DIST_10   CHAR(24) NOT NULL,
S_I_ID      INTEGER NOT NULL,
S_W_ID      INTEGER NOT NULL
)
IN STK_013
INDEX IN STK_013
ORGANIZE BY KEY SEQUENCE (
S_I_ID STARTING FROM 1 ENDING AT 10000,
S_W_ID STARTING FROM 9601 ENDING AT 10400

```

```

)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK14;
CREATE TABLE STOCK14
(
S_REMOTE_CNT INTEGER NOT NULL,
S_QUANTITY  INTEGER NOT NULL,
S_ORDER_CNT  INTEGER NOT NULL,
S_YTD       INTEGER NOT NULL,
S_DATA      VARCHAR(50) NOT NULL,
S_DIST_01   CHAR(24) NOT NULL,
S_DIST_02   CHAR(24) NOT NULL,
S_DIST_03   CHAR(24) NOT NULL,
S_DIST_04   CHAR(24) NOT NULL,
S_DIST_05   CHAR(24) NOT NULL,
S_DIST_06   CHAR(24) NOT NULL,
S_DIST_07   CHAR(24) NOT NULL,
S_DIST_08   CHAR(24) NOT NULL,
S_DIST_09   CHAR(24) NOT NULL,
S_DIST_10   CHAR(24) NOT NULL,
S_I_ID      INTEGER NOT NULL,
S_W_ID      INTEGER NOT NULL
)
IN STK_014
INDEX IN STK_014
ORGANIZE BY KEY SEQUENCE (
S_I_ID STARTING FROM 1 ENDING AT 10000,
S_W_ID STARTING FROM 10401 ENDING AT 11200
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK15;
CREATE TABLE STOCK15
(
S_REMOTE_CNT INTEGER NOT NULL,
S_QUANTITY  INTEGER NOT NULL,
S_ORDER_CNT  INTEGER NOT NULL,
S_YTD       INTEGER NOT NULL,
S_DATA      VARCHAR(50) NOT NULL,
S_DIST_01   CHAR(24) NOT NULL,
S_DIST_02   CHAR(24) NOT NULL,
S_DIST_03   CHAR(24) NOT NULL,
S_DIST_04   CHAR(24) NOT NULL,
S_DIST_05   CHAR(24) NOT NULL,
S_DIST_06   CHAR(24) NOT NULL,
S_DIST_07   CHAR(24) NOT NULL,
S_DIST_08   CHAR(24) NOT NULL,
S_DIST_09   CHAR(24) NOT NULL,
S_DIST_10   CHAR(24) NOT NULL,
S_I_ID      INTEGER NOT NULL,
S_W_ID      INTEGER NOT NULL
)
IN STK_015
INDEX IN STK_015
ORGANIZE BY KEY SEQUENCE (
S_I_ID STARTING FROM 1 ENDING AT 10000,
S_W_ID STARTING FROM 11201 ENDING AT 12000
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK16;
CREATE TABLE STOCK16
(
S_REMOTE_CNT INTEGER NOT NULL,

```

```

S_QUANTITY INTEGER NOT NULL,
S_ORDER_CNT INTEGER NOT NULL,
S_YTD INTEGER NOT NULL,
S_DATA VARCHAR(50) NOT NULL,
S_DIST_01 CHAR(24) NOT NULL,
S_DIST_02 CHAR(24) NOT NULL,
S_DIST_03 CHAR(24) NOT NULL,
S_DIST_04 CHAR(24) NOT NULL,
S_DIST_05 CHAR(24) NOT NULL,
S_DIST_06 CHAR(24) NOT NULL,
S_DIST_07 CHAR(24) NOT NULL,
S_DIST_08 CHAR(24) NOT NULL,
S_DIST_09 CHAR(24) NOT NULL,
S_DIST_10 CHAR(24) NOT NULL,
S_I_ID INTEGER NOT NULL,
S_W_ID INTEGER NOT NULL
)
IN STK_016
INDEX IN STK_016
ORGANIZE BY KEY SEQUENCE (
S_I_ID STARTING FROM 1 ENDING AT 10000,
S_W_ID STARTING FROM 12001 ENDING AT 12800
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK17;
CREATE TABLE STOCK17
(
S_REMOTE_CNT INTEGER NOT NULL,
S_QUANTITY INTEGER NOT NULL,
S_ORDER_CNT INTEGER NOT NULL,
S_YTD INTEGER NOT NULL,
S_DATA VARCHAR(50) NOT NULL,
S_DIST_01 CHAR(24) NOT NULL,
S_DIST_02 CHAR(24) NOT NULL,
S_DIST_03 CHAR(24) NOT NULL,
S_DIST_04 CHAR(24) NOT NULL,
S_DIST_05 CHAR(24) NOT NULL,
S_DIST_06 CHAR(24) NOT NULL,
S_DIST_07 CHAR(24) NOT NULL,
S_DIST_08 CHAR(24) NOT NULL,
S_DIST_09 CHAR(24) NOT NULL,
S_DIST_10 CHAR(24) NOT NULL,
S_I_ID INTEGER NOT NULL,
S_W_ID INTEGER NOT NULL
)
IN STK_017
INDEX IN STK_017
ORGANIZE BY KEY SEQUENCE (
S_I_ID STARTING FROM 1 ENDING AT 10000,
S_W_ID STARTING FROM 12801 ENDING AT 13600
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK18;
CREATE TABLE STOCK18
(
S_REMOTE_CNT INTEGER NOT NULL,
S_QUANTITY INTEGER NOT NULL,
S_ORDER_CNT INTEGER NOT NULL,
S_YTD INTEGER NOT NULL,
S_DATA VARCHAR(50) NOT NULL,
S_DIST_01 CHAR(24) NOT NULL,
S_DIST_02 CHAR(24) NOT NULL,
S_DIST_03 CHAR(24) NOT NULL,
S_DIST_04 CHAR(24) NOT NULL,

```

```

S_DIST_05 CHAR(24) NOT NULL,
S_DIST_06 CHAR(24) NOT NULL,
S_DIST_07 CHAR(24) NOT NULL,
S_DIST_08 CHAR(24) NOT NULL,
S_DIST_09 CHAR(24) NOT NULL,
S_DIST_10 CHAR(24) NOT NULL,
S_I_ID INTEGER NOT NULL,
S_W_ID INTEGER NOT NULL
)
IN STK_018
INDEX IN STK_018
ORGANIZE BY KEY SEQUENCE (
S_I_ID STARTING FROM 1 ENDING AT 10000,
S_W_ID STARTING FROM 13601 ENDING AT 14400
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK19;
CREATE TABLE STOCK19
(
S_REMOTE_CNT INTEGER NOT NULL,
S_QUANTITY INTEGER NOT NULL,
S_ORDER_CNT INTEGER NOT NULL,
S_YTD INTEGER NOT NULL,
S_DATA VARCHAR(50) NOT NULL,
S_DIST_01 CHAR(24) NOT NULL,
S_DIST_02 CHAR(24) NOT NULL,
S_DIST_03 CHAR(24) NOT NULL,
S_DIST_04 CHAR(24) NOT NULL,
S_DIST_05 CHAR(24) NOT NULL,
S_DIST_06 CHAR(24) NOT NULL,
S_DIST_07 CHAR(24) NOT NULL,
S_DIST_08 CHAR(24) NOT NULL,
S_DIST_09 CHAR(24) NOT NULL,
S_DIST_10 CHAR(24) NOT NULL,
S_I_ID INTEGER NOT NULL,
S_W_ID INTEGER NOT NULL
)
IN STK_019
INDEX IN STK_019
ORGANIZE BY KEY SEQUENCE (
S_I_ID STARTING FROM 1 ENDING AT 10000,
S_W_ID STARTING FROM 14401 ENDING AT 15200
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK20;
CREATE TABLE STOCK20
(
S_REMOTE_CNT INTEGER NOT NULL,
S_QUANTITY INTEGER NOT NULL,
S_ORDER_CNT INTEGER NOT NULL,
S_YTD INTEGER NOT NULL,
S_DATA VARCHAR(50) NOT NULL,
S_DIST_01 CHAR(24) NOT NULL,
S_DIST_02 CHAR(24) NOT NULL,
S_DIST_03 CHAR(24) NOT NULL,
S_DIST_04 CHAR(24) NOT NULL,
S_DIST_05 CHAR(24) NOT NULL,
S_DIST_06 CHAR(24) NOT NULL,
S_DIST_07 CHAR(24) NOT NULL,
S_DIST_08 CHAR(24) NOT NULL,
S_DIST_09 CHAR(24) NOT NULL,
S_DIST_10 CHAR(24) NOT NULL,
S_I_ID INTEGER NOT NULL,
S_W_ID INTEGER NOT NULL

```

```

)
IN STK_020
INDEX IN STK_020
ORGANIZE BY KEY SEQUENCE (
  S_I_ID STARTING FROM 1 ENDING AT 10000,
  S_W_ID STARTING FROM 15201 ENDING AT 16000
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK21;
CREATE TABLE STOCK21
(
  S_REMOTE_CNT INTEGER NOT NULL,
  S_QUANTITY INTEGER NOT NULL,
  S_ORDER_CNT INTEGER NOT NULL,
  S_YTD INTEGER NOT NULL,
  S_DATA VARCHAR(50) NOT NULL,
  S_DIST_01 CHAR(24) NOT NULL,
  S_DIST_02 CHAR(24) NOT NULL,
  S_DIST_03 CHAR(24) NOT NULL,
  S_DIST_04 CHAR(24) NOT NULL,
  S_DIST_05 CHAR(24) NOT NULL,
  S_DIST_06 CHAR(24) NOT NULL,
  S_DIST_07 CHAR(24) NOT NULL,
  S_DIST_08 CHAR(24) NOT NULL,
  S_DIST_09 CHAR(24) NOT NULL,
  S_DIST_10 CHAR(24) NOT NULL,
  S_I_ID INTEGER NOT NULL,
  S_W_ID INTEGER NOT NULL
)
IN STK_021
INDEX IN STK_021
ORGANIZE BY KEY SEQUENCE (
  S_I_ID STARTING FROM 1 ENDING AT 10000,
  S_W_ID STARTING FROM 16001 ENDING AT 16800
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK22;
CREATE TABLE STOCK22
(
  S_REMOTE_CNT INTEGER NOT NULL,
  S_QUANTITY INTEGER NOT NULL,
  S_ORDER_CNT INTEGER NOT NULL,
  S_YTD INTEGER NOT NULL,
  S_DATA VARCHAR(50) NOT NULL,
  S_DIST_01 CHAR(24) NOT NULL,
  S_DIST_02 CHAR(24) NOT NULL,
  S_DIST_03 CHAR(24) NOT NULL,
  S_DIST_04 CHAR(24) NOT NULL,
  S_DIST_05 CHAR(24) NOT NULL,
  S_DIST_06 CHAR(24) NOT NULL,
  S_DIST_07 CHAR(24) NOT NULL,
  S_DIST_08 CHAR(24) NOT NULL,
  S_DIST_09 CHAR(24) NOT NULL,
  S_DIST_10 CHAR(24) NOT NULL,
  S_I_ID INTEGER NOT NULL,
  S_W_ID INTEGER NOT NULL
)
IN STK_022
INDEX IN STK_022
ORGANIZE BY KEY SEQUENCE (
  S_I_ID STARTING FROM 1 ENDING AT 10000,
  S_W_ID STARTING FROM 16801 ENDING AT 17600
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK23;
CREATE TABLE STOCK23
(
  S_REMOTE_CNT INTEGER NOT NULL,
  S_QUANTITY INTEGER NOT NULL,
  S_ORDER_CNT INTEGER NOT NULL,
  S_YTD INTEGER NOT NULL,
  S_DATA VARCHAR(50) NOT NULL,
  S_DIST_01 CHAR(24) NOT NULL,
  S_DIST_02 CHAR(24) NOT NULL,
  S_DIST_03 CHAR(24) NOT NULL,
  S_DIST_04 CHAR(24) NOT NULL,
  S_DIST_05 CHAR(24) NOT NULL,
  S_DIST_06 CHAR(24) NOT NULL,
  S_DIST_07 CHAR(24) NOT NULL,
  S_DIST_08 CHAR(24) NOT NULL,
  S_DIST_09 CHAR(24) NOT NULL,
  S_DIST_10 CHAR(24) NOT NULL,
  S_I_ID INTEGER NOT NULL,
  S_W_ID INTEGER NOT NULL
)
IN STK_023
INDEX IN STK_023
ORGANIZE BY KEY SEQUENCE (
  S_I_ID STARTING FROM 1 ENDING AT 10000,
  S_W_ID STARTING FROM 17601 ENDING AT 18400
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK24;
CREATE TABLE STOCK24
(
  S_REMOTE_CNT INTEGER NOT NULL,
  S_QUANTITY INTEGER NOT NULL,
  S_ORDER_CNT INTEGER NOT NULL,
  S_YTD INTEGER NOT NULL,
  S_DATA VARCHAR(50) NOT NULL,
  S_DIST_01 CHAR(24) NOT NULL,
  S_DIST_02 CHAR(24) NOT NULL,
  S_DIST_03 CHAR(24) NOT NULL,
  S_DIST_04 CHAR(24) NOT NULL,
  S_DIST_05 CHAR(24) NOT NULL,
  S_DIST_06 CHAR(24) NOT NULL,
  S_DIST_07 CHAR(24) NOT NULL,
  S_DIST_08 CHAR(24) NOT NULL,
  S_DIST_09 CHAR(24) NOT NULL,
  S_DIST_10 CHAR(24) NOT NULL,
  S_I_ID INTEGER NOT NULL,
  S_W_ID INTEGER NOT NULL
)
IN STK_024
INDEX IN STK_024
ORGANIZE BY KEY SEQUENCE (
  S_I_ID STARTING FROM 1 ENDING AT 10000,
  S_W_ID STARTING FROM 18401 ENDING AT 19200
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE STOCK25;
CREATE TABLE STOCK25
(
  S_REMOTE_CNT INTEGER NOT NULL,
  S_QUANTITY INTEGER NOT NULL,
  S_ORDER_CNT INTEGER NOT NULL,

```

```

S_YTD      INTEGER NOT NULL,
S_DATA     VARCHAR(50) NOT NULL,
S_DIST_01  CHAR(24) NOT NULL,
S_DIST_02  CHAR(24) NOT NULL,
S_DIST_03  CHAR(24) NOT NULL,
S_DIST_04  CHAR(24) NOT NULL,
S_DIST_05  CHAR(24) NOT NULL,
S_DIST_06  CHAR(24) NOT NULL,
S_DIST_07  CHAR(24) NOT NULL,
S_DIST_08  CHAR(24) NOT NULL,
S_DIST_09  CHAR(24) NOT NULL,
S_DIST_10  CHAR(24) NOT NULL,
S_I_ID     INTEGER NOT NULL,
S_W_ID     INTEGER NOT NULL
)
IN STK_025
INDEX IN STK_025
ORGANIZE BY KEY SEQUENCE (
S_I_ID STARTING FROM 1 ENDING AT 100000,
S_W_ID STARTING FROM 19201 ENDING AT 20000
)
ALLOW OVERFLOW;
connect reset;

```

crtb_history.ddl

```

connect to TPCC in share mode;
DROP TABLE HISTORY1;
CREATE TABLE HISTORY1

```

```

(
H_C_ID     INTEGER NOT NULL,
H_C_D_ID   SMALLINT NOT NULL,
H_C_W_ID   INTEGER NOT NULL,
H_D_ID     SMALLINT NOT NULL,
H_W_ID     INTEGER NOT NULL,
H_DATE     BIGINT NOT NULL,
H_AMOUNT   INTEGER NOT NULL,
H_DATA     CHAR(24) NOT NULL
)
IN HST_001
INDEX IN HST_001;

```

```

ALTER TABLE HISTORY1 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY2;
CREATE TABLE HISTORY2

```

```

(
H_C_ID     INTEGER NOT NULL,
H_C_D_ID   SMALLINT NOT NULL,
H_C_W_ID   INTEGER NOT NULL,
H_D_ID     SMALLINT NOT NULL,
H_W_ID     INTEGER NOT NULL,
H_DATE     BIGINT NOT NULL,
H_AMOUNT   INTEGER NOT NULL,
H_DATA     CHAR(24) NOT NULL
)
IN HST_002
INDEX IN HST_002;

```

```

ALTER TABLE HISTORY2 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY3;
CREATE TABLE HISTORY3

```

```

(
H_C_ID     INTEGER NOT NULL,
H_C_D_ID   SMALLINT NOT NULL,
H_C_W_ID   INTEGER NOT NULL,

```

```

H_D_ID     SMALLINT NOT NULL,
H_W_ID     INTEGER NOT NULL,
H_DATE     BIGINT NOT NULL,
H_AMOUNT   INTEGER NOT NULL,
H_DATA     CHAR(24) NOT NULL
)

```

```

IN HST_003
INDEX IN HST_003;
ALTER TABLE HISTORY3 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY4;
CREATE TABLE HISTORY4

```

```

(
H_C_ID     INTEGER NOT NULL,
H_C_D_ID   SMALLINT NOT NULL,
H_C_W_ID   INTEGER NOT NULL,
H_D_ID     SMALLINT NOT NULL,
H_W_ID     INTEGER NOT NULL,
H_DATE     BIGINT NOT NULL,
H_AMOUNT   INTEGER NOT NULL,
H_DATA     CHAR(24) NOT NULL
)
IN HST_004
INDEX IN HST_004;

```

```

ALTER TABLE HISTORY4 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY5;
CREATE TABLE HISTORY5

```

```

(
H_C_ID     INTEGER NOT NULL,
H_C_D_ID   SMALLINT NOT NULL,
H_C_W_ID   INTEGER NOT NULL,
H_D_ID     SMALLINT NOT NULL,
H_W_ID     INTEGER NOT NULL,
H_DATE     BIGINT NOT NULL,
H_AMOUNT   INTEGER NOT NULL,
H_DATA     CHAR(24) NOT NULL
)
IN HST_005
INDEX IN HST_005;

```

```

ALTER TABLE HISTORY5 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY6;
CREATE TABLE HISTORY6

```

```

(
H_C_ID     INTEGER NOT NULL,
H_C_D_ID   SMALLINT NOT NULL,
H_C_W_ID   INTEGER NOT NULL,
H_D_ID     SMALLINT NOT NULL,
H_W_ID     INTEGER NOT NULL,
H_DATE     BIGINT NOT NULL,
H_AMOUNT   INTEGER NOT NULL,
H_DATA     CHAR(24) NOT NULL
)
IN HST_006
INDEX IN HST_006;

```

```

ALTER TABLE HISTORY6 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY7;
CREATE TABLE HISTORY7

```

```

(
H_C_ID     INTEGER NOT NULL,
H_C_D_ID   SMALLINT NOT NULL,
H_C_W_ID   INTEGER NOT NULL,

```



```

H_D_ID    SMALLINT NOT NULL,
H_W_ID    INTEGER  NOT NULL,
H_DATE    BIGINT  NOT NULL,
H_AMOUNT  INTEGER  NOT NULL,
H_DATA    CHAR(24) NOT NULL
)

```

```

IN HST_007
INDEX IN HST_007;

```

```
ALTER TABLE HISTORY7 APPEND ON;
```

```
connect reset;
```

```
connect to TPCC in share mode;
```

```
DROP TABLE HISTORY8;
```

```
CREATE TABLE HISTORY8
```

```

(
H_C_ID    INTEGER  NOT NULL,
H_C_D_ID  SMALLINT NOT NULL,
H_C_W_ID  INTEGER  NOT NULL,
H_D_ID    SMALLINT NOT NULL,
H_W_ID    INTEGER  NOT NULL,
H_DATE    BIGINT  NOT NULL,
H_AMOUNT  INTEGER  NOT NULL,
H_DATA    CHAR(24) NOT NULL
)

```

```

IN HST_008
INDEX IN HST_008;

```

```
ALTER TABLE HISTORY8 APPEND ON;
```

```
connect reset;
```

```
connect to TPCC in share mode;
```

```
DROP TABLE HISTORY9;
```

```
CREATE TABLE HISTORY9
```

```

(
H_C_ID    INTEGER  NOT NULL,
H_C_D_ID  SMALLINT NOT NULL,
H_C_W_ID  INTEGER  NOT NULL,
H_D_ID    SMALLINT NOT NULL,
H_W_ID    INTEGER  NOT NULL,
H_DATE    BIGINT  NOT NULL,
H_AMOUNT  INTEGER  NOT NULL,
H_DATA    CHAR(24) NOT NULL
)

```

```

IN HST_009
INDEX IN HST_009;

```

```
ALTER TABLE HISTORY9 APPEND ON;
```

```
connect reset;
```

```
connect to TPCC in share mode;
```

```
DROP TABLE HISTORY10;
```

```
CREATE TABLE HISTORY10
```

```

(
H_C_ID    INTEGER  NOT NULL,
H_C_D_ID  SMALLINT NOT NULL,
H_C_W_ID  INTEGER  NOT NULL,
H_D_ID    SMALLINT NOT NULL,
H_W_ID    INTEGER  NOT NULL,
H_DATE    BIGINT  NOT NULL,
H_AMOUNT  INTEGER  NOT NULL,
H_DATA    CHAR(24) NOT NULL
)

```

```

IN HST_010
INDEX IN HST_010;

```

```
ALTER TABLE HISTORY10 APPEND ON;
```

```
connect reset;
```

```
connect to TPCC in share mode;
```

```
DROP TABLE HISTORY11;
```

```
CREATE TABLE HISTORY11
```

```

(
H_C_ID    INTEGER  NOT NULL,
H_C_D_ID  SMALLINT NOT NULL,
H_C_W_ID  INTEGER  NOT NULL,

```

```

H_D_ID    SMALLINT NOT NULL,
H_W_ID    INTEGER  NOT NULL,
H_DATE    BIGINT  NOT NULL,
H_AMOUNT  INTEGER  NOT NULL,
H_DATA    CHAR(24) NOT NULL
)

```

```

IN HST_011
INDEX IN HST_011;

```

```
ALTER TABLE HISTORY11 APPEND ON;
```

```
connect reset;
```

```
connect to TPCC in share mode;
```

```
DROP TABLE HISTORY12;
```

```
CREATE TABLE HISTORY12
```

```

(
H_C_ID    INTEGER  NOT NULL,
H_C_D_ID  SMALLINT NOT NULL,
H_C_W_ID  INTEGER  NOT NULL,
H_D_ID    SMALLINT NOT NULL,
H_W_ID    INTEGER  NOT NULL,
H_DATE    BIGINT  NOT NULL,
H_AMOUNT  INTEGER  NOT NULL,
H_DATA    CHAR(24) NOT NULL
)

```

```

IN HST_012
INDEX IN HST_012;

```

```
ALTER TABLE HISTORY12 APPEND ON;
```

```
connect reset;
```

```
connect to TPCC in share mode;
```

```
DROP TABLE HISTORY13;
```

```
CREATE TABLE HISTORY13
```

```

(
H_C_ID    INTEGER  NOT NULL,
H_C_D_ID  SMALLINT NOT NULL,
H_C_W_ID  INTEGER  NOT NULL,
H_D_ID    SMALLINT NOT NULL,
H_W_ID    INTEGER  NOT NULL,
H_DATE    BIGINT  NOT NULL,
H_AMOUNT  INTEGER  NOT NULL,
H_DATA    CHAR(24) NOT NULL
)

```

```

IN HST_013
INDEX IN HST_013;

```

```
ALTER TABLE HISTORY13 APPEND ON;
```

```
connect reset;
```

```
connect to TPCC in share mode;
```

```
DROP TABLE HISTORY14;
```

```
CREATE TABLE HISTORY14
```

```

(
H_C_ID    INTEGER  NOT NULL,
H_C_D_ID  SMALLINT NOT NULL,
H_C_W_ID  INTEGER  NOT NULL,
H_D_ID    SMALLINT NOT NULL,
H_W_ID    INTEGER  NOT NULL,
H_DATE    BIGINT  NOT NULL,
H_AMOUNT  INTEGER  NOT NULL,
H_DATA    CHAR(24) NOT NULL
)

```

```

IN HST_014
INDEX IN HST_014;

```

```
ALTER TABLE HISTORY14 APPEND ON;
```

```
connect reset;
```

```
connect to TPCC in share mode;
```

```
DROP TABLE HISTORY15;
```

```
CREATE TABLE HISTORY15
```

```

(
H_C_ID    INTEGER  NOT NULL,
H_C_D_ID  SMALLINT NOT NULL,
H_C_W_ID  INTEGER  NOT NULL,

```

```

        H_D_ID    SMALLINT  NOT NULL,
        H_W_ID    INTEGER   NOT NULL,
        H_DATE    BIGINT   NOT NULL,
        H_AMOUNT  INTEGER   NOT NULL,
        H_DATA    CHAR(24)  NOT NULL
    )
    IN HST_015
    INDEX IN HST_015;
ALTER TABLE HISTORY15 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY16;
CREATE TABLE HISTORY16

```

```

(
    H_C_ID    INTEGER   NOT NULL,
    H_C_D_ID  SMALLINT  NOT NULL,
    H_C_W_ID  INTEGER   NOT NULL,
    H_D_ID    SMALLINT  NOT NULL,
    H_W_ID    INTEGER   NOT NULL,
    H_DATE    BIGINT   NOT NULL,
    H_AMOUNT  INTEGER   NOT NULL,
    H_DATA    CHAR(24)  NOT NULL
)
    IN HST_016
    INDEX IN HST_016;

```

```

ALTER TABLE HISTORY16 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY17;
CREATE TABLE HISTORY17

```

```

(
    H_C_ID    INTEGER   NOT NULL,
    H_C_D_ID  SMALLINT  NOT NULL,
    H_C_W_ID  INTEGER   NOT NULL,
    H_D_ID    SMALLINT  NOT NULL,
    H_W_ID    INTEGER   NOT NULL,
    H_DATE    BIGINT   NOT NULL,
    H_AMOUNT  INTEGER   NOT NULL,
    H_DATA    CHAR(24)  NOT NULL
)
    IN HST_017
    INDEX IN HST_017;

```

```

ALTER TABLE HISTORY17 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY18;
CREATE TABLE HISTORY18

```

```

(
    H_C_ID    INTEGER   NOT NULL,
    H_C_D_ID  SMALLINT  NOT NULL,
    H_C_W_ID  INTEGER   NOT NULL,
    H_D_ID    SMALLINT  NOT NULL,
    H_W_ID    INTEGER   NOT NULL,
    H_DATE    BIGINT   NOT NULL,
    H_AMOUNT  INTEGER   NOT NULL,
    H_DATA    CHAR(24)  NOT NULL
)
    IN HST_018
    INDEX IN HST_018;

```

```

ALTER TABLE HISTORY18 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY19;
CREATE TABLE HISTORY19

```

```

(
    H_C_ID    INTEGER   NOT NULL,
    H_C_D_ID  SMALLINT  NOT NULL,
    H_C_W_ID  INTEGER   NOT NULL,

```

```

        H_D_ID    SMALLINT  NOT NULL,
        H_W_ID    INTEGER   NOT NULL,
        H_DATE    BIGINT   NOT NULL,
        H_AMOUNT  INTEGER   NOT NULL,
        H_DATA    CHAR(24)  NOT NULL
    )
    IN HST_019
    INDEX IN HST_019;
ALTER TABLE HISTORY19 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY20;
CREATE TABLE HISTORY20

```

```

(
    H_C_ID    INTEGER   NOT NULL,
    H_C_D_ID  SMALLINT  NOT NULL,
    H_C_W_ID  INTEGER   NOT NULL,
    H_D_ID    SMALLINT  NOT NULL,
    H_W_ID    INTEGER   NOT NULL,
    H_DATE    BIGINT   NOT NULL,
    H_AMOUNT  INTEGER   NOT NULL,
    H_DATA    CHAR(24)  NOT NULL
)
    IN HST_020
    INDEX IN HST_020;

```

```

ALTER TABLE HISTORY20 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY21;
CREATE TABLE HISTORY21

```

```

(
    H_C_ID    INTEGER   NOT NULL,
    H_C_D_ID  SMALLINT  NOT NULL,
    H_C_W_ID  INTEGER   NOT NULL,
    H_D_ID    SMALLINT  NOT NULL,
    H_W_ID    INTEGER   NOT NULL,
    H_DATE    BIGINT   NOT NULL,
    H_AMOUNT  INTEGER   NOT NULL,
    H_DATA    CHAR(24)  NOT NULL
)
    IN HST_021
    INDEX IN HST_021;

```

```

ALTER TABLE HISTORY21 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY22;
CREATE TABLE HISTORY22

```

```

(
    H_C_ID    INTEGER   NOT NULL,
    H_C_D_ID  SMALLINT  NOT NULL,
    H_C_W_ID  INTEGER   NOT NULL,
    H_D_ID    SMALLINT  NOT NULL,
    H_W_ID    INTEGER   NOT NULL,
    H_DATE    BIGINT   NOT NULL,
    H_AMOUNT  INTEGER   NOT NULL,
    H_DATA    CHAR(24)  NOT NULL
)
    IN HST_022
    INDEX IN HST_022;

```

```

ALTER TABLE HISTORY22 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY23;
CREATE TABLE HISTORY23

```

```

(
    H_C_ID    INTEGER   NOT NULL,
    H_C_D_ID  SMALLINT  NOT NULL,
    H_C_W_ID  INTEGER   NOT NULL,

```

```

H_D_ID    SMALLINT  NOT NULL,
H_W_ID    INTEGER   NOT NULL,
H_DATE    BIGINT    NOT NULL,
H_AMOUNT  INTEGER   NOT NULL,
H_DATA    CHAR(24)  NOT NULL
)
IN HST_023
INDEX IN HST_023;
ALTER TABLE HISTORY23 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY24;
CREATE TABLE HISTORY24
(
H_C_ID    INTEGER   NOT NULL,
H_C_D_ID  SMALLINT  NOT NULL,
H_C_W_ID  INTEGER   NOT NULL,
H_D_ID    SMALLINT  NOT NULL,
H_W_ID    INTEGER   NOT NULL,
H_DATE    BIGINT    NOT NULL,
H_AMOUNT  INTEGER   NOT NULL,
H_DATA    CHAR(24)  NOT NULL
)
IN HST_024
INDEX IN HST_024;
ALTER TABLE HISTORY24 APPEND ON;
connect reset;
connect to TPCC in share mode;
DROP TABLE HISTORY25;
CREATE TABLE HISTORY25
(
H_C_ID    INTEGER   NOT NULL,
H_C_D_ID  SMALLINT  NOT NULL,
H_C_W_ID  INTEGER   NOT NULL,
H_D_ID    SMALLINT  NOT NULL,
H_W_ID    INTEGER   NOT NULL,
H_DATE    BIGINT    NOT NULL,
H_AMOUNT  INTEGER   NOT NULL,
H_DATA    CHAR(24)  NOT NULL
)
IN HST_025
INDEX IN HST_025;
ALTER TABLE HISTORY25 APPEND ON;
connect reset;

```

crtb_warehouse.ddl

```

connect to TPCC in share mode;
DROP TABLE WAREHOUSE1;
CREATE TABLE WAREHOUSE1
(
W_NAME    CHAR(10)  NOT NULL,
W_STREET_1 CHAR(20) NOT NULL,
W_STREET_2 CHAR(20) NOT NULL,
W_CITY    CHAR(20)  NOT NULL,
W_STATE   CHAR(2)   NOT NULL,
W_ZIP     CHAR(9)   NOT NULL,
W_TAX     INTEGER   NOT NULL,
W_YTD     BIGINT    NOT NULL,
W_ID      INTEGER   NOT NULL
)
IN WAR_001
INDEX IN WAR_001
ORGANIZE BY KEY SEQUENCE (
W_ID STARTING FROM 1 ENDING AT 800
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE2;
CREATE TABLE WAREHOUSE2
(
W_NAME    CHAR(10)  NOT NULL,
W_STREET_1 CHAR(20) NOT NULL,
W_STREET_2 CHAR(20) NOT NULL,
W_CITY    CHAR(20)  NOT NULL,
W_STATE   CHAR(2)   NOT NULL,
W_ZIP     CHAR(9)   NOT NULL,
W_TAX     INTEGER   NOT NULL,
W_YTD     BIGINT    NOT NULL,
W_ID      INTEGER   NOT NULL
)
IN WAR_002
INDEX IN WAR_002
ORGANIZE BY KEY SEQUENCE (
W_ID STARTING FROM 801 ENDING AT 1600
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE3;
CREATE TABLE WAREHOUSE3
(
W_NAME    CHAR(10)  NOT NULL,
W_STREET_1 CHAR(20) NOT NULL,
W_STREET_2 CHAR(20) NOT NULL,
W_CITY    CHAR(20)  NOT NULL,
W_STATE   CHAR(2)   NOT NULL,
W_ZIP     CHAR(9)   NOT NULL,
W_TAX     INTEGER   NOT NULL,
W_YTD     BIGINT    NOT NULL,
W_ID      INTEGER   NOT NULL
)
IN WAR_003
INDEX IN WAR_003
ORGANIZE BY KEY SEQUENCE (
W_ID STARTING FROM 1601 ENDING AT 2400
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE4;
CREATE TABLE WAREHOUSE4
(
W_NAME    CHAR(10)  NOT NULL,
W_STREET_1 CHAR(20) NOT NULL,
W_STREET_2 CHAR(20) NOT NULL,
W_CITY    CHAR(20)  NOT NULL,
W_STATE   CHAR(2)   NOT NULL,
W_ZIP     CHAR(9)   NOT NULL,
W_TAX     INTEGER   NOT NULL,
W_YTD     BIGINT    NOT NULL,
W_ID      INTEGER   NOT NULL
)
IN WAR_004
INDEX IN WAR_004
ORGANIZE BY KEY SEQUENCE (
W_ID STARTING FROM 2401 ENDING AT 3200
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE5;
CREATE TABLE WAREHOUSE5
(

```

```

W_NAME CHAR(10) NOT NULL,
W_STREET_1 CHAR(20) NOT NULL,
W_STREET_2 CHAR(20) NOT NULL,
W_CITY CHAR(20) NOT NULL,
W_STATE CHAR(2) NOT NULL,
W_ZIP CHAR(9) NOT NULL,
W_TAX INTEGER NOT NULL,
W_YTD BIGINT NOT NULL,
W_ID INTEGER NOT NULL
)
IN WAR_005
INDEX IN WAR_005
ORGANIZE BY KEY SEQUENCE (
W_ID STARTING FROM 3201 ENDING AT 4000
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE6;
CREATE TABLE WAREHOUSE6
(
W_NAME CHAR(10) NOT NULL,
W_STREET_1 CHAR(20) NOT NULL,
W_STREET_2 CHAR(20) NOT NULL,
W_CITY CHAR(20) NOT NULL,
W_STATE CHAR(2) NOT NULL,
W_ZIP CHAR(9) NOT NULL,
W_TAX INTEGER NOT NULL,
W_YTD BIGINT NOT NULL,
W_ID INTEGER NOT NULL
)
IN WAR_006
INDEX IN WAR_006
ORGANIZE BY KEY SEQUENCE (
W_ID STARTING FROM 4001 ENDING AT 4800
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE7;
CREATE TABLE WAREHOUSE7
(
W_NAME CHAR(10) NOT NULL,
W_STREET_1 CHAR(20) NOT NULL,
W_STREET_2 CHAR(20) NOT NULL,
W_CITY CHAR(20) NOT NULL,
W_STATE CHAR(2) NOT NULL,
W_ZIP CHAR(9) NOT NULL,
W_TAX INTEGER NOT NULL,
W_YTD BIGINT NOT NULL,
W_ID INTEGER NOT NULL
)
IN WAR_007
INDEX IN WAR_007
ORGANIZE BY KEY SEQUENCE (
W_ID STARTING FROM 4801 ENDING AT 5600
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE8;
CREATE TABLE WAREHOUSE8
(
W_NAME CHAR(10) NOT NULL,
W_STREET_1 CHAR(20) NOT NULL,
W_STREET_2 CHAR(20) NOT NULL,
W_CITY CHAR(20) NOT NULL,
W_STATE CHAR(2) NOT NULL,

```

```

W_ZIP CHAR(9) NOT NULL,
W_TAX INTEGER NOT NULL,
W_YTD BIGINT NOT NULL,
W_ID INTEGER NOT NULL
)
IN WAR_008
INDEX IN WAR_008
ORGANIZE BY KEY SEQUENCE (
W_ID STARTING FROM 5601 ENDING AT 6400
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE9;
CREATE TABLE WAREHOUSE9
(
W_NAME CHAR(10) NOT NULL,
W_STREET_1 CHAR(20) NOT NULL,
W_STREET_2 CHAR(20) NOT NULL,
W_CITY CHAR(20) NOT NULL,
W_STATE CHAR(2) NOT NULL,
W_ZIP CHAR(9) NOT NULL,
W_TAX INTEGER NOT NULL,
W_YTD BIGINT NOT NULL,
W_ID INTEGER NOT NULL
)
IN WAR_009
INDEX IN WAR_009
ORGANIZE BY KEY SEQUENCE (
W_ID STARTING FROM 6401 ENDING AT 7200
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE10;
CREATE TABLE WAREHOUSE10
(
W_NAME CHAR(10) NOT NULL,
W_STREET_1 CHAR(20) NOT NULL,
W_STREET_2 CHAR(20) NOT NULL,
W_CITY CHAR(20) NOT NULL,
W_STATE CHAR(2) NOT NULL,
W_ZIP CHAR(9) NOT NULL,
W_TAX INTEGER NOT NULL,
W_YTD BIGINT NOT NULL,
W_ID INTEGER NOT NULL
)
IN WAR_010
INDEX IN WAR_010
ORGANIZE BY KEY SEQUENCE (
W_ID STARTING FROM 7201 ENDING AT 8000
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE11;
CREATE TABLE WAREHOUSE11
(
W_NAME CHAR(10) NOT NULL,
W_STREET_1 CHAR(20) NOT NULL,
W_STREET_2 CHAR(20) NOT NULL,
W_CITY CHAR(20) NOT NULL,
W_STATE CHAR(2) NOT NULL,
W_ZIP CHAR(9) NOT NULL,
W_TAX INTEGER NOT NULL,
W_YTD BIGINT NOT NULL,
W_ID INTEGER NOT NULL
)

```

```

IN WAR_011
INDEX IN WAR_011
ORGANIZE BY KEY SEQUENCE (
  W_ID STARTING FROM 8001 ENDING AT 8800
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE12;
CREATE TABLE WAREHOUSE12
(
  W_NAME CHAR(10) NOT NULL,
  W_STREET_1 CHAR(20) NOT NULL,
  W_STREET_2 CHAR(20) NOT NULL,
  W_CITY CHAR(20) NOT NULL,
  W_STATE CHAR(2) NOT NULL,
  W_ZIP CHAR(9) NOT NULL,
  W_TAX INTEGER NOT NULL,
  W_YTD BIGINT NOT NULL,
  W_ID INTEGER NOT NULL
)
IN WAR_012
INDEX IN WAR_012
ORGANIZE BY KEY SEQUENCE (
  W_ID STARTING FROM 8801 ENDING AT 9600
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE13;
CREATE TABLE WAREHOUSE13
(
  W_NAME CHAR(10) NOT NULL,
  W_STREET_1 CHAR(20) NOT NULL,
  W_STREET_2 CHAR(20) NOT NULL,
  W_CITY CHAR(20) NOT NULL,
  W_STATE CHAR(2) NOT NULL,
  W_ZIP CHAR(9) NOT NULL,
  W_TAX INTEGER NOT NULL,
  W_YTD BIGINT NOT NULL,
  W_ID INTEGER NOT NULL
)
IN WAR_013
INDEX IN WAR_013
ORGANIZE BY KEY SEQUENCE (
  W_ID STARTING FROM 9601 ENDING AT 10400
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE14;
CREATE TABLE WAREHOUSE14
(
  W_NAME CHAR(10) NOT NULL,
  W_STREET_1 CHAR(20) NOT NULL,
  W_STREET_2 CHAR(20) NOT NULL,
  W_CITY CHAR(20) NOT NULL,
  W_STATE CHAR(2) NOT NULL,
  W_ZIP CHAR(9) NOT NULL,
  W_TAX INTEGER NOT NULL,
  W_YTD BIGINT NOT NULL,
  W_ID INTEGER NOT NULL
)
IN WAR_014
INDEX IN WAR_014
ORGANIZE BY KEY SEQUENCE (
  W_ID STARTING FROM 10401 ENDING AT 11200
)

```

```

ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE15;
CREATE TABLE WAREHOUSE15
(
  W_NAME CHAR(10) NOT NULL,
  W_STREET_1 CHAR(20) NOT NULL,
  W_STREET_2 CHAR(20) NOT NULL,
  W_CITY CHAR(20) NOT NULL,
  W_STATE CHAR(2) NOT NULL,
  W_ZIP CHAR(9) NOT NULL,
  W_TAX INTEGER NOT NULL,
  W_YTD BIGINT NOT NULL,
  W_ID INTEGER NOT NULL
)
IN WAR_015
INDEX IN WAR_015
ORGANIZE BY KEY SEQUENCE (
  W_ID STARTING FROM 11201 ENDING AT 12000
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE16;
CREATE TABLE WAREHOUSE16
(
  W_NAME CHAR(10) NOT NULL,
  W_STREET_1 CHAR(20) NOT NULL,
  W_STREET_2 CHAR(20) NOT NULL,
  W_CITY CHAR(20) NOT NULL,
  W_STATE CHAR(2) NOT NULL,
  W_ZIP CHAR(9) NOT NULL,
  W_TAX INTEGER NOT NULL,
  W_YTD BIGINT NOT NULL,
  W_ID INTEGER NOT NULL
)
IN WAR_016
INDEX IN WAR_016
ORGANIZE BY KEY SEQUENCE (
  W_ID STARTING FROM 12001 ENDING AT 12800
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE17;
CREATE TABLE WAREHOUSE17
(
  W_NAME CHAR(10) NOT NULL,
  W_STREET_1 CHAR(20) NOT NULL,
  W_STREET_2 CHAR(20) NOT NULL,
  W_CITY CHAR(20) NOT NULL,
  W_STATE CHAR(2) NOT NULL,
  W_ZIP CHAR(9) NOT NULL,
  W_TAX INTEGER NOT NULL,
  W_YTD BIGINT NOT NULL,
  W_ID INTEGER NOT NULL
)
IN WAR_017
INDEX IN WAR_017
ORGANIZE BY KEY SEQUENCE (
  W_ID STARTING FROM 12801 ENDING AT 13600
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE18;
CREATE TABLE WAREHOUSE18

```

```

(
W_NAME CHAR(10) NOT NULL,
W_STREET_1 CHAR(20) NOT NULL,
W_STREET_2 CHAR(20) NOT NULL,
W_CITY CHAR(20) NOT NULL,
W_STATE CHAR(2) NOT NULL,
W_ZIP CHAR(9) NOT NULL,
W_TAX INTEGER NOT NULL,
W_YTD BIGINT NOT NULL,
W_ID INTEGER NOT NULL
)
IN WAR_018
INDEX IN WAR_018
ORGANIZE BY KEY SEQUENCE (
W_ID STARTING FROM 13601 ENDING AT 14400
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE19;
CREATE TABLE WAREHOUSE19

```

```

(
W_NAME CHAR(10) NOT NULL,
W_STREET_1 CHAR(20) NOT NULL,
W_STREET_2 CHAR(20) NOT NULL,
W_CITY CHAR(20) NOT NULL,
W_STATE CHAR(2) NOT NULL,
W_ZIP CHAR(9) NOT NULL,
W_TAX INTEGER NOT NULL,
W_YTD BIGINT NOT NULL,
W_ID INTEGER NOT NULL
)
IN WAR_019
INDEX IN WAR_019
ORGANIZE BY KEY SEQUENCE (
W_ID STARTING FROM 14401 ENDING AT 15200
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE20;
CREATE TABLE WAREHOUSE20

```

```

(
W_NAME CHAR(10) NOT NULL,
W_STREET_1 CHAR(20) NOT NULL,
W_STREET_2 CHAR(20) NOT NULL,
W_CITY CHAR(20) NOT NULL,
W_STATE CHAR(2) NOT NULL,
W_ZIP CHAR(9) NOT NULL,
W_TAX INTEGER NOT NULL,
W_YTD BIGINT NOT NULL,
W_ID INTEGER NOT NULL
)
IN WAR_020
INDEX IN WAR_020
ORGANIZE BY KEY SEQUENCE (
W_ID STARTING FROM 15201 ENDING AT 16000
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE21;
CREATE TABLE WAREHOUSE21

```

```

(
W_NAME CHAR(10) NOT NULL,
W_STREET_1 CHAR(20) NOT NULL,
W_STREET_2 CHAR(20) NOT NULL,
W_CITY CHAR(20) NOT NULL,

```

```

W_STATE CHAR(2) NOT NULL,
W_ZIP CHAR(9) NOT NULL,
W_TAX INTEGER NOT NULL,
W_YTD BIGINT NOT NULL,
W_ID INTEGER NOT NULL
)
IN WAR_021
INDEX IN WAR_021
ORGANIZE BY KEY SEQUENCE (
W_ID STARTING FROM 16001 ENDING AT 16800
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE22;
CREATE TABLE WAREHOUSE22

```

```

(
W_NAME CHAR(10) NOT NULL,
W_STREET_1 CHAR(20) NOT NULL,
W_STREET_2 CHAR(20) NOT NULL,
W_CITY CHAR(20) NOT NULL,
W_STATE CHAR(2) NOT NULL,
W_ZIP CHAR(9) NOT NULL,
W_TAX INTEGER NOT NULL,
W_YTD BIGINT NOT NULL,
W_ID INTEGER NOT NULL
)
IN WAR_022
INDEX IN WAR_022
ORGANIZE BY KEY SEQUENCE (
W_ID STARTING FROM 16801 ENDING AT 17600
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE23;
CREATE TABLE WAREHOUSE23

```

```

(
W_NAME CHAR(10) NOT NULL,
W_STREET_1 CHAR(20) NOT NULL,
W_STREET_2 CHAR(20) NOT NULL,
W_CITY CHAR(20) NOT NULL,
W_STATE CHAR(2) NOT NULL,
W_ZIP CHAR(9) NOT NULL,
W_TAX INTEGER NOT NULL,
W_YTD BIGINT NOT NULL,
W_ID INTEGER NOT NULL
)
IN WAR_023
INDEX IN WAR_023
ORGANIZE BY KEY SEQUENCE (
W_ID STARTING FROM 17601 ENDING AT 18400
)
ALLOW OVERFLOW;

```

```

connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE24;
CREATE TABLE WAREHOUSE24

```

```

(
W_NAME CHAR(10) NOT NULL,
W_STREET_1 CHAR(20) NOT NULL,
W_STREET_2 CHAR(20) NOT NULL,
W_CITY CHAR(20) NOT NULL,
W_STATE CHAR(2) NOT NULL,
W_ZIP CHAR(9) NOT NULL,
W_TAX INTEGER NOT NULL,
W_YTD BIGINT NOT NULL,
W_ID INTEGER NOT NULL
)

```

```

)
IN WAR_024
INDEX IN WAR_024
ORGANIZE BY KEY SEQUENCE (
W_ID STARTING FROM 18401 ENDING AT 19200
)
ALLOW OVERFLOW;
connect reset;
connect to TPCC in share mode;
DROP TABLE WAREHOUSE25;
CREATE TABLE WAREHOUSE25
(
W_NAME CHAR(10) NOT NULL,
W_STREET_1 CHAR(20) NOT NULL,
W_STREET_2 CHAR(20) NOT NULL,
W_CITY CHAR(20) NOT NULL,
W_STATE CHAR(2) NOT NULL,
W_ZIP CHAR(9) NOT NULL,
W_TAX INTEGER NOT NULL,
W_YTD BIGINT NOT NULL,
W_ID INTEGER NOT NULL
)
IN WAR_025
INDEX IN WAR_025
ORGANIZE BY KEY SEQUENCE (
W_ID STARTING FROM 19201 ENDING AT 20000
)
ALLOW OVERFLOW;
connect reset;

```

crvw_customer.ddl

```

connect to TPCC in share mode;
DROP VIEW CUSTOMER;
CREATE VIEW CUSTOMER
(C_ID,
C_STATE,
C_ZIP,
C_PHONE,
C_SINCE,
C_CREDIT_LIM,
C_MIDDLE,
C_CREDIT,
C_DISCOUNT,
C_DATA,
C_LAST,
C_FIRST,
C_STREET_1,
C_STREET_2,
C_CITY,
C_D_ID,
C_W_ID,
C_DELIVERY_CNT,
C_BALANCE,
C_YTD_PAYMENT,
C_PAYMENT_CNT
) AS SELECT * FROM CUSTOMER1 UNION ALL
SELECT * FROM CUSTOMER2 UNION ALL
SELECT * FROM CUSTOMER3 UNION ALL
SELECT * FROM CUSTOMER4 UNION ALL
SELECT * FROM CUSTOMER5 UNION ALL
SELECT * FROM CUSTOMER6 UNION ALL
SELECT * FROM CUSTOMER7 UNION ALL
SELECT * FROM CUSTOMER8 UNION ALL
SELECT * FROM CUSTOMER9 UNION ALL
SELECT * FROM CUSTOMER10 UNION ALL
SELECT * FROM CUSTOMER11 UNION ALL

```

```

SELECT * FROM CUSTOMER12 UNION ALL
SELECT * FROM CUSTOMER13 UNION ALL
SELECT * FROM CUSTOMER14 UNION ALL
SELECT * FROM CUSTOMER15 UNION ALL
SELECT * FROM CUSTOMER16 UNION ALL
SELECT * FROM CUSTOMER17 UNION ALL
SELECT * FROM CUSTOMER18 UNION ALL
SELECT * FROM CUSTOMER19 UNION ALL
SELECT * FROM CUSTOMER20 UNION ALL
SELECT * FROM CUSTOMER21 UNION ALL
SELECT * FROM CUSTOMER22 UNION ALL
SELECT * FROM CUSTOMER23 UNION ALL
SELECT * FROM CUSTOMER24 UNION ALL
SELECT * FROM CUSTOMER25
WITH ROW MOVEMENT;
COMMIT WORK;
connect reset;

```

crvw_district.ddl

```

connect to TPCC in share mode;
DROP VIEW DISTRICT;
CREATE VIEW DISTRICT
(D_NEXT_O_ID,
D_TAX,
D_YTD,
D_NAME,
D_STREET_1,
D_STREET_2,
D_CITY,
D_STATE,
D_ZIP,
D_ID,
D_W_ID
) AS SELECT * FROM DISTRICT1 UNION ALL
SELECT * FROM DISTRICT2 UNION ALL
SELECT * FROM DISTRICT3 UNION ALL
SELECT * FROM DISTRICT4 UNION ALL
SELECT * FROM DISTRICT5 UNION ALL
SELECT * FROM DISTRICT6 UNION ALL
SELECT * FROM DISTRICT7 UNION ALL
SELECT * FROM DISTRICT8 UNION ALL
SELECT * FROM DISTRICT9 UNION ALL
SELECT * FROM DISTRICT10 UNION ALL
SELECT * FROM DISTRICT11 UNION ALL
SELECT * FROM DISTRICT12 UNION ALL
SELECT * FROM DISTRICT13 UNION ALL
SELECT * FROM DISTRICT14 UNION ALL
SELECT * FROM DISTRICT15 UNION ALL
SELECT * FROM DISTRICT16 UNION ALL
SELECT * FROM DISTRICT17 UNION ALL
SELECT * FROM DISTRICT18 UNION ALL
SELECT * FROM DISTRICT19 UNION ALL
SELECT * FROM DISTRICT20 UNION ALL
SELECT * FROM DISTRICT21 UNION ALL
SELECT * FROM DISTRICT22 UNION ALL
SELECT * FROM DISTRICT23 UNION ALL
SELECT * FROM DISTRICT24 UNION ALL
SELECT * FROM DISTRICT25
WITH ROW MOVEMENT;
COMMIT WORK;
connect reset;

```

crvw_history.ddl

```

connect to TPCC in share mode;
DROP VIEW HISTORY;
CREATE VIEW HISTORY
  (H_C_ID,
   H_C_D_ID,
   H_C_W_ID,
   H_D_ID,
   H_W_ID,
   H_DATE,
   H_AMOUNT,
   H_DATA
  ) AS SELECT * FROM HISTORY1 UNION ALL
SELECT * FROM HISTORY2 UNION ALL
SELECT * FROM HISTORY3 UNION ALL
SELECT * FROM HISTORY4 UNION ALL
SELECT * FROM HISTORY5 UNION ALL
SELECT * FROM HISTORY6 UNION ALL
SELECT * FROM HISTORY7 UNION ALL
SELECT * FROM HISTORY8 UNION ALL
SELECT * FROM HISTORY9 UNION ALL
SELECT * FROM HISTORY10 UNION ALL
SELECT * FROM HISTORY11 UNION ALL
SELECT * FROM HISTORY12 UNION ALL
SELECT * FROM HISTORY13 UNION ALL
SELECT * FROM HISTORY14 UNION ALL
SELECT * FROM HISTORY15 UNION ALL
SELECT * FROM HISTORY16 UNION ALL
SELECT * FROM HISTORY17 UNION ALL
SELECT * FROM HISTORY18 UNION ALL
SELECT * FROM HISTORY19 UNION ALL
SELECT * FROM HISTORY20 UNION ALL
SELECT * FROM HISTORY21 UNION ALL
SELECT * FROM HISTORY22 UNION ALL
SELECT * FROM HISTORY23 UNION ALL
SELECT * FROM HISTORY24 UNION ALL
SELECT * FROM HISTORY25
WITH ROW MOVEMENT;
COMMIT WORK;
connect reset;

```

crvw_new_order.ddl

```

connect to TPCC in share mode;
DROP VIEW NEW_ORDER;
CREATE VIEW NEW_ORDER
  (NO_O_ID,
   NO_D_ID,
   NO_W_ID
  ) AS SELECT * FROM NEW_ORDERA1 UNION ALL
SELECT * FROM NEW_ORDERA2 UNION ALL
SELECT * FROM NEW_ORDERA3 UNION ALL
SELECT * FROM NEW_ORDERA4 UNION ALL
SELECT * FROM NEW_ORDERA5 UNION ALL
SELECT * FROM NEW_ORDERA6 UNION ALL
SELECT * FROM NEW_ORDERA7 UNION ALL
SELECT * FROM NEW_ORDERA8 UNION ALL
SELECT * FROM NEW_ORDERA9 UNION ALL
SELECT * FROM NEW_ORDERA10 UNION ALL
SELECT * FROM NEW_ORDERA11 UNION ALL
SELECT * FROM NEW_ORDERA12 UNION ALL
SELECT * FROM NEW_ORDERA13 UNION ALL
SELECT * FROM NEW_ORDERA14 UNION ALL
SELECT * FROM NEW_ORDERA15 UNION ALL
SELECT * FROM NEW_ORDERA16 UNION ALL
SELECT * FROM NEW_ORDERA17 UNION ALL
SELECT * FROM NEW_ORDERA18 UNION ALL
SELECT * FROM NEW_ORDERA19 UNION ALL

```

```

SELECT * FROM NEW_ORDERA20 UNION ALL
SELECT * FROM NEW_ORDERA21 UNION ALL
SELECT * FROM NEW_ORDERA22 UNION ALL
SELECT * FROM NEW_ORDERA23 UNION ALL
SELECT * FROM NEW_ORDERA24 UNION ALL
SELECT * FROM NEW_ORDERA25 UNION ALL
SELECT * FROM NEW_ORDERB1 UNION ALL
SELECT * FROM NEW_ORDERB2 UNION ALL
SELECT * FROM NEW_ORDERB3 UNION ALL
SELECT * FROM NEW_ORDERB4 UNION ALL
SELECT * FROM NEW_ORDERB5 UNION ALL
SELECT * FROM NEW_ORDERB6 UNION ALL
SELECT * FROM NEW_ORDERB7 UNION ALL
SELECT * FROM NEW_ORDERB8 UNION ALL
SELECT * FROM NEW_ORDERB9 UNION ALL
SELECT * FROM NEW_ORDERB10 UNION ALL
SELECT * FROM NEW_ORDERB11 UNION ALL
SELECT * FROM NEW_ORDERB12 UNION ALL
SELECT * FROM NEW_ORDERB13 UNION ALL
SELECT * FROM NEW_ORDERB14 UNION ALL
SELECT * FROM NEW_ORDERB15 UNION ALL
SELECT * FROM NEW_ORDERB16 UNION ALL
SELECT * FROM NEW_ORDERB17 UNION ALL
SELECT * FROM NEW_ORDERB18 UNION ALL
SELECT * FROM NEW_ORDERB19 UNION ALL
SELECT * FROM NEW_ORDERB20 UNION ALL
SELECT * FROM NEW_ORDERB21 UNION ALL
SELECT * FROM NEW_ORDERB22 UNION ALL
SELECT * FROM NEW_ORDERB23 UNION ALL
SELECT * FROM NEW_ORDERB24 UNION ALL
SELECT * FROM NEW_ORDERB25
WITH ROW MOVEMENT;
COMMIT WORK;
connect reset;

```

crvw_order_line.ddl

```

connect to TPCC in share mode;
DROP VIEW ORDER_LINE;
CREATE VIEW ORDER_LINE
  (OL_DELIVERY_D,
   OL_AMOUNT,
   OL_I_ID,
   OL_SUPPLY_W_ID,
   OL_QUANTITY,
   OL_DIST_INFO,
   OL_O_ID,
   OL_D_ID,
   OL_W_ID,
   OL_NUMBER
  ) AS SELECT * FROM ORDER_LINE1 UNION ALL
SELECT * FROM ORDER_LINE2 UNION ALL
SELECT * FROM ORDER_LINE3 UNION ALL
SELECT * FROM ORDER_LINE4 UNION ALL
SELECT * FROM ORDER_LINE5 UNION ALL
SELECT * FROM ORDER_LINE6 UNION ALL
SELECT * FROM ORDER_LINE7 UNION ALL
SELECT * FROM ORDER_LINE8 UNION ALL
SELECT * FROM ORDER_LINE9 UNION ALL
SELECT * FROM ORDER_LINE10 UNION ALL
SELECT * FROM ORDER_LINE11 UNION ALL
SELECT * FROM ORDER_LINE12 UNION ALL
SELECT * FROM ORDER_LINE13 UNION ALL
SELECT * FROM ORDER_LINE14 UNION ALL
SELECT * FROM ORDER_LINE15 UNION ALL
SELECT * FROM ORDER_LINE16 UNION ALL
SELECT * FROM ORDER_LINE17 UNION ALL

```



```

SELECT * FROM ORDER_LINE18 UNION ALL
SELECT * FROM ORDER_LINE19 UNION ALL
SELECT * FROM ORDER_LINE20 UNION ALL
SELECT * FROM ORDER_LINE21 UNION ALL
SELECT * FROM ORDER_LINE22 UNION ALL
SELECT * FROM ORDER_LINE23 UNION ALL
SELECT * FROM ORDER_LINE24 UNION ALL
SELECT * FROM ORDER_LINE25
WITH ROW MOVEMENT;
COMMIT WORK;
connect reset;

```

crvw_orders.ddl

```

connect to TPCC in share mode;
DROP VIEW ORDERS;
CREATE VIEW ORDERS
(O_C_ID,
O_ENTRY_D,
O_CARRIER_ID,
O_OL_CNT,
O_ALL_LOCAL,
O_ID,
O_W_ID,
O_D_ID
) AS SELECT * FROM ORDERS1 UNION ALL
SELECT * FROM ORDERS2 UNION ALL
SELECT * FROM ORDERS3 UNION ALL
SELECT * FROM ORDERS4 UNION ALL
SELECT * FROM ORDERS5 UNION ALL
SELECT * FROM ORDERS6 UNION ALL
SELECT * FROM ORDERS7 UNION ALL
SELECT * FROM ORDERS8 UNION ALL
SELECT * FROM ORDERS9 UNION ALL
SELECT * FROM ORDERS10 UNION ALL
SELECT * FROM ORDERS11 UNION ALL
SELECT * FROM ORDERS12 UNION ALL
SELECT * FROM ORDERS13 UNION ALL
SELECT * FROM ORDERS14 UNION ALL
SELECT * FROM ORDERS15 UNION ALL
SELECT * FROM ORDERS16 UNION ALL
SELECT * FROM ORDERS17 UNION ALL
SELECT * FROM ORDERS18 UNION ALL
SELECT * FROM ORDERS19 UNION ALL
SELECT * FROM ORDERS20 UNION ALL
SELECT * FROM ORDERS21 UNION ALL
SELECT * FROM ORDERS22 UNION ALL
SELECT * FROM ORDERS23 UNION ALL
SELECT * FROM ORDERS24 UNION ALL
SELECT * FROM ORDERS25
WITH ROW MOVEMENT;
COMMIT WORK;
connect reset;

```

crvw_stock.ddl

```

connect to TPCC in share mode;
DROP VIEW STOCK;
CREATE VIEW STOCK
(S_REMOTE_CNT,
S_QUANTITY,
S_ORDER_CNT,
S_YTD,
S_DATA,
S_DIST_01,

```

```

S_DIST_02,
S_DIST_03,
S_DIST_04,
S_DIST_05,
S_DIST_06,
S_DIST_07,
S_DIST_08,
S_DIST_09,
S_DIST_10,
S_I_ID,
S_W_ID
) AS SELECT * FROM STOCK1 UNION ALL

```

```

SELECT * FROM STOCK2 UNION ALL
SELECT * FROM STOCK3 UNION ALL
SELECT * FROM STOCK4 UNION ALL
SELECT * FROM STOCK5 UNION ALL
SELECT * FROM STOCK6 UNION ALL
SELECT * FROM STOCK7 UNION ALL
SELECT * FROM STOCK8 UNION ALL
SELECT * FROM STOCK9 UNION ALL
SELECT * FROM STOCK10 UNION ALL
SELECT * FROM STOCK11 UNION ALL
SELECT * FROM STOCK12 UNION ALL
SELECT * FROM STOCK13 UNION ALL
SELECT * FROM STOCK14 UNION ALL
SELECT * FROM STOCK15 UNION ALL
SELECT * FROM STOCK16 UNION ALL
SELECT * FROM STOCK17 UNION ALL
SELECT * FROM STOCK18 UNION ALL
SELECT * FROM STOCK19 UNION ALL
SELECT * FROM STOCK20 UNION ALL
SELECT * FROM STOCK21 UNION ALL
SELECT * FROM STOCK22 UNION ALL
SELECT * FROM STOCK23 UNION ALL
SELECT * FROM STOCK24 UNION ALL
SELECT * FROM STOCK25
WITH ROW MOVEMENT;
COMMIT WORK;
connect reset;

```

crvw_warehouse.ddl

```

connect to TPCC in share mode;
DROP VIEW WAREHOUSE;
CREATE VIEW WAREHOUSE
(W_NAME,
W_STREET_1,
W_STREET_2,
W_CITY,
W_STATE,
W_ZIP,
W_TAX,
W_YTD,
W_ID
) AS SELECT * FROM WAREHOUSE1 UNION ALL
SELECT * FROM WAREHOUSE2 UNION ALL
SELECT * FROM WAREHOUSE3 UNION ALL
SELECT * FROM WAREHOUSE4 UNION ALL
SELECT * FROM WAREHOUSE5 UNION ALL
SELECT * FROM WAREHOUSE6 UNION ALL
SELECT * FROM WAREHOUSE7 UNION ALL
SELECT * FROM WAREHOUSE8 UNION ALL
SELECT * FROM WAREHOUSE9 UNION ALL
SELECT * FROM WAREHOUSE10 UNION ALL
SELECT * FROM WAREHOUSE11 UNION ALL
SELECT * FROM WAREHOUSE12 UNION ALL
SELECT * FROM WAREHOUSE13 UNION ALL

```

```

SELECT * FROM WAREHOUSE14 UNION ALL
SELECT * FROM WAREHOUSE15 UNION ALL
SELECT * FROM WAREHOUSE16 UNION ALL
SELECT * FROM WAREHOUSE17 UNION ALL
SELECT * FROM WAREHOUSE18 UNION ALL
SELECT * FROM WAREHOUSE19 UNION ALL
SELECT * FROM WAREHOUSE20 UNION ALL
SELECT * FROM WAREHOUSE21 UNION ALL
SELECT * FROM WAREHOUSE22 UNION ALL
SELECT * FROM WAREHOUSE23 UNION ALL
SELECT * FROM WAREHOUSE24 UNION ALL
SELECT * FROM WAREHOUSE25
WITH ROW MOVEMENT;
COMMIT WORK;
connect reset;

```

gen_customer.bat

```

C:\tpc-c.ibm\dbgen\gendata.exe -t 7 -r 1 800 -fl
C:\flats\flat_001\customer_001_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 7 -r 801 1600 -fl
C:\flats\flat_002\customer_002_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 7 -r 1601 2400 -fl
C:\flats\flat_003\customer_003_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 7 -r 2401 3200 -fl
C:\flats\flat_004\customer_004_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 7 -r 3201 4000 -fl
C:\flats\flat_005\customer_005_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 7 -r 4001 4800 -fl
C:\flats\flat_006\customer_006_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 7 -r 4801 5600 -fl
C:\flats\flat_007\customer_007_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 7 -r 5601 6400 -fl
C:\flats\flat_008\customer_008_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 7 -r 6401 7200 -fl
C:\flats\flat_009\customer_009_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 7 -r 7201 8000 -fl
C:\flats\flat_010\customer_010_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 7 -r 8001 8800 -fl
C:\flats\flat_011\customer_011_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 7 -r 8801 9600 -fl
C:\flats\flat_012\customer_012_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 7 -r 9601 10400 -fl
C:\flats\flat_013\customer_013_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 7 -r 10401 11200 -fl
C:\flats\flat_014\customer_014_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 7 -r 11201 12000 -fl
C:\flats\flat_015\customer_015_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 7 -r 12001 12800 -fl
C:\flats\flat_016\customer_016_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 7 -r 12801 13600 -fl
C:\flats\flat_017\customer_017_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 7 -r 13601 14400 -fl
C:\flats\flat_018\customer_018_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 7 -r 14401 15200 -fl
C:\flats\flat_019\customer_019_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 7 -r 15201 16000 -fl
C:\flats\flat_020\customer_020_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 7 -r 16001 16800 -fl
C:\flats\flat_021\customer_021_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 7 -r 16801 17600 -fl
C:\flats\flat_022\customer_022_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 7 -r 17601 18400 -fl
C:\flats\flat_023\customer_023_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 7 -r 18401 19200 -fl
C:\flats\flat_024\customer_024_1.dat

```

```

C:\tpc-c.ibm\dbgen\gendata.exe -t 7 -r 19201 20000 -fl
C:\flats\flat_025\customer_025_1.dat

```

gen_district.bat

```

C:\tpc-c.ibm\dbgen\gendata.exe -t 4 -r 1 800 -fl
C:\flats\flat_001\district_001_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 4 -r 801 1600 -fl
C:\flats\flat_002\district_002_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 4 -r 1601 2400 -fl
C:\flats\flat_003\district_003_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 4 -r 2401 3200 -fl
C:\flats\flat_004\district_004_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 4 -r 3201 4000 -fl
C:\flats\flat_005\district_005_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 4 -r 4001 4800 -fl
C:\flats\flat_006\district_006_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 4 -r 4801 5600 -fl
C:\flats\flat_007\district_007_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 4 -r 5601 6400 -fl
C:\flats\flat_008\district_008_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 4 -r 6401 7200 -fl
C:\flats\flat_009\district_009_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 4 -r 7201 8000 -fl
C:\flats\flat_010\district_010_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 4 -r 8001 8800 -fl
C:\flats\flat_011\district_011_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 4 -r 8801 9600 -fl
C:\flats\flat_012\district_012_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 4 -r 9601 10400 -fl
C:\flats\flat_013\district_013_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 4 -r 10401 11200 -fl
C:\flats\flat_014\district_014_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 4 -r 11201 12000 -fl
C:\flats\flat_015\district_015_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 4 -r 12001 12800 -fl
C:\flats\flat_016\district_016_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 4 -r 12801 13600 -fl
C:\flats\flat_017\district_017_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 4 -r 13601 14400 -fl
C:\flats\flat_018\district_018_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 4 -r 14401 15200 -fl
C:\flats\flat_019\district_019_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 4 -r 15201 16000 -fl
C:\flats\flat_020\district_020_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 4 -r 16001 16800 -fl
C:\flats\flat_021\district_021_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 4 -r 16801 17600 -fl
C:\flats\flat_022\district_022_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 4 -r 17601 18400 -fl
C:\flats\flat_023\district_023_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 4 -r 18401 19200 -fl
C:\flats\flat_024\district_024_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 4 -r 19201 20000 -fl
C:\flats\flat_025\district_025_1.dat

```

gen_history.bat

```

C:\tpc-c.ibm\dbgen\gendata.exe -t 8 -r 1 200 -fl
C:\flats\flat_001\history_001_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 8 -r 201 400 -fl
C:\flats\flat_001\history_001_2.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 8 -r 401 600 -fl
C:\flats\flat_001\history_001_3.dat

```



```

C:\tpc-c.ibm\dbgen\gendata.exe -t 8 -r 14201 14400 -fl
C:\flats\flat_018\history_018_4.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 8 -r 14401 14600 -fl
C:\flats\flat_019\history_019_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 8 -r 14601 14800 -fl
C:\flats\flat_019\history_019_2.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 8 -r 14801 15000 -fl
C:\flats\flat_019\history_019_3.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 8 -r 15001 15200 -fl
C:\flats\flat_019\history_019_4.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 8 -r 15201 15400 -fl
C:\flats\flat_020\history_020_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 8 -r 15401 15600 -fl
C:\flats\flat_020\history_020_2.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 8 -r 15601 15800 -fl
C:\flats\flat_020\history_020_3.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 8 -r 15801 16000 -fl
C:\flats\flat_020\history_020_4.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 8 -r 16001 16200 -fl
C:\flats\flat_021\history_021_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 8 -r 16201 16400 -fl
C:\flats\flat_021\history_021_2.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 8 -r 16401 16600 -fl
C:\flats\flat_021\history_021_3.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 8 -r 16601 16800 -fl
C:\flats\flat_021\history_021_4.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 8 -r 16801 17000 -fl
C:\flats\flat_022\history_022_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 8 -r 17001 17200 -fl
C:\flats\flat_022\history_022_2.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 8 -r 17201 17400 -fl
C:\flats\flat_022\history_022_3.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 8 -r 17401 17600 -fl
C:\flats\flat_022\history_022_4.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 8 -r 17601 17800 -fl
C:\flats\flat_023\history_023_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 8 -r 17801 18000 -fl
C:\flats\flat_023\history_023_2.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 8 -r 18001 18200 -fl
C:\flats\flat_023\history_023_3.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 8 -r 18201 18400 -fl
C:\flats\flat_023\history_023_4.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 8 -r 18401 18600 -fl
C:\flats\flat_024\history_024_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 8 -r 18601 18800 -fl
C:\flats\flat_024\history_024_2.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 8 -r 18801 19000 -fl
C:\flats\flat_024\history_024_3.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 8 -r 19001 19200 -fl
C:\flats\flat_024\history_024_4.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 8 -r 19201 19400 -fl
C:\flats\flat_025\history_025_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 8 -r 19401 19600 -fl
C:\flats\flat_025\history_025_2.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 8 -r 19601 19800 -fl
C:\flats\flat_025\history_025_3.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 8 -r 19801 20000 -fl
C:\flats\flat_025\history_025_4.dat

```

gen_item.bat

```
C:\tpc-c.ibm\dbgen\gendata.exe -t 5 -fl C:\flats\flat\item_1.dat
```

gen_new_order.bat

```

C:\tpc-c.ibm\dbgen\gendata.exe -t 11 -r 1 800 -fl
C:\flats\flat_001\neworder_001_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 11 -r 801 1600 -fl
C:\flats\flat_002\neworder_002_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 11 -r 1601 2400 -fl
C:\flats\flat_003\neworder_003_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 11 -r 2401 3200 -fl
C:\flats\flat_004\neworder_004_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 11 -r 3201 4000 -fl
C:\flats\flat_005\neworder_005_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 11 -r 4001 4800 -fl
C:\flats\flat_006\neworder_006_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 11 -r 4801 5600 -fl
C:\flats\flat_007\neworder_007_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 11 -r 5601 6400 -fl
C:\flats\flat_008\neworder_008_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 11 -r 6401 7200 -fl
C:\flats\flat_009\neworder_009_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 11 -r 7201 8000 -fl
C:\flats\flat_010\neworder_010_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 11 -r 8001 8800 -fl
C:\flats\flat_011\neworder_011_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 11 -r 8801 9600 -fl
C:\flats\flat_012\neworder_012_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 11 -r 9601 10400 -fl
C:\flats\flat_013\neworder_013_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 11 -r 10401 11200 -fl
C:\flats\flat_014\neworder_014_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 11 -r 11201 12000 -fl
C:\flats\flat_015\neworder_015_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 11 -r 12001 12800 -fl
C:\flats\flat_016\neworder_016_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 11 -r 12801 13600 -fl
C:\flats\flat_017\neworder_017_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 11 -r 13601 14400 -fl
C:\flats\flat_018\neworder_018_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 11 -r 14401 15200 -fl
C:\flats\flat_019\neworder_019_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 11 -r 15201 16000 -fl
C:\flats\flat_020\neworder_020_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 11 -r 16001 16800 -fl
C:\flats\flat_021\neworder_021_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 11 -r 16801 17600 -fl
C:\flats\flat_022\neworder_022_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 11 -r 17601 18400 -fl
C:\flats\flat_023\neworder_023_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 11 -r 18401 19200 -fl
C:\flats\flat_024\neworder_024_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 11 -r 19201 20000 -fl
C:\flats\flat_025\neworder_025_1.dat

```

gen_orders.bat

```

C:\tpc-c.ibm\dbgen\gendata.exe -t 9 -r 1 800 -fl
C:\flats\flat_001\orders_001_1.dat -f2 C:\flats\flat_001\orderline_001_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 9 -r 801 1600 -fl
C:\flats\flat_002\orders_002_1.dat -f2 C:\flats\flat_002\orderline_002_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 9 -r 1601 2400 -fl
C:\flats\flat_003\orders_003_1.dat -f2 C:\flats\flat_003\orderline_003_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 9 -r 2401 3200 -fl
C:\flats\flat_004\orders_004_1.dat -f2 C:\flats\flat_004\orderline_004_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 9 -r 3201 4000 -fl
C:\flats\flat_005\orders_005_1.dat -f2 C:\flats\flat_005\orderline_005_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 9 -r 4001 4800 -fl
C:\flats\flat_006\orders_006_1.dat -f2 C:\flats\flat_006\orderline_006_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 9 -r 4801 5600 -fl
C:\flats\flat_007\orders_007_1.dat -f2 C:\flats\flat_007\orderline_007_1.dat

```

```

C:\tpc-c.ibm\dbgen\gendata.exe -t 9 -r 5601 6400 -fl
C:\flats\flat_008\orders_008_1.dat -f2 C:\flats\flat_008\orderline_008_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 9 -r 6401 7200 -fl
C:\flats\flat_009\orders_009_1.dat -f2 C:\flats\flat_009\orderline_009_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 9 -r 7201 8000 -fl
C:\flats\flat_010\orders_010_1.dat -f2 C:\flats\flat_010\orderline_010_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 9 -r 8001 8800 -fl
C:\flats\flat_011\orders_011_1.dat -f2 C:\flats\flat_011\orderline_011_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 9 -r 8801 9600 -fl
C:\flats\flat_012\orders_012_1.dat -f2 C:\flats\flat_012\orderline_012_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 9 -r 9601 10400 -fl
C:\flats\flat_013\orders_013_1.dat -f2 C:\flats\flat_013\orderline_013_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 9 -r 10401 11200 -fl
C:\flats\flat_014\orders_014_1.dat -f2 C:\flats\flat_014\orderline_014_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 9 -r 11201 12000 -fl
C:\flats\flat_015\orders_015_1.dat -f2 C:\flats\flat_015\orderline_015_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 9 -r 12001 12800 -fl
C:\flats\flat_016\orders_016_1.dat -f2 C:\flats\flat_016\orderline_016_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 9 -r 12801 13600 -fl
C:\flats\flat_017\orders_017_1.dat -f2 C:\flats\flat_017\orderline_017_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 9 -r 13601 14400 -fl
C:\flats\flat_018\orders_018_1.dat -f2 C:\flats\flat_018\orderline_018_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 9 -r 14401 15200 -fl
C:\flats\flat_019\orders_019_1.dat -f2 C:\flats\flat_019\orderline_019_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 9 -r 15201 16000 -fl
C:\flats\flat_020\orders_020_1.dat -f2 C:\flats\flat_020\orderline_020_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 9 -r 16001 16800 -fl
C:\flats\flat_021\orders_021_1.dat -f2 C:\flats\flat_021\orderline_021_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 9 -r 16801 17600 -fl
C:\flats\flat_022\orders_022_1.dat -f2 C:\flats\flat_022\orderline_022_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 9 -r 17601 18400 -fl
C:\flats\flat_023\orders_023_1.dat -f2 C:\flats\flat_023\orderline_023_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 9 -r 18401 19200 -fl
C:\flats\flat_024\orders_024_1.dat -f2 C:\flats\flat_024\orderline_024_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 9 -r 19201 20000 -fl
C:\flats\flat_025\orders_025_1.dat -f2 C:\flats\flat_025\orderline_025_1.dat

```

gen_stock.bat

```

C:\tpc-c.ibm\dbgen\gendata.exe -t 6 -r 1 800 -fl
C:\flats\flat_001\stock_001_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 6 -r 801 1600 -fl
C:\flats\flat_002\stock_002_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 6 -r 1601 2400 -fl
C:\flats\flat_003\stock_003_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 6 -r 2401 3200 -fl
C:\flats\flat_004\stock_004_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 6 -r 3201 4000 -fl
C:\flats\flat_005\stock_005_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 6 -r 4001 4800 -fl
C:\flats\flat_006\stock_006_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 6 -r 4801 5600 -fl
C:\flats\flat_007\stock_007_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 6 -r 5601 6400 -fl
C:\flats\flat_008\stock_008_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 6 -r 6401 7200 -fl
C:\flats\flat_009\stock_009_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 6 -r 7201 8000 -fl
C:\flats\flat_010\stock_010_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 6 -r 8001 8800 -fl
C:\flats\flat_011\stock_011_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 6 -r 8801 9600 -fl
C:\flats\flat_012\stock_012_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 6 -r 9601 10400 -fl
C:\flats\flat_013\stock_013_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 6 -r 10401 11200 -fl
C:\flats\flat_014\stock_014_1.dat

```

```

C:\tpc-c.ibm\dbgen\gendata.exe -t 6 -r 11201 12000 -fl
C:\flats\flat_015\stock_015_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 6 -r 12001 12800 -fl
C:\flats\flat_016\stock_016_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 6 -r 12801 13600 -fl
C:\flats\flat_017\stock_017_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 6 -r 13601 14400 -fl
C:\flats\flat_018\stock_018_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 6 -r 14401 15200 -fl
C:\flats\flat_019\stock_019_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 6 -r 15201 16000 -fl
C:\flats\flat_020\stock_020_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 6 -r 16001 16800 -fl
C:\flats\flat_021\stock_021_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 6 -r 16801 17600 -fl
C:\flats\flat_022\stock_022_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 6 -r 17601 18400 -fl
C:\flats\flat_023\stock_023_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 6 -r 18401 19200 -fl
C:\flats\flat_024\stock_024_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 6 -r 19201 20000 -fl
C:\flats\flat_025\stock_025_1.dat

```

gen_warehouse.bat

```

C:\tpc-c.ibm\dbgen\gendata.exe -t 3 -r 1 800 -fl
C:\flats\flat_001\warehouse_001_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 3 -r 801 1600 -fl
C:\flats\flat_002\warehouse_002_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 3 -r 1601 2400 -fl
C:\flats\flat_003\warehouse_003_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 3 -r 2401 3200 -fl
C:\flats\flat_004\warehouse_004_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 3 -r 3201 4000 -fl
C:\flats\flat_005\warehouse_005_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 3 -r 4001 4800 -fl
C:\flats\flat_006\warehouse_006_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 3 -r 4801 5600 -fl
C:\flats\flat_007\warehouse_007_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 3 -r 5601 6400 -fl
C:\flats\flat_008\warehouse_008_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 3 -r 6401 7200 -fl
C:\flats\flat_009\warehouse_009_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 3 -r 7201 8000 -fl
C:\flats\flat_010\warehouse_010_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 3 -r 8001 8800 -fl
C:\flats\flat_011\warehouse_011_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 3 -r 8801 9600 -fl
C:\flats\flat_012\warehouse_012_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 3 -r 9601 10400 -fl
C:\flats\flat_013\warehouse_013_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 3 -r 10401 11200 -fl
C:\flats\flat_014\warehouse_014_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 3 -r 11201 12000 -fl
C:\flats\flat_015\warehouse_015_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 3 -r 12001 12800 -fl
C:\flats\flat_016\warehouse_016_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 3 -r 12801 13600 -fl
C:\flats\flat_017\warehouse_017_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 3 -r 13601 14400 -fl
C:\flats\flat_018\warehouse_018_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 3 -r 14401 15200 -fl
C:\flats\flat_019\warehouse_019_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 3 -r 15201 16000 -fl
C:\flats\flat_020\warehouse_020_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 3 -r 16001 16800 -fl
C:\flats\flat_021\warehouse_021_1.dat

```

```

C:\tpc-c.ibm\dbgen\gendata.exe -t 3 -r 16801 17600 -fl
C:\flats\flat_022\warehouse_022_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 3 -r 17601 18400 -fl
C:\flats\flat_023\warehouse_023_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 3 -r 18401 19200 -fl
C:\flats\flat_024\warehouse_024_1.dat
C:\tpc-c.ibm\dbgen\gendata.exe -t 3 -r 19201 20000 -fl
C:\flats\flat_025\warehouse_025_1.dat

```

load_customer_all.ddl

```

CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER1 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_001\customer_001_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
24000000 INSERT INTO CUSTOMER1;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER2 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_002\customer_002_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
24000000 INSERT INTO CUSTOMER2;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER3 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_003\customer_003_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
24000000 INSERT INTO CUSTOMER3;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER4 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_004\customer_004_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
24000000 INSERT INTO CUSTOMER4;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER5 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_005\customer_005_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
24000000 INSERT INTO CUSTOMER5;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER6 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_006\customer_006_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
24000000 INSERT INTO CUSTOMER6;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER7 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_007\customer_007_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
24000000 INSERT INTO CUSTOMER7;
COMMIT WORK;
CONNECT RESET;

```

```

CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER8 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_008\customer_008_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
24000000 INSERT INTO CUSTOMER8;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER9 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_009\customer_009_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
24000000 INSERT INTO CUSTOMER9;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER10 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_010\customer_010_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
24000000 INSERT INTO CUSTOMER10;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER11 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_011\customer_011_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
24000000 INSERT INTO CUSTOMER11;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER12 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_012\customer_012_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
24000000 INSERT INTO CUSTOMER12;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER13 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_013\customer_013_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
24000000 INSERT INTO CUSTOMER13;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER14 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_014\customer_014_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
24000000 INSERT INTO CUSTOMER14;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER15 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_015\customer_015_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
24000000 INSERT INTO CUSTOMER15;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER16 ACTIVATE NOT LOGGED INITIALLY;

```

```

IMPORT FROM C:\flats\flat_016\customer_016_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
24000000 INSERT INTO CUSTOMER16;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER17 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_017\customer_017_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
24000000 INSERT INTO CUSTOMER17;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER18 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_018\customer_018_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
24000000 INSERT INTO CUSTOMER18;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER19 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_019\customer_019_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
24000000 INSERT INTO CUSTOMER19;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER20 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_020\customer_020_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
24000000 INSERT INTO CUSTOMER20;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER21 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_021\customer_021_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
24000000 INSERT INTO CUSTOMER21;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER22 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_022\customer_022_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
24000000 INSERT INTO CUSTOMER22;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER23 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_023\customer_023_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
24000000 INSERT INTO CUSTOMER23;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER24 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_024\customer_024_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
24000000 INSERT INTO CUSTOMER24;
COMMIT WORK;

```

```

CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER25 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_025\customer_025_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
24000000 INSERT INTO CUSTOMER25;
COMMIT WORK;
CONNECT RESET;

```

load_district_all.ddl

```

CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_001\district_001_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO DISTRICT1;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_002\district_002_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO DISTRICT2;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_003\district_003_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO DISTRICT3;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_004\district_004_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO DISTRICT4;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_005\district_005_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO DISTRICT5;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_006\district_006_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO DISTRICT6;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_007\district_007_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO DISTRICT7;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_008\district_008_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO DISTRICT8;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_009\district_009_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO DISTRICT9;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;

```

```

IMPORT FROM C:\flats\flat_010\district_010_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO DISTRICT10;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_011\district_011_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO DISTRICT11;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_012\district_012_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO DISTRICT12;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_013\district_013_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO DISTRICT13;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_014\district_014_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO DISTRICT14;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_015\district_015_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO DISTRICT15;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_016\district_016_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO DISTRICT16;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_017\district_017_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO DISTRICT17;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_018\district_018_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO DISTRICT18;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_019\district_019_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO DISTRICT19;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_020\district_020_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO DISTRICT20;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;

```

```

IMPORT FROM C:\flats\flat_021\district_021_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO DISTRICT21;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_022\district_022_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO DISTRICT22;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_023\district_023_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO DISTRICT23;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_024\district_024_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO DISTRICT24;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_025\district_025_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO DISTRICT25;
COMMIT WORK;
CONNECT RESET;

```

Load_history_all.ddl

connect to TPCC in share mode;

```

LOAD FROM C:\flats\flat_001\history_001_1.dat,
C:\flats\flat_001\history_001_2.dat, C:\flats\flat_001\history_001_3.dat,
C:\flats\flat_001\history_001_4.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS FASTPARSE REPLACE INTO HISTORY1
NONRECOVERABLE DATA BUFFER 5000 CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM C:\flats\flat_002\history_002_1.dat,
C:\flats\flat_002\history_002_2.dat, C:\flats\flat_002\history_002_3.dat,
C:\flats\flat_002\history_002_4.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS FASTPARSE REPLACE INTO HISTORY2
NONRECOVERABLE DATA BUFFER 5000 CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM C:\flats\flat_003\history_003_1.dat,
C:\flats\flat_003\history_003_2.dat, C:\flats\flat_003\history_003_3.dat,
C:\flats\flat_003\history_003_4.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS FASTPARSE REPLACE INTO HISTORY3
NONRECOVERABLE DATA BUFFER 5000 CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM C:\flats\flat_004\history_004_1.dat,
C:\flats\flat_004\history_004_2.dat, C:\flats\flat_004\history_004_3.dat,
C:\flats\flat_004\history_004_4.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS FASTPARSE REPLACE INTO HISTORY4
NONRECOVERABLE DATA BUFFER 5000 CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM C:\flats\flat_005\history_005_1.dat,
C:\flats\flat_005\history_005_2.dat, C:\flats\flat_005\history_005_3.dat,
C:\flats\flat_005\history_005_4.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS FASTPARSE REPLACE INTO HISTORY5
NONRECOVERABLE DATA BUFFER 5000 CPU_PARALLELISM 4 ;

```



```
LOAD FROM C:\flats\flat_025\history_025_1.dat,
C:\flats\flat_025\history_025_2.dat, C:\flats\flat_025\history_025_3.dat,
C:\flats\flat_025\history_025_4.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS FASTPARSE REPLACE INTO HISTORY25
NONRECOVERABLE DATA BUFFER 5000 CPU_PARALLELISM 4 ;
connect reset;
```

load_item_all.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat\item_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS COMPOUND=50 COMMITCOUNT 1000 INSERT INTO
ITEM;
COMMIT WORK;
CONNECT RESET;
```

load_new_order_all.ddl

```
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_001\neworder_001_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 20000 INSERT INTO NEW_ORDERA1;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_002\neworder_002_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 20000 INSERT INTO NEW_ORDERA2;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_003\neworder_003_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 20000 INSERT INTO NEW_ORDERA3;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_004\neworder_004_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 20000 INSERT INTO NEW_ORDERA4;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_005\neworder_005_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 20000 INSERT INTO NEW_ORDERA5;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_006\neworder_006_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 20000 INSERT INTO NEW_ORDERA6;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_007\neworder_007_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 20000 INSERT INTO NEW_ORDERA7;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_008\neworder_008_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 20000 INSERT INTO NEW_ORDERA8;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
```

```
IMPORT FROM C:\flats\flat_009\neworder_009_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 20000 INSERT INTO NEW_ORDERA9;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_010\neworder_010_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 20000 INSERT INTO NEW_ORDERA10;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_011\neworder_011_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 20000 INSERT INTO NEW_ORDERA11;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_012\neworder_012_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 20000 INSERT INTO NEW_ORDERA12;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_013\neworder_013_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 20000 INSERT INTO NEW_ORDERA13;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_014\neworder_014_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 20000 INSERT INTO NEW_ORDERA14;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_015\neworder_015_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 20000 INSERT INTO NEW_ORDERA15;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_016\neworder_016_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 20000 INSERT INTO NEW_ORDERA16;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_017\neworder_017_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 20000 INSERT INTO NEW_ORDERA17;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_018\neworder_018_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 20000 INSERT INTO NEW_ORDERA18;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_019\neworder_019_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 20000 INSERT INTO NEW_ORDERA19;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
```

```

IMPORT FROM C:\flats\flat_020\neworder_020_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 20000 INSERT INTO NEW_ORDERA20;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_021\neworder_021_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 20000 INSERT INTO NEW_ORDERA21;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_022\neworder_022_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 20000 INSERT INTO NEW_ORDERA22;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_023\neworder_023_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 20000 INSERT INTO NEW_ORDERA23;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_024\neworder_024_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 20000 INSERT INTO NEW_ORDERA24;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_025\neworder_025_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 20000 INSERT INTO NEW_ORDERA25;
COMMIT WORK;
CONNECT RESET;

```

load_order_line_all.ddl

```

CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE1 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_001\orderline_001_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
264000000 INSERT INTO ORDER_LINE1;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE2 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_002\orderline_002_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
264000000 INSERT INTO ORDER_LINE2;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE3 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_003\orderline_003_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
264000000 INSERT INTO ORDER_LINE3;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE4 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_004\orderline_004_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
264000000 INSERT INTO ORDER_LINE4;

```

```

COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE5 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_005\orderline_005_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
264000000 INSERT INTO ORDER_LINE5;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE6 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_006\orderline_006_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
264000000 INSERT INTO ORDER_LINE6;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE7 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_007\orderline_007_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
264000000 INSERT INTO ORDER_LINE7;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE8 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_008\orderline_008_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
264000000 INSERT INTO ORDER_LINE8;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE9 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_009\orderline_009_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
264000000 INSERT INTO ORDER_LINE9;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE10 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_010\orderline_010_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
264000000 INSERT INTO ORDER_LINE10;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE11 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_011\orderline_011_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
264000000 INSERT INTO ORDER_LINE11;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE12 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_012\orderline_012_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT
264000000 INSERT INTO ORDER_LINE12;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;

```

```

ALTER TABLE ORDER_LINE13 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_013\orderline_013_1.dat OF DEL MODIFIED
BY COLDEL|KEEPBLANKS COMPOUND=50 COMMITCOUNT
264000000 INSERT INTO ORDER_LINE13;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE14 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_014\orderline_014_1.dat OF DEL MODIFIED
BY COLDEL|KEEPBLANKS COMPOUND=50 COMMITCOUNT
264000000 INSERT INTO ORDER_LINE14;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE15 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_015\orderline_015_1.dat OF DEL MODIFIED
BY COLDEL|KEEPBLANKS COMPOUND=50 COMMITCOUNT
264000000 INSERT INTO ORDER_LINE15;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE16 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_016\orderline_016_1.dat OF DEL MODIFIED
BY COLDEL|KEEPBLANKS COMPOUND=50 COMMITCOUNT
264000000 INSERT INTO ORDER_LINE16;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE17 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_017\orderline_017_1.dat OF DEL MODIFIED
BY COLDEL|KEEPBLANKS COMPOUND=50 COMMITCOUNT
264000000 INSERT INTO ORDER_LINE17;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE18 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_018\orderline_018_1.dat OF DEL MODIFIED
BY COLDEL|KEEPBLANKS COMPOUND=50 COMMITCOUNT
264000000 INSERT INTO ORDER_LINE18;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE19 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_019\orderline_019_1.dat OF DEL MODIFIED
BY COLDEL|KEEPBLANKS COMPOUND=50 COMMITCOUNT
264000000 INSERT INTO ORDER_LINE19;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE20 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_020\orderline_020_1.dat OF DEL MODIFIED
BY COLDEL|KEEPBLANKS COMPOUND=50 COMMITCOUNT
264000000 INSERT INTO ORDER_LINE20;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE21 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_021\orderline_021_1.dat OF DEL MODIFIED
BY COLDEL|KEEPBLANKS COMPOUND=50 COMMITCOUNT
264000000 INSERT INTO ORDER_LINE21;

```

```

COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE22 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_022\orderline_022_1.dat OF DEL MODIFIED
BY COLDEL|KEEPBLANKS COMPOUND=50 COMMITCOUNT
264000000 INSERT INTO ORDER_LINE22;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE23 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_023\orderline_023_1.dat OF DEL MODIFIED
BY COLDEL|KEEPBLANKS COMPOUND=50 COMMITCOUNT
264000000 INSERT INTO ORDER_LINE23;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE24 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_024\orderline_024_1.dat OF DEL MODIFIED
BY COLDEL|KEEPBLANKS COMPOUND=50 COMMITCOUNT
264000000 INSERT INTO ORDER_LINE24;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE25 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_025\orderline_025_1.dat OF DEL MODIFIED
BY COLDEL|KEEPBLANKS COMPOUND=50 COMMITCOUNT
264000000 INSERT INTO ORDER_LINE25;
COMMIT WORK;
CONNECT RESET;

```

load_orders_all.ddl

```

CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS1 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_001\orders_001_1.dat OF DEL MODIFIED BY
COLDEL|KEEPBLANKS COMPOUND=50 COMMITCOUNT 24000000
INSERT INTO ORDERS1;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS2 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_002\orders_002_1.dat OF DEL MODIFIED BY
COLDEL|KEEPBLANKS COMPOUND=50 COMMITCOUNT 24000000
INSERT INTO ORDERS2;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS3 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_003\orders_003_1.dat OF DEL MODIFIED BY
COLDEL|KEEPBLANKS COMPOUND=50 COMMITCOUNT 24000000
INSERT INTO ORDERS3;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS4 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_004\orders_004_1.dat OF DEL MODIFIED BY
COLDEL|KEEPBLANKS COMPOUND=50 COMMITCOUNT 24000000
INSERT INTO ORDERS4;
COMMIT WORK;

```

```

CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS5 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_005\orders_005_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 24000000
INSERT INTO ORDERS5;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS6 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_006\orders_006_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 24000000
INSERT INTO ORDERS6;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS7 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_007\orders_007_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 24000000
INSERT INTO ORDERS7;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS8 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_008\orders_008_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 24000000
INSERT INTO ORDERS8;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS9 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_009\orders_009_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 24000000
INSERT INTO ORDERS9;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS10 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_010\orders_010_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 24000000
INSERT INTO ORDERS10;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS11 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_011\orders_011_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 24000000
INSERT INTO ORDERS11;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS12 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_012\orders_012_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 24000000
INSERT INTO ORDERS12;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS13 ACTIVATE NOT LOGGED INITIALLY;

```

```

IMPORT FROM C:\flats\flat_013\orders_013_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 24000000
INSERT INTO ORDERS13;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS14 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_014\orders_014_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 24000000
INSERT INTO ORDERS14;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS15 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_015\orders_015_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 24000000
INSERT INTO ORDERS15;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS16 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_016\orders_016_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 24000000
INSERT INTO ORDERS16;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS17 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_017\orders_017_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 24000000
INSERT INTO ORDERS17;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS18 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_018\orders_018_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 24000000
INSERT INTO ORDERS18;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS19 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_019\orders_019_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 24000000
INSERT INTO ORDERS19;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS20 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_020\orders_020_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 24000000
INSERT INTO ORDERS20;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS21 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_021\orders_021_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 24000000
INSERT INTO ORDERS21;
COMMIT WORK;

```

```

CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS22 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_022\orders_022_1.dat OF DEL MODIFIED BY
COLDEL|KEEPBLANKS COMPOUND=50 COMMITCOUNT 24000000
INSERT INTO ORDERS22;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS23 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_023\orders_023_1.dat OF DEL MODIFIED BY
COLDEL|KEEPBLANKS COMPOUND=50 COMMITCOUNT 24000000
INSERT INTO ORDERS23;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS24 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_024\orders_024_1.dat OF DEL MODIFIED BY
COLDEL|KEEPBLANKS COMPOUND=50 COMMITCOUNT 24000000
INSERT INTO ORDERS24;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDERS25 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_025\orders_025_1.dat OF DEL MODIFIED BY
COLDEL|KEEPBLANKS COMPOUND=50 COMMITCOUNT 24000000
INSERT INTO ORDERS25;
COMMIT WORK;
CONNECT RESET;

```

load_stock_all.ddl

```

CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK1 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_001\stock_001_1.dat OF DEL MODIFIED BY
COLDEL|KEEPBLANKS COMPOUND=50 COMMITCOUNT 80000000
INSERT INTO STOCK1;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK2 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_002\stock_002_1.dat OF DEL MODIFIED BY
COLDEL|KEEPBLANKS COMPOUND=50 COMMITCOUNT 80000000
INSERT INTO STOCK2;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK3 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_003\stock_003_1.dat OF DEL MODIFIED BY
COLDEL|KEEPBLANKS COMPOUND=50 COMMITCOUNT 80000000
INSERT INTO STOCK3;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK4 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_004\stock_004_1.dat OF DEL MODIFIED BY
COLDEL|KEEPBLANKS COMPOUND=50 COMMITCOUNT 80000000
INSERT INTO STOCK4;
COMMIT WORK;
CONNECT RESET;

```

```

CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK5 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_005\stock_005_1.dat OF DEL MODIFIED BY
COLDEL|KEEPBLANKS COMPOUND=50 COMMITCOUNT 80000000
INSERT INTO STOCK5;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK6 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_006\stock_006_1.dat OF DEL MODIFIED BY
COLDEL|KEEPBLANKS COMPOUND=50 COMMITCOUNT 80000000
INSERT INTO STOCK6;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK7 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_007\stock_007_1.dat OF DEL MODIFIED BY
COLDEL|KEEPBLANKS COMPOUND=50 COMMITCOUNT 80000000
INSERT INTO STOCK7;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK8 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_008\stock_008_1.dat OF DEL MODIFIED BY
COLDEL|KEEPBLANKS COMPOUND=50 COMMITCOUNT 80000000
INSERT INTO STOCK8;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK9 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_009\stock_009_1.dat OF DEL MODIFIED BY
COLDEL|KEEPBLANKS COMPOUND=50 COMMITCOUNT 80000000
INSERT INTO STOCK9;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK10 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_010\stock_010_1.dat OF DEL MODIFIED BY
COLDEL|KEEPBLANKS COMPOUND=50 COMMITCOUNT 80000000
INSERT INTO STOCK10;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK11 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_011\stock_011_1.dat OF DEL MODIFIED BY
COLDEL|KEEPBLANKS COMPOUND=50 COMMITCOUNT 80000000
INSERT INTO STOCK11;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK12 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_012\stock_012_1.dat OF DEL MODIFIED BY
COLDEL|KEEPBLANKS COMPOUND=50 COMMITCOUNT 80000000
INSERT INTO STOCK12;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK13 ACTIVATE NOT LOGGED INITIALLY;

```

```

IMPORT FROM C:\flats\flat_013\stock_013_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 80000000
INSERT INTO STOCK13;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK14 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_014\stock_014_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 80000000
INSERT INTO STOCK14;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK15 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_015\stock_015_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 80000000
INSERT INTO STOCK15;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK16 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_016\stock_016_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 80000000
INSERT INTO STOCK16;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK17 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_017\stock_017_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 80000000
INSERT INTO STOCK17;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK18 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_018\stock_018_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 80000000
INSERT INTO STOCK18;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK19 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_019\stock_019_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 80000000
INSERT INTO STOCK19;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK20 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_020\stock_020_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 80000000
INSERT INTO STOCK20;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK21 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_021\stock_021_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 80000000
INSERT INTO STOCK21;
COMMIT WORK;

```

```

CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK22 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_022\stock_022_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 80000000
INSERT INTO STOCK22;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK23 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_023\stock_023_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 80000000
INSERT INTO STOCK23;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK24 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_024\stock_024_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 80000000
INSERT INTO STOCK24;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK25 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM C:\flats\flat_025\stock_025_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 80000000
INSERT INTO STOCK25;
COMMIT WORK;
CONNECT RESET;

```

load_warehouse_all.ddl

```

CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_001\warehouse_001_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO WAREHOUSE1;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_002\warehouse_002_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO WAREHOUSE2;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_003\warehouse_003_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO WAREHOUSE3;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_004\warehouse_004_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO WAREHOUSE4;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_005\warehouse_005_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO WAREHOUSE5;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;

```

```

IMPORT FROM C:\flats\flat_006\warehouse_006_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO WAREHOUSE6;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_007\warehouse_007_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO WAREHOUSE7;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_008\warehouse_008_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO WAREHOUSE8;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_009\warehouse_009_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO WAREHOUSE9;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_010\warehouse_010_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO WAREHOUSE10;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_011\warehouse_011_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO WAREHOUSE11;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_012\warehouse_012_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO WAREHOUSE12;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_013\warehouse_013_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO WAREHOUSE13;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_014\warehouse_014_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO WAREHOUSE14;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_015\warehouse_015_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO WAREHOUSE15;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_016\warehouse_016_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO WAREHOUSE16;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;

```

```

IMPORT FROM C:\flats\flat_017\warehouse_017_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO WAREHOUSE17;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_018\warehouse_018_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO WAREHOUSE18;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_019\warehouse_019_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO WAREHOUSE19;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_020\warehouse_020_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO WAREHOUSE20;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_021\warehouse_021_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO WAREHOUSE21;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_022\warehouse_022_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO WAREHOUSE22;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_023\warehouse_023_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO WAREHOUSE23;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_024\warehouse_024_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO WAREHOUSE24;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM C:\flats\flat_025\warehouse_025_1.dat OF DEL MODIFIED
BY COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS
COMMITCOUNT 1000 INSERT INTO WAREHOUSE25;
COMMIT WORK;
CONNECT RESET;

```

rnst_customer.ddl

```

connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.CUSTOMER1 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.CUSTOMER2 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.CUSTOMER3 AND INDEXES ALL;
COMMIT WORK;
connect reset;

```



```

COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.HISTORY17 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.HISTORY18 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.HISTORY19 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.HISTORY20 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.HISTORY21 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.HISTORY22 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.HISTORY23 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.HISTORY24 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.HISTORY25 AND INDEXES ALL;
COMMIT WORK;
connect reset;

```

rnst_item.ddl

```

connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.ITEM AND INDEXES ALL;
COMMIT WORK;
connect reset;

```

rnst_new_ordera.ddl

```

connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERA1 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERA2 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERA3 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERA4 AND INDEXES ALL;
COMMIT WORK;

```

```

connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERA5 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERA6 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERA7 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERA8 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERA9 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERA10 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERA11 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERA12 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERA13 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERA14 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERA15 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERA16 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERA17 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;

```

```

RUNSTATS ON TABLE Administrator.NEW_ORDERA18 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERA19 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERA20 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERA21 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERA22 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERA23 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERA24 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERA25 AND INDEXES
ALL;
COMMIT WORK;
connect reset;

```

rnst_new_orderb.ddl

```

connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERB1 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERB2 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERB3 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERB4 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERB5 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;

```

```

RUNSTATS ON TABLE Administrator.NEW_ORDERB6 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERB7 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERB8 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERB9 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERB10 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERB11 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERB12 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERB13 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERB14 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERB15 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERB16 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERB17 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERB18 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERB19 AND INDEXES
ALL;
COMMIT WORK;

```

```

connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERB20 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERB21 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERB22 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERB23 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERB24 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.NEW_ORDERB25 AND INDEXES
ALL;
COMMIT WORK;
connect reset;

```

rnst_order_line.ddl

```

connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.ORDER_LINE1 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.ORDER_LINE2 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.ORDER_LINE3 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.ORDER_LINE4 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.ORDER_LINE5 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.ORDER_LINE6 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.ORDER_LINE7 AND INDEXES
ALL;
COMMIT WORK;

```

```

connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.ORDER_LINE8 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.ORDER_LINE9 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.ORDER_LINE10 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.ORDER_LINE11 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.ORDER_LINE12 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.ORDER_LINE13 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.ORDER_LINE14 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.ORDER_LINE15 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.ORDER_LINE16 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.ORDER_LINE17 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.ORDER_LINE18 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.ORDER_LINE19 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.ORDER_LINE20 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;

```



```

connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.WAREHOUSE9 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.WAREHOUSE10 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.WAREHOUSE11 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.WAREHOUSE12 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.WAREHOUSE13 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.WAREHOUSE14 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.WAREHOUSE15 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.WAREHOUSE16 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.WAREHOUSE17 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.WAREHOUSE18 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.WAREHOUSE19 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.WAREHOUSE20 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.WAREHOUSE21 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;

```

```

RUNSTATS ON TABLE Administrator.WAREHOUSE22 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.WAREHOUSE23 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.WAREHOUSE24 AND INDEXES
ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE Administrator.WAREHOUSE25 AND INDEXES
ALL;
COMMIT WORK;
connect reset;

```

DBGEN

dbgen\gendata.c

```

/*****
*****
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2005
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****
*****/

/*
 * gendata.c - Generate data for TPC-C database
 */

#include <stdlib.h>
#include <stdio.h>
#include <string.h>
#include <sqlutil.h>
/* NT named pipe support */
#include <windows.h>
#include <time.h>

#include "platform.h"
#include "db2tpcc.h"
#include "tpccrnd.h"
#include "tpccmisc.h"
#include "lval.h"

/* PROTOTYPES. */
void gen_dist_tbl( void );
void gen_cust_tbl( void );
void gen_hist_tbl( void );
void gen_nu_ord_tbl( void );
void gen_ordr_tbl( void );
void gen_item_tbl( void );
void gen_stock_tbl( void );
void gen_ware_tbl( void );

```



```

if (option < 3 || option > 11 || option == 10)
{
    fprintf(stderr,"gendata: Invalid table selected: %d\n",option);
    exit(-1);
}

/* Validate Delimiter Argument */
if (delim == NULL) {
    // default delimiter is used for IMPORT & LOAD, no changes necessary
    using_rctload = 0;
} else if (strlen(delim) == 1 && !isalnum(delim[0]) &&
    delim[0] != '.' && delim[0] != '%')
{
    // user-supplied delimiter used for rctload
    InitFormatStrings(delim[0]);
    using_rctload = 1;
} else {
    fprintf(stderr,"gendata: Invalid delimiter specified: %s\n",delim);
    exit(-1);
}

/* Validate File/Pipe Arguments */
if (option != 9 && outtype1 > 0 && outtype2 > 0)
{
    fprintf(stderr,"gendata: Specifying two output file/pipes allowed only when
generating\norders/orderline.\n");
    exit(-1);
}
if (option == 9 && ((outtype1 == 0) || (outtype2 == 0)))
{
    fprintf(stderr,"gendata: Must specify two output file/pipes when generating
orders/orderline.\n");
    exit(-1);
}
if (outtype1 == 0 || outname1 == NULL || strcmp(outname1,"") == 0)
{
    fprintf(stderr,"gendata: Invalid 1st output file/pipe specified.\n");
    exit(-1);
}
if (option == 9 && (outtype2 == 0 || outname2 == NULL ||
strcmp(outname2,"") == 0))
{
    fprintf(stderr,"gendata: Invalid 2nd output file/pipe specified.\n");
    exit(-1);
}
/* Ensure O/OL flat files are opened in append mode. This is required */
/* because we generate O/OL concurrently. See comments in genload.pl */
/* for further details on why this is necessary. */
if (option == 9)
{
    if (outtype1 == IOH_FILE) outtype1 = IOH_FILE_APPEND;
    if (outtype2 == IOH_FILE) outtype2 = IOH_FILE_APPEND;
}

/* Validate Range Arguments */
if (ware_start <= 0 || ware_start > WAREHOUSES) {
    fprintf(stderr,"gendata: Invalid range starting value: %d\n",ware_start);
    exit(-1);
}
if (ware_end <= 0 || ware_end > WAREHOUSES || ware_end < ware_start) {
    fprintf(stderr,"gendata: Invalid range ending value: %d\n",ware_end);
    exit(-1);
}

initialize_random();

```

```

/*
*****
*/
/* Generate Data */
/*
*****
*/
switch (option) {
case 3: /* WAREHOUSE */
    gen_ware_tbl();
    break;
case 4: /* DISTRICT */
    gen_dist_tbl();
    break;
case 5: /* ITEM */
    gen_item_tbl();
    break;
case 6: /* STOCK */
    gen_stock_tbl();
    break;
case 7: /* CUSTOMER */
    gen_cust_tbl();
    break;
case 8: /* HISTORY */
    gen_hist_tbl();
    break;
case 9: /* ORDERS + ORDER_LINE */
    gen_ordr_tbl();
    break;
case 11: /* NEW_ORDER */
    gen_nu_ord_tbl();
    break;
case 2:
case 10:
default:
    fprintf(stderr, "Error: invalid option = %d\n", (option));
    break;
}
return 0;
}

/*-----*/
/* generate item table */
/*-----*/

void gen_item_tbl( void )
{
    sqlint32 item_num = 0;
    sqlint32 item_im_id;
    char item_name[25];
    sqlint32 item_price;
    char item_data[51];

    IOH_NUM numBytes;
    ioHandle hnd;
    char Buffer[1024];

    timestamp1 = current_time();

    rc = GenericOpen(&hnd, outtype1, outname1);
    if (rc != 0) { goto item_done; }

    for(item_num = 1; item_num <= ITEMS; item_num++)
    {
        /* create image id field */
        item_im_id = rand_integer( 1, 10000 );
        /* create name field */
        create_random_a_string( item_name, 14, 24);
    }
}

```

```

/* create price field */
item_price = rand_integer( 100, 10000 );
/* create ORIGINAL field */
create_a_string_with_original( item_data, 26, 50, 10 );

numBytes = sprintf(Buffer, fmtItem,
    item_name,
    item_price,
    item_data,
    item_im_id,
    item_num);

rc = GenericWrite(&hnd, Buffer, numBytes);
if (rc != 0) { goto item_done; }

} /* end for... */

rc = GenericClose(&hnd);

item_done:

timestamp2 = current_time();
elapsed = timestamp2 - timestamp1;
if (rc == 0) {
    fprintf(stdout, "\nITEM table generated in %8.2f seconds.\n\n", elapsed);
    fflush(stdout);
} else {
    fprintf(stderr, "\nITEM table FAILED at (I %d) after %8.2f
seconds.\n\n", item_num, elapsed);
    fflush(stderr);
}
}

/*-----*/
/* generate stock table */
/*-----*/
void gen_stock_tbl( void )
{
    sqlint32 ware_num = 0 ;
    sqlint32 stock_num = 0 ;
    sqlint32 stock_quant;
    sqlint32 s_ytd;
    sqlint32 s_order_cnt, s_remote_cnt;
    char stock_dist_01[25] ;
    char stock_dist_02[25] ;
    char stock_dist_03[25] ;
    char stock_dist_04[25] ;
    char stock_dist_05[25] ;
    char stock_dist_06[25] ;
    char stock_dist_07[25] ;
    char stock_dist_08[25] ;
    char stock_dist_09[25] ;
    char stock_dist_10[25] ;
    char stock_data[51] ;

    IOH_NUM numBytes;
    ioHandle hnd;
    char Buffer[1024];

    timestamp1 = current_time();

    rc = GenericOpen(&hnd, outtype1, outname1);
    if (rc != 0) { goto stock_done; }

    for (stock_num = 1; stock_num <= STOCK_PER_WAREHOUSE;
stock_num++)
    {
        if (!quiet_mode && (stock_num%500 == 0))

```

```

{
    fprintf(stdout, "STOCK for Item #%"d\n", stock_num);
    fflush(stdout);
}
}
for (ware_num = ware_start; ware_num <= ware_end; ware_num++)
{
    stock_quant = rand_integer( 10, 100 );
    create_random_a_string( stock_dist_01, 24, 24);
    create_random_a_string( stock_dist_02, 24, 24);
    create_random_a_string( stock_dist_03, 24, 24);
    create_random_a_string( stock_dist_04, 24, 24);
    create_random_a_string( stock_dist_05, 24, 24);
    create_random_a_string( stock_dist_06, 24, 24);
    create_random_a_string( stock_dist_07, 24, 24);
    create_random_a_string( stock_dist_08, 24, 24);
    create_random_a_string( stock_dist_09, 24, 24);
    create_random_a_string( stock_dist_10, 24, 24);

    /* create ORIGINAL field */
    create_a_string_with_original( stock_data, 26, 50, 10 );
    s_ytd = s_order_cnt = s_remote_cnt = 0;

    numBytes = sprintf(Buffer, fmtStock,
        s_remote_cnt,
        stock_quant,
        s_order_cnt,
        s_ytd,
        stock_data,
        stock_dist_01,
        stock_dist_02,
        stock_dist_03,
        stock_dist_04,
        stock_dist_05,
        stock_dist_06,
        stock_dist_07,
        stock_dist_08,
        stock_dist_09,
        stock_dist_10,
        stock_num,
        ware_num);

    rc = GenericWrite(&hnd, Buffer, numBytes);
    if (rc != 0) { goto stock_done; }

} /* end for... */
} /* end for... */

rc = GenericClose(&hnd);

stock_done:

timestamp2 = current_time();
elapsed = timestamp2 - timestamp1;
if (rc == 0) {
    fprintf(stdout, "\nSTOCK table generated in %8.2f seconds.\n\n", elapsed);
    fflush(stdout);
} else {
    fprintf(stderr, "\nSTOCK table FAILED at (S %d W %d) after %8.2f
seconds.\n\n", stock_num, ware_num, elapsed);
    fflush(stderr);
}
}
}

/*-----*/
/* generate warehouse table */
/*-----*/
void gen_ware_tbl( void )
{

```

```

sqlint32 ware_num = 0 ;
char ware_name[11] ;
char ware_street_1[21] ;
char ware_street_2[21] ;
char ware_city[21] ;
char ware_state[3] ;
char ware_zip[10] ;
sqlint32 ware_tax ;
sqlint64 ware_YTD ;

IOH_NUM numBytes;
ioHandle hnd;
char Buffer[1024];

timestamp1 = current_time();

rc = GenericOpen(&hnd, outtype1, outname1);
if (rc != 0) { goto ware_done; }

for (ware_num = ware_start; ware_num <= ware_end; ware_num++)
{
    if (!quiet_mode && ((ware_num % 500) == 0)) {
        fprintf(stdout, "Warehouse #%d\n", ware_num);
        fflush(stdout);
    }

    create_random_a_string( ware_name, 6,10); /* create name */
    create_random_a_string( ware_street_1, 10,20); /* create street 1 */
    create_random_a_string( ware_street_2, 10,20); /* create street 2 */
    create_random_a_string( ware_city, 10,20); /* create city */
    create_random_a_string( ware_state, 2,2); /* create state */
    create_random_n_string( ware_zip, 4,4); /* create zip */
    strcat(ware_zip, "11111");

    ware_tax = rand_integer(0, 2000);
    ware_YTD = 30000000;

    numBytes = sprintf(Buffer, fmtWare,
        ware_name,
        ware_street_1,
        ware_street_2,
        ware_city,
        ware_state,
        ware_zip,
        ware_tax,
        ware_YTD,
        ware_num);

    rc = GenericWrite(&hnd, Buffer, numBytes);
    if (rc != 0) { goto ware_done; }
} /* end for */

rc = GenericClose(&hnd);

ware_done:

timestamp2 = current_time();
elapsed = timestamp2 - timestamp1;
if (rc == 0) {
    fprintf(stdout, "\nWAREHOUSE table generated in %8.2f
seconds.\n\n", elapsed);
    fflush(stdout);
} else {
    fprintf(stderr, "\nWAREHOUSE table FAILED at (W %d) after %8.2f
seconds.\n\n", ware_num, elapsed);
    fflush(stderr);
}

```

```

}
/*-----*/
/* generate dist table */
/*-----*/
void gen_dist_tbl( void )
{
    sqlint32 ware_num = 0 ;
    sqlint32 dist_num = 0 ;
    char dist_name[11];
    char dist_street_1[21];
    char dist_street_2[21];
    char dist_city[21];
    char dist_state[3];
    char dist_zip[10];
    sqlint32 dist_tax;
    sqlint32 next_o_id;
    sqlint64 dist_YTD;

    IOH_NUM numBytes;
    ioHandle hnd;
    char Buffer[1024];

    next_o_id = CUSTOMERS_PER_DISTRICT + 1;
    timestamp1 = current_time();

    rc = GenericOpen(&hnd, outtype1, outname1);
    if (rc != 0) { goto dist_done; }

    for (ware_num = ware_start; ware_num <= ware_end; ware_num++)
    {
        if (!quiet_mode) {
            fprintf(stdout, "DISTRICT for Warehouse #%d\n", ware_num);
            fflush(stdout);
        }
        for (dist_num = 1; dist_num <= DISTRICTS_PER_WAREHOUSE;
            dist_num++)
        {
            create_random_a_string( dist_name, 6,10); /* create name */
            create_random_a_string( dist_street_1, 10,20); /* create street 1 */
            create_random_a_string( dist_street_2, 10,20); /* create street 2 */
            create_random_a_string( dist_city, 10,20); /* create city */
            create_random_a_string( dist_state, 2,2); /* create state */
            create_random_n_string( dist_zip, 4,4); /* create zip */
            strcat(dist_zip, "11111");
            dist_tax = rand_integer(0, 2000);
            dist_YTD = 3000000;

            numBytes = sprintf(Buffer, fmtDist,
                next_o_id,
                dist_tax,
                dist_YTD,
                dist_name,
                dist_street_1,
                dist_street_2,
                dist_city,
                dist_state,
                dist_zip,
                dist_num,
                ware_num);

            rc = GenericWrite(&hnd, Buffer, numBytes);
            if (rc != 0) { goto dist_done; }

        } /* end for... */
    } /* end for... */

    rc = GenericClose(&hnd);

```

```

dist_done:

timestamp2 = current_time();
elapsed = timestamp2 - timestamp1;
if (rc == 0) {
    fprintf(stdout, "\nDISTRICT table generated in %8.2f seconds.\n\n", elapsed);
    fflush(stdout);
} else {
    fprintf(stderr, "\nDISTRICT table FAILED at (W %d D %d) after %8.2f
seconds.\n\n", ware_num, dist_num, elapsed);
    fflush(stderr);
}
}

/*-----*/
/* generate customer table */
/*-----*/
void gen_cust_tbl( void )
{
    sqlint32 ware_num = 0 ;
    sqlint32 dist_num = 0 ;
    sqlint32 cust_num = 0 ;
    char cust_last[17];
    char cust_middle[3];
    char cust_first[17];
    char cust_street_1[21];
    char cust_street_2[21];
    char cust_city[21];
    char cust_state[3];
    char cust_zip[10];
    char cust_phone[17];
    char cust_credit[3];
    char cust_data[501];
    sqlint32 cust_discount;
    sqlint64 currtmstp;
    sqlint64 cust_balance;
    sqlint64 cust_YTD_payment;
    sqlint64 cust_credit_lim;

    IOH_NUM numBytes;
    ioHandle hnd;
    char Buffer[1024];
    int len, pos;

    timestamp1 = current_time();

    rc = GenericOpen(&hnd, outtype1, outname1);
    if (rc != 0) { goto cust_done; }

    strcpy(cust_middle, "OE");
    currtmstp = time(NULL);

    for (cust_num = 1; cust_num <= CUSTOMERS_PER_DISTRICT;
cust_num++)
    {
        if (!quiet_mode) {
            fprintf(stdout, "CUSTOMER #%d:\n", cust_num);
            fflush(stdout);
        }

        for (ware_num = ware_start; ware_num <= ware_end; ware_num++)
        {
            for (dist_num = 1; dist_num <= DISTRICTS_PER_WAREHOUSE;
dist_num++)
            {
                if (cust_num <= 1000) /* create last name */
                    create_random_last_name(cust_last, cust_num);

```

```

else /* create last name */
    create_random_last_name(cust_last, 0);
    create_random_a_string(cust_first, 8, 16); /* create first name */
    create_random_a_string(cust_street_1, 10, 20); /* create street 1 */
    create_random_a_string(cust_street_2, 10, 20); /* create street 2 */
    create_random_a_string(cust_city, 10, 20); /* create city */
    create_random_a_string(cust_state, 2, 2); /* create state */
    create_random_n_string(cust_zip, 4, 4); /* create zip */
    strcat(cust_zip, "11111");

    /* create phone number */
    create_random_n_string(cust_phone, 16, 16);
    if (rand_integer(1, 100) <= 10)
        strcpy(cust_credit, "BC");
    else
        strcpy(cust_credit, "GC");

    /* create discount rate */
    cust_discount = rand_integer(0, 5000);

    /* create customer data */
    create_random_a_string(cust_data, 300, 500);

    /* pad customer data (only for non-rtload) */
    if (using_rtload == 0) {
        for (pos = strlen(cust_data); pos < 500; pos++)
            cust_data[pos] = ' ';
        cust_data[500] = '\0';
    }

    cust_credit_lim = 5000000;
    cust_balance = -1000;
    cust_YTD_payment = 1000;

    if (cust_num == 1 && dist_num == 1 && ware_num == 1)
    {
        sprintf(cust_first, "C_LAST_LOAD=%d", C_C_LAST_LOAD);
    }

    numBytes = sprintf(Buffer, fmtCust,
        cust_num,
        cust_state,
        cust_zip,
        cust_phone,
        currtmstp,
        cust_credit_lim,
        cust_middle,
        cust_credit,
        cust_discount,
        cust_data,
        cust_last,
        cust_first,
        cust_street_1,
        cust_street_2,
        cust_city,
        dist_num,
        ware_num,
        0,
        cust_balance,
        cust_YTD_payment,
        1);

    rc = GenericWrite(&hnd, Buffer, numBytes);
    if (rc != 0) { goto cust_done; }

} /* end for district... */
} /* end for warehouse... */
} /* end for customer... */

```

```

rc = GenericClose(&hnd);

cust_done:

timestamp2 = current_time();
elapse = timestamp2 - timestamp1;
if (rc == 0) {
    fprintf(stdout, "\nCUSTOMER table generated in %8.2f
seconds.\n\n", elapse);
    fflush(stdout);
} else {
    fprintf(stderr, "\nCUSTOMER table FAILED at (W %d D %d C %d) after
%8.2f seconds.\n\n", ware_num, dist_num, cust_num, elapse);
    fflush(stderr);
}
}

/*-----*/
/* generate hist table */
/*-----*/

void gen_hist_tbl( void )
{
    sqlint32 ware_num = 0 ;
    sqlint32 dist_num = 0 ;
    sqlint32 cust_num = 0 ;
    char hist_data[25] ;
    sqlint64 currtmstmp;

    IOH_NUM numBytes;
    ioHandle hnd;
    char Buffer[1024];

    timestamp1 = current_time();

    rc = GenericOpen(&hnd, outtype1, outname1);
    if (rc != 0) { goto hist_done; }

    currtmstmp = time(NULL);

    for (ware_num = ware_start; ware_num <= ware_end; ware_num++)
    {
        if (!quiet_mode) {
            fprintf(stdout, "HISTORY for Warehouse #%d:\n", ware_num);
            fflush(stdout);
        }
        for (dist_num = 1; dist_num <= DISTRICTS_PER_WAREHOUSE;
dist_num++)
        {
            for (cust_num = 1; cust_num <= CUSTOMERS_PER_DISTRICT;
cust_num++)
            {
                /* create history data */
                create_random_a_string( hist_data, 12,24) ;

                numBytes = sprintf(Buffer, fmtHist,
                    cust_num,
                    dist_num,
                    ware_num,
                    dist_num,
                    ware_num,
                    currtmstmp,
                    1000,
                    hist_data);

                rc = GenericWrite(&hnd, Buffer, numBytes);
                if (rc != 0) { goto hist_done; }
            }
        }
    }
}

```

```

} /* end for customer... */
} /* end for district... */
} /* end for warehouse... */

rc = GenericClose(&hnd);

hist_done:

timestamp2 = current_time();
elapse = timestamp2 - timestamp1;
if (rc == 0) {
    fprintf(stdout, "\nHISTORY table generated in %8.2f seconds.\n\n", elapse);
    fflush(stdout);
} else {
    fprintf(stderr, "\nHISTORY table FAILED at (W %d D %d C %d) after
%8.2f seconds.\n\n", ware_num, dist_num, cust_num, elapse);
    fflush(stderr);
}
}

/*-----*/
/* generate nu_ord table */
/*-----*/

void gen_nu_ord_tbl( void )
{
    sqlint32 ware_num = 0 ;
    sqlint32 dist_num = 0 ;
    sqlint32 nu_ord_id = 0 ;
    int nu_ord_hi ;

    IOH_NUM numBytes;
    ioHandle hnd;
    char Buffer[1024];

    /* compute maximum and minimum
order numbers for this
district */
    nu_ord_hi = CUSTOMERS_PER_DISTRICT -
NU_ORDERS_PER_DISTRICT + 1;
    if (nu_ord_hi < 0) {
        nu_ord_hi = CUSTOMERS_PER_DISTRICT -
(CUSTOMERS_PER_DISTRICT / 3) + 1;
        fprintf(stderr, "\n**** WARNING **** NU_ORDERS_PER_DISTRICT is
> CUSTOMERS_PER_DISTRICT\n");
        fprintf(stderr, "                Check the values in file lval.h\n");
        fprintf(stderr, "                Loading New-Order with 1/3 of
CUSTOMERS_PER_DISTRICT\n");
    }

    timestamp1 = current_time();

    rc = GenericOpen(&hnd, outtype1, outname1);
    if (rc != 0) { goto neword_done; }

    for (nu_ord_id = nu_ord_hi;
        nu_ord_id <= CUSTOMERS_PER_DISTRICT;
        nu_ord_id++)
    {
        if (!quiet_mode) {
            fprintf(stdout, "NEW_ORDER for Customer #%d:\n", nu_ord_id);
            fflush(stdout);
        }
        for (ware_num = ware_start; ware_num <= ware_end; ware_num++)
        {
            for (dist_num = 1; dist_num <= DISTRICTS_PER_WAREHOUSE;
dist_num++)
            {
                numBytes = sprintf(Buffer, fmtNewOrd,

```

```

        nu_ord_id,
        dist_num,
        ware_num);

    rc = GenericWrite(&hnd, Buffer, numBytes);
    if (rc != 0) { goto neword_done; }

    } /* end for... */
} /* end for... */
} /* end for... */

rc = GenericClose(&hnd);

neword_done:

    timestamp2 = current_time();
    elapse = timestamp2 - timestamp1;
    if (rc == 0) {
        fprintf(stdout, "\nNEW_ORDER table generated in %8.2f
seconds.\n\n", elapse);
        fflush(stdout);
    } else {
        fprintf(stderr, "\nNEW_ORDER table FAILED at (W %d D %d O %d) after
%8.2f seconds.\n\n", ware_num, dist_num, nu_ord_id, elapse);
        fflush(stderr);
    }
}

/*-----*/
/* generate order and order_line tables */
/*-----*/
void gen_ordr_tbl( void )
{
    sqlint32 ware_num = 0 ;
    sqlint32 dist_num = 0 ;
    sqlint32 cust_num = 0 ;
    sqlint32 ord_num = 0 ;
    sqlint32 ord_r_carrier_id;
    sqlint32 ord_r_ol_cnt;
    sqlint32 oline_ol_num;
    sqlint32 oline_item_num;

    sqlint32 oline_amount;
    char oline_dist_info[25];
    sqlint64 nulltmstmp = 0;
    sqlint64 currtmstmp;

    IOH_NUM numBytes;
    ioHandle hnd1, hnd2;
    char Buffer[1024];

    oline_dist_info[24] = '\0';

    timestamp1 = current_time();

    rc1 = GenericOpen(&hnd1, outtype1, outname1);
    if (rc1 != 0) { goto ool_done; }
    rc2 = GenericOpen(&hnd2, outtype2, outname2);
    if (rc2 != 0) { goto ool_done; }

    currtmstmp = time(NULL);

    for (ware_num = ware_start; ware_num <= ware_end; ware_num++)
    {
        if (!quiet_mode) {
            fprintf(stdout, "ORDERS & ORDER_LINE for Warehouse #%d\n",
ware_num);
            fflush(stdout);
        }
    }
}

}
for (dist_num = 1; dist_num <= DISTRICTS_PER_WAREHOUSE;
dist_num++)
{
    if (!quiet_mode) {
        fprintf(stdout, "District #%d\t", dist_num);
        fflush(stdout);
    }

    seed_1_3000();

    for (ord_num = 1; ord_num <= CUSTOMERS_PER_DISTRICT;
ord_num++)
    {
        if (ord_num < 2101)
            ord_carrier_id = rand_integer( 1, 10 );
        else
            ord_carrier_id = 0;

        cust_num = random_1_3000();
        ord_ol_cnt =
rand_integer(MIN_OL_PER_ORDER, MAX_OL_PER_ORDER);

        numBytes = sprintf(Buffer, fmtOrdr,
            cust_num,
            currtmstmp,
            ord_r_carrier_id,
            ord_r_ol_cnt,
            1,
            ord_num,
            ware_num,
            dist_num);

        rc1 = GenericWrite(&hnd1, Buffer, numBytes);
        if (rc1 != 0) { goto ool_done; }

        for ( oline_ol_num = 1; oline_ol_num <= ord_r_ol_cnt; oline_ol_num++)
        {
            oline_item_num = rand_integer(1, ITEMS);
            create_random_a_string( oline_dist_info, 24, 24 );

            numBytes = sprintf(Buffer, fmtOLine,
                ((ord_num < 2101) ? currtmstmp : nulltmstmp),
                ((ord_num < 2101) ? 0 : rand_integer(1,999999)),
                oline_item_num,
                ware_num,
                5,
                oline_dist_info,
                ord_num,
                dist_num,
                ware_num,
                oline_ol_num);

            rc2 = GenericWrite(&hnd2, Buffer, numBytes);
            if (rc2 != 0) { goto ool_done; }

        } /* for order_line */
    } /* for order */
} /* for dist */
} /* for ware */

rc1 = GenericClose(&hnd2);
rc2 = GenericClose(&hnd1);

ool_done:

    timestamp2 = current_time();
    elapse = timestamp2 - timestamp1;

```

```

if (rc1 == 0 && rc2 == 0) {
    fprintf(stdout, "\nORDERS & ORDER_LINE table(s) generated in %8.2f
seconds.\n\n", elapse);
    fflush(stdout);
} else {
    fprintf(stderr, "\nORDERS & ORDER_LINE table(s) FAILED at (W %d D
%d O %d OL %d) after %8.2f seconds.\n\n", ware_num, dist_num, ord_num,
oline_ol_num, elapse);
    fflush(stderr);
}
}

```

```

// This routine will initialize the printf format strings and replace the
// delimiter with the one provided. The pipe symbol is the default.
void InitFormatStrings(char delim)
{

```

```

char *p;

// Check if Using Default Delimiter
if (delim == '|') return;

```

```

// Replace Delimiters
while (p = strchr(fmtWare, '|')) { *p = delim; }
while (p = strchr(fmtDist, '|')) { *p = delim; }
while (p = strchr(fmtItem, '|')) { *p = delim; }
while (p = strchr(fmtStock, '|')) { *p = delim; }
while (p = strchr(fmtCust, '|')) { *p = delim; }
while (p = strchr(fmtHist, '|')) { *p = delim; }
while (p = strchr(fmtOrd, '|')) { *p = delim; }
while (p = strchr(fmtOLine, '|')) { *p = delim; }
while (p = strchr(fmtNewOrd, '|')) { *p = delim; }
}

```

```

void ScalingReport(void)
{
    /* Print Scaling Values */
    fprintf(stdout, "Scaling Values in Use\n");
    fprintf(stdout, "-----\n");
    fprintf(stdout, "Warehouses:      %d\n", WAREHOUSES);
    fprintf(stdout, "Districts/Warehouse: %d\n",
DISTRICTS_PER_WAREHOUSE);
    fprintf(stdout, "Customers/District:  %d\n",
CUSTOMERS_PER_DISTRICT);
    fprintf(stdout, "Items:                %d\n", ITEMS);
    fprintf(stdout, "Stock/Warehouse:     %d\n", STOCK_PER_WAREHOUSE);
    fprintf(stdout, "Min Order Lines/Order: %d\n", MIN_OL_PER_ORDER);
    fprintf(stdout, "Max Order Lines/Order: %d\n", MAX_OL_PER_ORDER);
    fprintf(stdout, "New Orders/District: %d\n",
NU_ORDERS_PER_DISTRICT);
    fprintf(stdout, "-----\n");
}

```

dbgen\makefile

```

#####
#####
## Licensed Materials - Property of IBM
##
## Governed under the terms of the International
## License Agreement for Non-Warranted Sample Code.
##
## (C) COPYRIGHT International Business Machines Corp. 1996 - 2005
## All Rights Reserved.
##
## US Government Users Restricted Rights - Use, duplication or
## disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

```

```

#####
#####
# Makefile - Build gendata tool
#
!include $(TPCC_ROOT)/Makefile.config
#
#####
#####
# Preprocessor, Compiler and Linker Flags
#
#####
#####
INCLUDES =      -I$(TPCC_SQLLIB)$(SLASH)include
               -I$(TPCC_ROOT)$(SLASH)include
CFLAGS =        $(INCLUDES) $(CFLAGS_OS) -DLINT_ARGS
               -DSQLA_NOLINES \
               -D$(DB2EDITION) -D$(DB2VERSION)
$(CFLAGS_DEBUG)
LDLFLAGS =      $(LDLFLAGS_EXEC) $(LDLFLAGS_LIB)
#
#####
#####
# File Collections
#
#####
#####
OBJS =          tpcrnd$(OBJEXT) \
               $(TPCC_ROOT)/Src.Common/tpcmisc$(OBJEXT)
OBJ_EEE =       $(TPCC_ROOT)/Src.Common/tpceclwh$(OBJEXT)
EXEC =          gendata$(BINEXT)
#
#####
#####
# End-User Targets
#
#####
#####
all:            $(EXEC)
clean:         - $(ERASE) *$(OBJEXT) $(EXEC)
#
#####
#####
# Build Rules
#
#####
#####
.SUFFIXES:
.SUFFIXES:     $(OBJEXT) .c
$(EXEC):
               $(LD_EXEC) $(LDLFLAGS) $(OBJS) *$(OBJEXT)
               $(LDLFLAGS_OUT)$@

```



```

#
#####
#####
# Dependencies
#
#####
#####

# Link Dependencies
gendata$(BINEXT): $(OBJS) gendata$(OBJEXT)

# Build Dependencies
# Source
gendata$(OBJEXT): gendata.c

# Headers
gendata.c: $(TPCC_ROOT)/include/tpccrnd.h $(TPCC_ROOT)/include/lval.h

dbgen\tpccrnd.c

/*****
*****
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2005
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****/

/*
 * tpccrnd.c - Random generation functions for TPC-C
 */

#include <windows.h>
#include <stdio.h>
#include <string.h>
#include "db2tpcc.h"
#include "tpccmisc.h"
#include "lval.h"

static char tbl_cust[CUSTOMERS_PER_DISTRICT];

static char alnum[] =

"0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ
WXYZ";

static char *last_name_parts[] =
{
"BAR",
"OUGHT",
"ABLE",
"PRI",
"PRES",
"ESE",
"ANTI",
"CALLY",
"ATION",
"EING"
};

```

```

/*
*****
*****
 * seed_1_3000
 *
 *
*****
*****
*/

void seed_1_3000( void )
{
int i;

for (i = 0; i < CUSTOMERS_PER_DISTRICT; i++) {
tbl_cust[i] = 0;
}
}

/*
*****
*****
 * random_1_3000
 *
 *
*****
*****
*/

int random_1_3000( void )
{
static int i;
static int x;

x = rand_integer(0, CUSTOMERS_PER_DISTRICT - 1);

for (i = 0; i < CUSTOMERS_PER_DISTRICT; i++)
{
if (tbl_cust[x] == 0)
{
tbl_cust[x] = 1;
return(x+1);
} else {
x++;
}
if (x == CUSTOMERS_PER_DISTRICT)
x=0;
}

printf("\nfatal error in random_1_3000 \n");
abort();
}

/*
*****
*****
 * initialize_random
*****
*****
*/

void initialize_random(void)
{
int t = current_time();

srand(t);
srandom(t);

```

```

}

/*
*****
*****
* create_random_a_string
*
* create a random alphanumeric string, of random length between lo and
* hi and place them in designated buffer. Routine returns the actual
* length.
*
* parameters
* -----
* lo end of acceptable length range
* hi end of acceptable length range
*
* output
* -----
* actual length
* random alphanumeric string
*
*****
*****
*/

int create_random_a_string( char *out_buffer, int length_lo, int length_hi )
{
    int i, actual_length ;

    actual_length = rand_integer( length_lo, length_hi ) ;

    for ( i = 0; i < actual_length; i++ )
    {
        out_buffer[i] = alnum[rand_integer( 0, 61 )] ;
    }
    out_buffer[actual_length] = '\0' ;

    return (actual_length);
}

/*
*****
*****
* create_random_n_string
*
* create a random numeric string, of random length between lo and
* hi and place them in designated buffer. Routine returns the actual
* length.
*
* parameters
* -----
* lo end of acceptable length range
* hi end of acceptable length range
*
* output
* -----
* actual length
* random numeric string
*
*****
*****
*/

int create_random_n_string( char *out_buffer, int length_lo, int length_hi )
{
    int i, actual_length ;

```

```

actual_length = rand_integer( length_lo, length_hi ) ;

for ( i = 0; i < actual_length; i++ )
{
    out_buffer[i] = (char)rand_integer( 48,57 ) ;
}
out_buffer[actual_length] = '\0' ;

return (actual_length);
}

/*
*****
*****
* NUrnd_val
*
* create a non-uniform random numeric value of type integer, of random
* value between lo and hi. Number is NOT placed in BUFFER, and IS
* simply RETURNED.
*
* Routine RETURNS the VALUE.
*
* parameters
* -----
* lo end of acceptable value range
* hi end of acceptable value range
*
* output
* -----
* random integer value RETURNED
*
*****
*****
*/

int NUrnd_val ( int A, int x, int y, int C )
{
    return((((rand_integer(0,A)|rand_integer(x,y))+C)%(y-x+1))+x);
}

/*
*****
*****
* rand_integer
*
* create a uniform random numeric value of type integer, of random
* value between lo and hi. Number is NOT placed in BUFFER, and IS
* simply RETURNED.
*
* Routine RETURNS the VALUE.
*
* parameters
* -----
* lo end of acceptable value range
* hi end of acceptable value range
*
* output
* -----
* random integer value RETURNED
*
*****
*****
*/

int rand_integer ( int val_lo, int val_hi )
{
    return((random()%(val_hi-val_lo+1))+val_lo);
}

```

```

}

/*
*****
*****
* create_a_string_with_original
*
* create a random alphanumeric string, of random length between lo and
* hi and place them in designated buffer. Routine returns the actual
* length.
*
* the word "ORIGINAL" is placed at a random location in the buffer at
* random, for a given percent of the records.
*
* percent_to_set must be an integer value from 0 to 100.
* if 0, no records will be set. If 100, all records will be set.
*
* CANNOT USE ON STRINGS OF LENGTH LESS THAN 8 ! LOWER
LIMIT MUST BE > 8 !
*
* parameters
* -----
* lo end of acceptable length range
* hi end of acceptable length range
* percentage of records to set to ORIGINAL
*
* output
* -----
* actual length
* random alphanumeric string with the word "ORIGINAL" is placed at a
* random location
*****
*****
*/

int create_a_string_with_original( char *out_buffer, int length_lo,
                                int length_hi, int percent_to_set )
{
    int actual_length, start_pos ;

    actual_length = create_random_a_string( out_buffer, length_lo, length_hi ) ;

    if ( rand_integer( 1, 100 ) <= percent_to_set )
    {
        start_pos = rand_integer( 0, actual_length-8 ) ;
        strncpy(out_buffer+start_pos,"ORIGINAL",8) ;
    }

    return (actual_length);
}

/*
*****
*****
* create_random_last_name
*
* parameters:
* out_buffer - target buffer for the generated last name
*
* description:
* create_random_last_name generates a random number from 0 to 999
* inclusive. a random name is generated by associating a random string
* with each digit of the generated number. the three strings are
* concatenated to generate the name
*
*/

```

```

*****
*****/

int create_random_last_name(char *out_buffer, int cust_num)
{
    int random_num;

    if (cust_num == 0)
        random_num = NUrnd_val( A_C_LAST, 0, 999, C_C_LAST_LOAD );
    else
        random_num = cust_num - 1;

    strcpy(out_buffer, last_name_parts[random_num / 100]);
    random_num %= 100;
    strcat(out_buffer, last_name_parts[random_num / 10]);
    random_num %= 10;
    strcat(out_buffer, last_name_parts[random_num]);

    return(strlen(out_buffer));
}

dbgen\include\db2tpcc.h

/*
*****
*****
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2005
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****
*****/

/*
* db2tpcc.h - Macros and Miscellany
*/

#ifndef _DB2TPCC_H
#define _DB2TPCC_H

#include <sys/types.h>
typedef __int16 int16_t;
typedef __int32 int32_t;
typedef __int64 int64_t;

#include "lval.h"

/*
*****
*****
** Transaction Return Codes (s_transtatus)
**
*****
*****

#define INVALID_ITEM      100
#define TRAN_OK           0
#define FATAL_SQLERROR   -1

/*
*****
*****
** Definition of Unused and Bad Items
**
*/

```

```

/*
*****
***** */
/* Define unused item ID to be 0. This allows the SUT to determine the */
/* number of items in the order as required by 2.4.1.3 and 2.4.2.2 since */
/* the assumption that any item with OL_I_ID = 0 is unused will be true. */
/* This in turn requires that the value used for an invalid item is */
/* equal to ITEMS + 1. */
/*
*****
***** */

#define INVALID_ITEM_ID (2 * ITEMS) + 1
#define UNUSED_ITEM_ID 0

#define MIN_WAREHOUSE 1
#define MAX_WAREHOUSE WAREHOUSES

/*****
*****/
/* NURand Constants */
/* C_C_LAST_RUN and C_C_LAST_LOAD must adhere to clause 2.1.6. */
/*
/* Analysis indicates that a C_LAST delta of 85 is optimal. */
/*****
*****/
#define C_C_LAST_RUN 88
#define C_C_LAST_LOAD 173
#define C_C_ID 319
#define C_OL_I_ID 3849
#define A_C_LAST 255
#define A_C_ID 1023
#define A_OL_I_ID 8191

/*****
*****/
/* Transaction Type Identifiers */
/*****
*****/

#define CLIENT_SQL 0
#define NEWORD_SQL 1
#define PAYMENT_SQL 2
#define ORDSTAT_SQL 3
#define DELIVERY_SQL 4
#define STOCKLEV_SQL 5

#define SPGENERAL_PAD 3
#define SPGENERAL_ADJUST sizeof(int16_t)

struct in_neword_struct {
    int16_t len;
    int16_t pad[SPGENERAL_PAD];
    struct in_items_struct {
        int32_t s_OL_I_ID;
        int32_t s_OL_SUPPLY_W_ID;
        int16_t s_OL_QUANTITY;
        int16_t pad1[3];
    } in_item[15];
    int64_t s_O_ENTRY_D_time; /* init by SUT */
    int32_t s_C_ID;
    int32_t s_W_ID;
    int16_t s_D_ID;
    int16_t s_O_OL_CNT; /* init by SUT */
    int16_t s_all_local;
    int16_t duplicate_items;
};

```

```

struct out_neword_struct {
    int16_t len;
    int16_t pad[SPGENERAL_PAD];
    struct items_struct {
        int32_t s_I_PRICE;
        int32_t s_OL_AMOUNT;
        int16_t s_S_QUANTITY;
        int16_t pad2;
        char s_I_NAME[25];
        char s_brand_generic;
    } item[15];
    int64_t s_O_ENTRY_D_time;
    int32_t s_W_TAX;
    int32_t s_D_TAX;
    int32_t s_C_DISCOUNT;
    int32_t s_total_amount;
    int32_t s_O_ID;
    int16_t s_O_OL_CNT;
    int16_t s_transtatus;
    int16_t deadlocks;
    char s_C_LAST[17];
    char s_C_CREDIT[3];
};

struct in_payment_struct {
    int16_t len;
    int16_t pad[SPGENERAL_PAD];
    int64_t s_H_DATE_time; /* init by SUT */
    int64_t s_H_AMOUNT;
    int32_t s_W_ID;
    int32_t s_C_W_ID;
    int32_t s_C_ID;
    int16_t s_C_D_ID;
    int16_t s_D_ID;
    char s_C_LAST[17];
};

struct out_payment_struct {
    int16_t len;
    int16_t pad[SPGENERAL_PAD];
    int64_t s_H_DATE_time;
    int64_t s_C_SINCE_time;
    int64_t s_C_CREDIT_LIM;
    int64_t s_C_BALANCE;
    int32_t s_C_DISCOUNT;
    int32_t s_C_ID;
    int16_t s_transtatus;
    int16_t deadlocks;
    char s_W_STREET_1[21];
    char s_W_STREET_2[21];
    char s_W_CITY[21];
    char s_W_STATE[3];
    char s_W_ZIP[10];
    char s_D_STREET_1[21];
    char s_D_STREET_2[21];
    char s_D_CITY[21];
    char s_D_STATE[3];
    char s_D_ZIP[10];
    char s_C_FIRST[17];
    char s_C_MIDDLE[3];
    char s_C_LAST[17];
    char s_C_STREET_1[21];
    char s_C_STREET_2[21];
    char s_C_CITY[21];
    char s_C_STATE[3];
    char s_C_ZIP[10];
    char s_C_PHONE[17];
    char s_C_CREDIT[3];
};

```

```

char s_C_DATA[201];
};

struct in_ordstat_struct {
int16_t len;
int16_t pad[SPGENERAL_PAD];
int32_t s_C_ID;
int32_t s_W_ID;
int16_t s_D_ID;
int16_t pad1[3];
char s_C_LAST[17];
};

struct out_ordstat_struct {
int16_t len;
int16_t pad[SPGENERAL_PAD];
int64_t s_C_BALANCE;
int64_t s_O_ENTRY_D_time;
int32_t s_C_ID;
int32_t s_O_ID;
int16_t s_O_CARRIER_ID;
int16_t s_ol_cnt;
int16_t pad1[2];
struct oitems_struct {
int64_t s_OL_DELIVERY_D_time;
int32_t s_OL_AMOUNT;
int32_t s_OL_I_ID;
int32_t s_OL_SUPPLY_W_ID;
int16_t s_OL_QUANTITY;
int16_t pad2;
} item[15];
int16_t s_transtatus;
int16_t deadlocks;
char s_C_FIRST[17];
char s_C_MIDDLE[3];
char s_C_LAST[17];
};

struct in_delivery_struct {
int16_t len;
int16_t pad[SPGENERAL_PAD];
int64_t s_O_DELIVERY_D_time; /* init by SUT */
int32_t s_W_ID;
int16_t s_O_CARRIER_ID;
};

struct out_delivery_struct {
int16_t len;
int16_t pad[SPGENERAL_PAD];
int32_t s_O_ID[10];
int16_t s_transtatus;
int16_t deadlocks;
};

struct in_stocklev_struct {
int16_t len;
int16_t pad[SPGENERAL_PAD];
int32_t s_threshold;
int32_t s_W_ID;
int16_t s_D_ID;
};

struct out_stocklev_struct {
int16_t len;
int16_t pad[SPGENERAL_PAD];
int32_t s_low_stock;
int16_t s_transtatus;
int16_t deadlocks;
};
};

/*
***** */
/* Transaction Prototypes */
/*
***** */

#ifdef __cplusplus
extern "C" {
#endif

extern int neword_sql(struct in_neword_struct*, struct out_neword_struct*);
extern int payment_sql(struct in_payment_struct*, struct out_payment_struct*);
extern int ordstat_sql(struct in_ordstat_struct*, struct out_ordstat_struct*);
extern int delivery_sql(struct in_delivery_struct*, struct out_delivery_struct*);
extern int stocklev_sql(struct in_stocklev_struct*, struct out_stocklev_struct*);

#ifdef __cplusplus
}
#endif

/*
***** */
/* DB2 Connect/Disconnect & Thread Context Wrappers */
/*
***** */

#ifdef __cplusplus
extern "C" {
#endif

extern int connect_to_TM(char*);
extern int connect_to_TM_auth(char*, char*, char*);
extern int disconnect_from_TM(void);

extern int create_context(void);
extern int destroy_context(void);
extern int get_context(void**);
extern int attach_context(void*);
extern int detach_context(void*);

#ifdef __cplusplus
}
#endif

#endif // __DB2TPCC_H

dbgen\include\val.h

#ifdef __LVAL_H
#define __LVAL_H
#define WAREHOUSES 20000
#define DISTRICTS_PER_WAREHOUSE 10
#define CUSTOMERS_PER_DISTRICT 3000
#define ITEMS 100000
#define STOCK_PER_WAREHOUSE 100000
#define MIN_OL_PER_ORDER 5
#define MAX_OL_PER_ORDER 15
#define NU_ORDERS_PER_DISTRICT 900
#endif // __LVAL_H

```

dbgen\include\platform.h

```
#define IOH_PIPE 2
#define IOH_FILE_APPEND 3

#define IOH_ERRMSG(hnd, msg) \
if (rc != 0) { \
    fprintf(stderr, "Error %d %s fd %d (%d, %s)\n", GEN_ERRCODE, msg, \
        hnd->fd, hnd->type, hnd->name); \
    return rc; \
}

struct _ioh {
    IOH_HND fd;
    int type;
    char *name;
};

typedef struct _ioh ioHandle;

/*
*****
** Generic I/O Routine Prototypes **
**
*****
int GenericOpen(ioHandle *hnd, int type, char *name);
int GenericWrite(ioHandle *hnd, char *Buffer, unsigned int numBytes);
int GenericClose(ioHandle *hnd);

/*
*****
** Generic I/O Routines **
**
*****
int GenericOpen(ioHandle *hnd, int type, char *name)
{
    int rc = 0;

    IOH_INIT(hnd, type, name)

    IOH_CREATE(hnd)
    IOH_ERRMSG(hnd, "creating")

    IOH_OPEN(hnd)
    IOH_ERRMSG(hnd, "opening")

    return rc;
}

int GenericWrite(ioHandle *hnd, char *Buffer, unsigned int numBytes)
{
    int rc = 0;
    int numBytesWritten = -1;

    IOH_WRITE(hnd, Buffer, numBytes, numBytesWritten)
    IOH_ERRMSG(hnd, "writing")
    if (numBytes != numBytesWritten) {
        fprintf(stderr, "Truncated data writing to fd %d (%d, %s)\n", hnd->fd,
            hnd->type, hnd->name);
        rc = -1;
    }
    return rc;
}
```

```
int GenericClose(ioHandle *hnd)
{
    int rc = 0;

    IOH_FLUSH(hnd)
    IOH_ERRMSG(hnd, "flushing")

    IOH_CLOSE(hnd)
    IOH_ERRMSG(hnd, "closing")

    IOH_DELETE(hnd)
    IOH_ERRMSG(hnd, "deleting")

    return rc;
}

#endif // __PLATFORM_H
```

dbgen\include\tpccrnd.h

```
/*
*****
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2005
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****
*/

/*
* tpccrnd.h - Random generation functions for TPC-C
*/

#ifndef __TPCCRND_H
#define __TPCCRND_H

void initialize_random(void);
int rand_integer( int val_lo, int val_hi );
int NUrnd_val( int A, int val_lo, int val_hi, int C );

void seed_1_3000( void );
int random_1_3000( void );

int create_random_a_string( char *out_buffer,
    int length_lo,
    int length_hi );
int create_random_n_string( char *out_buffer,
    int length_lo,
    int length_hi );
int create_a_string_with_original( char *out_buffer,
    int length_lo,
    int length_hi,
    int percent_to_set );
int create_random_last_name(char *out_buffer, int cust_num);

#endif // __TPCCRND_H
```

dbgen\makefile.config

```
#####
#####
## Licensed Materials - Property of IBM
##
## Governed under the terms of the International
## License Agreement for Non-Warranted Sample Code.
##
## (C) COPYRIGHT International Business Machines Corp. 1996 - 2005
## All Rights Reserved.
##
## US Government Users Restricted Rights - Use, duplication or
## disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
#####
#####

#
# Makefile.config - NT/Win2000 Makefile Configuration
#

# Make Configuration (MSVC)
MAKE=nmake.exe

# Compiler Configuration (MSVC).
# CFLAGS_DEBUG may be set to "-Zi -Od", "-DDEBUGIT" "-Zi -Od
-DDEBUGIT" or left blank
CC=cl.exe
CFLAGS_OS=-DSQLWINT -MT -GS- -DWIN64 -J -Zp8
-DREG_KIT_METHOD
CFLAGS_OUT=/Fo
CFLAGS_DEBUG=

# Linker Configuration (MSVC)
LD_EXEC=link.exe
LD_STORP=link.exe
LDFLAGS_EXEC=
LDFLAGS_SHLIB=/DLL
LDFLAGS_STORP=$(LDFLAGS_SHLIB) /DEF:rpctpc.def
LDFLAGS_LIB=/LIBPATH:$(TPCC_SQLLIB)\lib
/LIBPATH:"C:\MsSDKx64\Lib\AMD64" db2api.lib WinMM.lib
LDFLAGS_OUT=/OUT:

# Library Configuration
AR=lib.exe
ARFLAGS=
ARFLAGS_LIB=
ARFLAGS_OUT=/OUT:

# OS Commands
ERASE=del /F
ERASEDIR=rmdir /S
MOVE=MOVE
COPY=COPY

# OS File Extensions & Path Separator
OBJEXT=.obj
LIBEXT=.lib
SHLIBEXT=.dll
BINEXT=.exe
SLASH=\\
CMDSEP=&

dbgen\Src.Common\makefile

#####
#####
## Licensed Materials - Property of IBM
##
```

```
## Governed under the terms of the International
## License Agreement for Non-Warranted Sample Code.
##
## (C) COPYRIGHT International Business Machines Corp. 1996 - 2005
## All Rights Reserved.
##
## US Government Users Restricted Rights - Use, duplication or
## disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
#####
#####

#
# Makefile - Makefile for Src.Common
#

!include $(TPCC_ROOT)/Makefile.config

#
#####
#####
# Preprocessor, Compiler and Linker Flags
#
#####
#####

BND_OPTS =      GRANT PUBLIC \
                MESSAGES $*.bnd.msg
PRP_OPTS =      BINDFILE \
                OPTLEVEL 1 \
                ISOLATION RR \
                MESSAGES $*.prep.msg \
                LEVEL $(TPCC_VERSION) \
                NOLINEMACRO

INCLUDES =      -I$(TPCC_SQLLIB)\$(SLASH)include
                -I$(TPCC_ROOT)\$(SLASH)include

CFLAGS =        $(CFLAGS_OS) $(CFLAGS_DEBUG) $(INCLUDES) \
                -DSQLA_NOLINES -D$(DB2EDITION)
                -D$(DB2VERSION) \
                -D$(TPCC_SPTYPE)

UTIL_OBJ =      tpcmisc$(OBJEXT) tpcdbg$(OBJEXT)
tpccctx$(OBJEXT)

#
#####
#####
# User Targets
#
#####
#####

all:            connect $(UTIL_OBJ) disconnect

clean:
                - $(ERASE) *$(OBJEXT) *.bnd *.msg tpcctx.c

#
#####
#####
# Helper Targets
#
#####
#####

connect:
                - db2 connect to $(TPCC_DBNAME)
```

```

disconnect:
    - db2 connect reset
    - db2 terminate

rebind:
    db2 bind tpcctx.bnd $(BND_OPTS)

#
#####
#####
# Build Rules
#
#####
#####

.SUFFIXES:
.SUFFIXES: $(OBJEXT) .c .sqc

.sqc.c:
    @echo "Prepping $*.sqc"
    -db2 prep $*.sqc $(PRP_OPTS)
    @echo "Binding $*.bnd"
    db2 bind $*.bnd $(BND_OPTS)

#
#####
#####
# Dependencies
#
#####
#####

# Source
tpccdbg$(OBJEXT): tpcdbg.c
tpccctx$(OBJEXT): tpcctx.c
tpccmisc$(OBJEXT): tpccmisc.c

# Headers
tpccdbg.c: $(TPCC_ROOT)/include/db2tpcc.h

dbgen\Src.Common\tpccmisc.c

/*****
*****
** Licensed Materials - Property of IBM
**
** Governed under the terms of the International
** License Agreement for Non-Warranted Sample Code.
**
** (C) COPYRIGHT International Business Machines Corp. 1996 - 2005
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****/

/*
 * tpccmisc.c - Miscellaneous routines
 */

#include <windows.h>

#define RAND_A 16807
#define RAND_M 2147483647
#define RAND_M1 2147483646
#define RAND_MD 2147483647.0

```

```

#define RAND_Q 127773
#define RAND_R 2836

static int seed = 1;
static int seedflag = 0;

void srandom(int);
int random(void);
double current_time_ms(void);
double current_time(void);

void srandom (int initial_seed)
{
    seed = initial_seed;
    if ((seed < 1) || (seed > RAND_M1)) seed = 1;
}

int random (void)
{
    int lo;
    int hi;
    int test;

    hi = seed / RAND_Q;
    lo = seed % RAND_Q;
    test = RAND_A * lo - RAND_R * hi;
    if (test > 0) seed = test;
    else seed = test + RAND_M;

    return (seed);
}

/* Current time in SECONDS, precision SECONDS */
double current_time(void)
{
    /* truncate fractional seconds -> seconds */
    return (double)((int)(current_time_ms()));
}

/* Current time in SECONDS, precision MILLISECONDS */
double current_time_ms(void)
{
    /* GetCurrentTime() returns ms */
    /* convert to fractional seconds */
    return (GetCurrentTime() / 1000);
}

dbgen\tpccenv.bat

@REM
*****
*****
@REM Licensed Materials - Property of IBM
@REM
@REM Governed under the terms of the International
@REM License Agreement for Non-Warranted Sample Code.
@REM
@REM (C) COPYRIGHT International Business Machines Corp. 1996 - 2005
@REM All Rights Reserved.
@REM
@REM US Government Users Restricted Rights - Use, duplication or
@REM disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
@REM
*****
*****
@REM
@REM tpccenv.bat - Windows Environment Setup

```



```
@REM

@REM The Kit Version
set TPCC_VERSION=CK041012

@REM The DB2 Instance Name (for DB2)
set DB2INSTANCE=DB2

@REM The OS being used (i.e. "UNIX", "WINDOWS")
set PLATFORM=WINDOWS

@REM The type of make command and slash used by the OS
@REM (i.e. UNIX - "/", WINDOWS - "\")
@REM These are referenced all over the kit.
set SLASH=\
set MAKE=nmake

set TPCC_SPTYPE=SPGENERAL

set DB2VERSION=v8

@REM The schema name is typically the SQL authorization ID (or username).
@REM This is required for runstats and EEE.
set TPCC_SCHEMA=%USERNAME%

@REM DB2 EE/EEE Configuration
set DB2EDITION=EE
set DB2NODE=0
set DB2NODES=1

@REM TPCC General Configuration
set HOME=C
set TPCC_DBNAME=TPCC
set TPCC_ROOT=c:\tpc-c.ibm
set TPCC_SQLLIB=c:\SQLLIB
set TPCC_RUNDATA=c:\tpccdata

@REM TPCC Debug Configuration
set TPCC_DEBUGDIR=c:\temp

@REM Specifies where stored procedures should be placed and if they should
@REM be fenced.
set TPCC_SPDIR=%TPCC_SQLLIB%\function
set TPCC_FENCED=NO
```

Appendix C: Tunable Parameters

IBM DB2 UDB

Database Manager Configuration

Database Manager Configuration

Node type = Database Server with local and remote clients

Database manager configuration release level = 0x0a00

Maximum total of files open (MAXTOTFILOP) = 16000

CPU speed (millisec/instruction) (CPUSPEED) = 3.739392e-007

Max number of concurrently active databases (NUMDB) = 1

Data Links support (DATALINKS) = NO

Federated Database System Support (FEDERATED) = NO

Transaction processor monitor name (TP_MON_NAME) =

Default charge-back account (DFT_ACCOUNT_STR) =

Java Development Kit installation path (JDK_PATH) = C:\SQLLIB\java\jdk

Diagnostic error capture level (DIAGLEVEL) = 1

Notify Level (NOTIFYLEVEL) = 1

Diagnostic data directory path (DIAGPATH) =

Default database monitor switches

Buffer pool (DFT_MON_BUFPOOL) = OFF

Lock (DFT_MON_LOCK) = OFF

Sort (DFT_MON_SORT) = OFF

Statement (DFT_MON_STMT) = OFF

Table (DFT_MON_TABLE) = OFF

Timestamp (DFT_MON_TIMESTAMP) = OFF

Unit of work (DFT_MON_UOW) = OFF

Monitor health of instance and databases (HEALTH_MON) = OFF

SYSADM group name (SYSADM_GROUP) =

SYSCTRL group name (SYSCTRL_GROUP) =

SYSMAINT group name (SYSMAINT_GROUP) =

SYSMON group name (SYSMON_GROUP) =

Client Userid-Password Plugin (CLNT_PW_PLUGIN) =

Client Kerberos Plugin (CLNT_KRB_PLUGIN) = IBMkrb5

Group Plugin (GROUP_PLUGIN) =

GSS Plugin for Local Authorization (LOCAL_GSSPLUGIN) =

Server Plugin Mode (SRV_PLUGIN_MODE) = UNFENCED

Server List of GSS Plugins (SRVCON_GSSPLUGIN_LIST) =

Server Userid-Password Plugin (SRVCON_PW_PLUGIN) =

Server Connection Authentication (SRVCON_AUTH) = NOT_SPECIFIED

Database manager authentication (AUTHENTICATION) = CLIENT

Cataloging allowed without authority (CATALOG_NOAUTH) = NO

Trust all clients (TRUST_ALLCLNTS) = YES

Trusted client authentication (TRUST_CLNTAUTH) = CLIENT

Bypass federated authentication (FED_NOAUTH) = NO

Default database path (DFTDBPATH) = C:

Database monitor heap size (4KB) (MON_HEAP_SZ) = 4096

Java Virtual Machine heap size (4KB) (JAVA_HEAP_SZ) = 1024

Audit buffer size (4KB) (AUDIT_BUF_SZ) = 0

Size of instance shared memory (4KB) (INSTANCE_MEMORY) = AUTOMATIC

Backup buffer default size (4KB) (BACKBUFSZ) = 1024

Restore buffer default size (4KB) (RESTBUFSZ) = 1024

Agent stack size (AGENT_STACK_SZ) = 16

Minimum committed private memory (4KB) (MIN_PRIV_MEM) = 32

Private memory threshold (4KB) (PRIV_MEM_THRESH) = 20000

Sort heap threshold (4KB) (SHEAPTHRES) = 10000

Directory cache support (DIR_CACHE) = YES

Application support layer heap size (4KB) (ASLHEAPSZ) = 15

Max requester I/O block size (bytes) (RQRIOBLK) = 4096

DOS requester I/O block size (bytes) (DOS_RQRIOBLK) = 4096

Query heap size (4KB) (QUERY_HEAP_SZ) = 1000

Workload impact by throttled utilities(UTIL_IMPACT_LIM) = 10

Priority of agents (AGENTPRI) = SYSTEM

Max number of existing agents (MAXAGENTS) = 560

Agent pool size (NUM_POOLAGENTS) = 0

Initial number of agents in pool (NUM_INITAGENTS) = 0

Max number of coordinating agents (MAX_COORDAGENTS) = MAXAGENTS
 Max no. of concurrent coordinating agents (MAXCAGENTS) = MAX_COORDAGENTS
 Max number of client connections (MAX_CONNECTIONS) = MAX_COORDAGENTS
 Keep fenced process (KEEPFENCED) = YES
 Number of pooled fenced processes (FENCED_POOL) = MAX_COORDAGENTS
 Initial number of fenced processes (NUM_INITFENCED) = 0
 Index re-creation time and redo index build (INDEXREC) = RESTART
 Transaction manager database name (TM_DATABASE) = 1ST_CONN
 Transaction resync interval (sec) (RESYNC_INTERVAL) = 180
 SPM name (SPM_NAME) =
 SPM log size (SPM_LOG_FILE_SZ) = 256
 SPM resync agent limit (SPM_MAX_RESYNC) = 20
 SPM log path (SPM_LOG_PATH) =
 NetBIOS Workstation name (NNAME) =
 TCP/IP Service name (SVCENAME) =
 Discovery mode (DISCOVER) = SEARCH
 Discover server instance (DISCOVER_INST) = ENABLE
 Maximum query degree of parallelism (MAX_QUERYDEGREE) = ANY
 Enable intra-partition parallelism (INTRA_PARALLEL) = NO
 No. of int. communication buffers(4KB)(FCM_NUM_BUFFERS) = 1024
 Number of FCM request blocks (FCM_NUM_RQB) = AUTOMATIC
 Number of FCM connection entries (FCM_NUM_CONNECT) = AUTOMATIC
 Number of FCM message anchors (FCM_NUM_ANCHORS) = AUTOMATIC

Database Configuration

Database Configuration for Database tpcc

Database configuration release level = 0x0a00
 Database release level = 0x0a00
 Database territory = US
 Database code page = 1252
 Database code set = IBM-1252
 Database country/region code = 1

Database collating sequence = BINARY
 Alternate collating sequence (ALT_COLLATE) =
 Database page size = 4096
 Dynamic SQL Query management (DYN_QUERY_MGMT) = DISABLE
 Discovery support for this database (DISCOVER_DB) = ENABLE
 Default query optimization class (DFT_QUERYOPT) = 5
 Degree of parallelism (DFT_DEGREE) = 1
 Continue upon arithmetic exceptions (DFT_SQLMATHWARN) = NO
 Default refresh age (DFT_REFRESH_AGE) = 0
 Default maintained table types for opt (DFT_MTTB_TYPES) = SYSTEM
 Number of frequent values retained (NUM_FREQVALUES) = 10
 Number of quantiles retained (NUM_QUANTILES) = 20
 Backup pending = NO
 Database is consistent = NO
 Rollforward pending = NO
 Restore pending = NO
 Multi-page file allocation enabled = YES
 Log retain for recovery status = RECOVERY
 User exit for logging status = NO
 Data Links Token Expiry Interval (sec) (DL_EXPINT) = 60
 Data Links Write Token Init Expiry Intvl(DL_WT_IEXPINT) = 60
 Data Links Number of Copies (DL_NUM_COPIES) = 1
 Data Links Time after Drop (days) (DL_TIME_DROP) = 1
 Data Links Token in Uppercase (DL_UPPER) = NO
 Data Links Token Algorithm (DL_TOKEN) = MAC0
 Database heap (4KB) (DBHEAP) = 8192
 Size of database shared memory (4KB) (DATABASE_MEMORY) = AUTOMATIC
 Catalog cache size (4KB) (CATALOGCACHE_SZ) = (MAXAPPLS*4)
 Log buffer size (4KB) (LOGBUFSZ) = 3000
 Utilities heap size (4KB) (UTIL_HEAP_SZ) = 5000
 Buffer pool size (pages) (BUFFPAGE) = 250
 Extended storage segments size (4KB) (ESTORE_SEG_SZ) = 16000

Number of extended storage segments (NUM_ESTORE_SEGS) = 0
 Max storage for lock list (4KB) (LOCKLIST) = 100000
 Max size of appl. group mem set (4KB) (APPGROUP_MEM_SZ) = 30000
 Percent of mem for appl. group heap (GROUPHEAP_RATIO) = 70
 Max appl. control heap size (4KB) (APP_CTL_HEAP_SZ) = 128
 Sort heap thres for shared sorts (4KB) (SHEAPTHRES_SHR) = (SHEAPTHRES)
 Sort list heap (4KB) (SORTHEAP) = 10000
 SQL statement heap (4KB) (STMTHEAP) = 4096
 Default application heap (4KB) (APPLHEAPSZ) = 328
 Package cache size (4KB) (PCKCACHESZ) = 1000
 Statistics heap size (4KB) (STAT_HEAP_SZ) = 4384
 Interval for checking deadlock (ms) (DLCHKTIME) = 3000
 Percent. of lock lists per application (MAXLOCKS) = 100
 Lock timeout (sec) (LOCKTIMEOUT) = -1

 Changed pages threshold (CHNGPGS_THRESH) = 99
 Number of asynchronous page cleaners (NUM_IOCLEANERS) = 8
 Number of I/O servers (NUM_IOSERVERS) = 8
 Index sort flag (INDEXSORT) = YES
 Sequential detect flag (SEQDETECT) = YES
 Default prefetch size (pages) (DFT_PREFETCH_SZ) = AUTOMATIC
 Track modified pages (TRACKMOD) = OFF
 Default number of containers = 1
 Default tablespace extentsize (pages) (DFT_EXTENT_SZ) = 32
 Max number of active applications (MAXAPPLS) = 560
 Average number of active applications (AVG_APPLS) = 1
 Max DB files open per application (MAXFILOP) = 800
 Log file size (4KB) (LOGFILSIZ) = 256000
 Number of primary log files (LOGPRIMARY) = 31
 Number of secondary log files (LOGSECOND) = 0
 Changed path to log files (NEWLOGPATH) =
 Path to log files = \\.\M:
 Overflow log path (OVERFLOWLOGPATH) =
 Mirror log path (MIRRORLOGPATH) =

First active log file = S0000152.LOG
 Block log on disk full (BLK_LOG_DSK_FUL) = NO
 Percent of max active log space by transaction (MAX_LOG) = 0
 Num. of active log files for 1 active UOW (NUM_LOG_SPAN) = 0
 Group commit count (MINCOMMIT) = 1
 Percent log file reclaimed before soft chkpt (SOFTMAX) = 1670
 Log retain for recovery enabled (LOGRETAIN) = RECOVERY
 User exit for logging enabled (USEREXIT) = OFF
 HADR database role = STANDARD
 HADR local host name (HADR_LOCAL_HOST) =
 HADR local service name (HADR_LOCAL_SVC) =
 HADR remote host name (HADR_REMOTE_HOST) =
 HADR remote service name (HADR_REMOTE_SVC) =
 HADR instance name of remote server (HADR_REMOTE_INST) =
 HADR timeout value (HADR_TIMEOUT) = 120
 HADR log write synchronization mode (HADR_SYNCMODE) = NEARSYNC
 First log archive method (LOGARCHMETH1) = LOGRETAIN
 Options for logarchmeth1 (LOGARCHOPT1) =
 Second log archive method (LOGARCHMETH2) = OFF
 Options for logarchmeth2 (LOGARCHOPT2) =
 Failover log archive path (FAILARCHPATH) =
 Number of log archive retries on error (NUMARCHRETRY) = 5
 Log archive retry Delay (secs) (ARCHRETRYDELAY) = 20
 Vendor options (VENDOROPT) =
 Auto restart enabled (AUTORESTART) = ON
 Index re-creation time and redo index build (INDEXREC) = SYSTEM (RESTART)
 Log pages during index build (LOGINDEXBUILD) = OFF
 Default number of loadrec sessions (DFT_LOADREC_SES) = 1
 Number of database backups to retain (NUM_DB_BACKUPS) = 12
 Recovery history retention (days) (REC_HIS_RETENTN) = 366
 TSM management class (TSM_MGMTCLASS) =
 TSM node name (TSM_NODENAME) =
 TSM owner (TSM_OWNER) =

TSM password (TSM_PASSWORD) =

Automatic maintenance (AUTO_MAINT) = OFF

Automatic database backup (AUTO_DB_BACKUP) = OFF

Automatic table maintenance (AUTO_TBL_MAINT) = OFF

Automatic runstats (AUTO_RUNSTATS) = OFF

Automatic statistics profiling (AUTO_STATS_PROF) = OFF

Automatic profile updates (AUTO_PROF_UPD) = OFF

Automatic reorganization (AUTO_REORG) = OFF

DB2set Parameters

DB2_RESOURCE_POLICY=c:\tpc-c.ibm\cfg\aff.cfg
 DB2_SELUDI_COMM_BUFFER=Y
 DB2_USE_ALTERNATE_PAGE_CLEANING=YES
 DB2_MAX_NON_TABLE_LOCKS=500
 DB2_LGPAGE_BP=ON
 DB2_TRUSTED_BINDIN=ON
 DB2_KEEPTABLELOCK=ON
 DB2_NO_FORK_CHECK=ON
 DB2_FMP_COMM_HEAPSZ=0
 DB2_APM_PERFORMANCE=ALL
 DB2_ENABLE_BUFPD=OFF
 DB2_PINNED_BP=ON
 DB2_SELECTIVITY=ON
 DB2ASSUMEUPDATE=ON
 DB2CHECKCLIENTINTERVAL=0
 DB2_HASH_JOIN=OFF
 DB2_COLLECT_TS_REC_INFO=false
 DB2COMM=tcip
 DB2CHKPTR=OFF

Aff.cfg

```
<RESOURCE_POLICY>
<DATABASE_RESOURCE_POLICY>
<DBNAME>TPCC</DBNAME>
<METHOD>CPUMASK</METHOD>
<RESOURCE_BINDING>
<RESOURCE>255</RESOURCE>
<DBMEM_PERCENTAGE>0.0</DBMEM_PERCENTAGE>
<SERVICE_NAME>4550</SERVICE_NAME>
<BUFFERPOOL_BINDING>
<NUM_CLEANERS>12</NUM_CLEANERS>
<BUFFERPOOL_ID>4</BUFFERPOOL_ID>
<BUFFERPOOL_ID>5</BUFFERPOOL_ID>
<BUFFERPOOL_ID>7</BUFFERPOOL_ID>
<BUFFERPOOL_ID>8</BUFFERPOOL_ID>
<BUFFERPOOL_ID>9</BUFFERPOOL_ID>
<BUFFERPOOL_ID>10</BUFFERPOOL_ID>
<BUFFERPOOL_ID>11</BUFFERPOOL_ID>
<BUFFERPOOL_ID>12</BUFFERPOOL_ID>
<BUFFERPOOL_ID>13</BUFFERPOOL_ID>
<BUFFERPOOL_ID>14</BUFFERPOOL_ID>
</BUFFERPOOL_BINDING>
</RESOURCE_BINDING>
</RESOURCE_BINDING>
<RESOURCE>65280</RESOURCE>
<DBMEM_PERCENTAGE>0.0</DBMEM_PERCENTAGE>
```

```
<SERVICE_NAME>5550</SERVICE_NAME>
<BUFFERPOOL_BINDING>
<NUM_CLEANERS>12</NUM_CLEANERS>
<BUFFERPOOL_ID>15</BUFFERPOOL_ID>
<BUFFERPOOL_ID>16</BUFFERPOOL_ID>
<BUFFERPOOL_ID>17</BUFFERPOOL_ID>
<BUFFERPOOL_ID>18</BUFFERPOOL_ID>
<BUFFERPOOL_ID>19</BUFFERPOOL_ID>
<BUFFERPOOL_ID>20</BUFFERPOOL_ID>
<BUFFERPOOL_ID>21</BUFFERPOOL_ID>
<BUFFERPOOL_ID>22</BUFFERPOOL_ID>
<BUFFERPOOL_ID>23</BUFFERPOOL_ID>
<BUFFERPOOL_ID>24</BUFFERPOOL_ID>
</BUFFERPOOL_BINDING>
</RESOURCE_BINDING>
</DATABASE_RESOURCE_POLICY>
</RESOURCE_POLICY>
```

Microsoft Windows Server 2003 Enterprise x64 Edition

Server Configuration Parameters

Server Configuration Parameters

Microsoft Windows Server 2003 x64 Enterprise Edition Configuration

The following services were set as manual on the server:

- Automatic Updates
- Computer Browser
- Cryptographic Services
- DHCP Client
- Distributed Link Tracking Client
- DNS Client
- Help and Support
- IPSEC service
- Print Spooler
- Remote Registry
- System Event Notification
- Task Scheduler
- Windows Time
- Wireless Configuration

System Information Report

System Information report written at: 05/20/05 10:49:15

System Name: DB2SERV2

[System Summary]

Item	Value
OS Name	Microsoft(R) Windows(R) Server 2003 Enterprise x64 Edition
Version	5.2.3790 Service Pack 1 Build 3790
Other OS Description	Not Available
OS Manufacturer	Microsoft Corporation
Activation Status	Activation Pending (12 days remaining)
System Name	DB2SERV2
System Manufacturer	IBM

System Model eserver xSeries 460-[88724RZ]-
 System Type x64-based PC
 Processor EM64T Family 15 Model 4 Stepping 1 GenuineIntel ~3336 Mhz
 Processor EM64T Family 15 Model 4 Stepping 1 GenuineIntel ~3336 Mhz
 Processor EM64T Family 15 Model 4 Stepping 1 GenuineIntel ~3336 Mhz
 Processor EM64T Family 15 Model 4 Stepping 1 GenuineIntel ~3336 Mhz
 Processor EM64T Family 15 Model 4 Stepping 1 GenuineIntel ~3336 Mhz
 Processor EM64T Family 15 Model 4 Stepping 1 GenuineIntel ~3336 Mhz
 Processor EM64T Family 15 Model 4 Stepping 1 GenuineIntel ~3336 Mhz
 Processor EM64T Family 15 Model 4 Stepping 1 GenuineIntel ~3336 Mhz
 Processor EM64T Family 15 Model 4 Stepping 1 GenuineIntel ~3336 Mhz
 Processor EM64T Family 15 Model 4 Stepping 1 GenuineIntel ~3336 Mhz
 Processor EM64T Family 15 Model 4 Stepping 1 GenuineIntel ~3336 Mhz
 Processor EM64T Family 15 Model 4 Stepping 1 GenuineIntel ~3336 Mhz
 Processor EM64T Family 15 Model 4 Stepping 1 GenuineIntel ~3336 Mhz
 Processor EM64T Family 15 Model 4 Stepping 1 GenuineIntel ~3336 Mhz
 Processor EM64T Family 15 Model 4 Stepping 1 GenuineIntel ~3336 Mhz
 BIOS Version/Date IBM-[ZUE128AUS-1.03]-, 4/25/2005
 SMBIOS Version 2.3
 Windows Directory C:\WINDOWS
 System Directory C:\WINDOWS\system32
 Boot Device \Device\HarddiskVolume319
 Locale United States
 Hardware Abstraction Layer Version = "5.2.3790.1830 (srv03_sp1_rtm.050324-1447)"
 User NameDB2SERV2\Administrator
 Time ZoneEastern Daylight Time
 Total Physical Memory 129,022.88 MB
 Available Physical Memory 122.81 GB
 Total Virtual Memory 251.15 GB
 Available Virtual Memory 250.75 GB
 Page File Space 128.00 GB
 Page File C:\pagefile.sys
 [Hardware Resources]

[Conflicts/Sharing]

Resource Device
 Memory Address 0xF8900000-0xFFFFFFFF PCI bus
 Memory Address 0xF8900000-0xFFFFFFFF Broadcom NetXtreme Gigabit Ethernet #2
 I/O Port 0x00000000-0x0000E07F PCI bus
 I/O Port 0x00000000-0x0000E07F Direct memory access controller
 Memory Address 0xF0000000-0xF80FFFFFFF PCI bus
 Memory Address 0xF0000000-0xF80FFFFFFF Radeon 7000 / RADEON VE Family (Microsoft Corporation)
 I/O Port 0x00004400-0x0000FE7F PCI bus
 I/O Port 0x00004400-0x0000FE7F QLogic Fibre Channel Adapter
 Memory Address 0xF9500000-0xFFFFFFFF PCI bus
 Memory Address 0xF9500000-0xFFFFFFFF Broadcom NetXtreme Gigabit Ethernet #3
 I/O Port 0x00002600-0x0000FE7F PCI bus
 I/O Port 0x00002600-0x0000FE7F QLogic Fibre Channel Adapter
 IRQ 20 NEC PCI to USB Open Host Controller
 IRQ 20 NEC PCI to USB Open Host Controller
 IRQ 20 Standard Enhanced PCI to USB Host Controller
 Memory Address 0xF8800000-0xFFFFFFFF PCI bus
 Memory Address 0xF8800000-0xFFFFFFFF Radeon 7000 / RADEON VE Family (Microsoft Corporation)
 I/O Port 0x00002200-0x0000FE7F PCI bus
 I/O Port 0x00002200-0x0000FE7F QLogic Fibre Channel Adapter
 I/O Port 0x00002000-0x0000FE7F PCI bus
 I/O Port 0x00002000-0x0000FE7F QLogic Fibre Channel Adapter
 I/O Port 0x00004600-0x0000FE7F PCI bus
 I/O Port 0x00004600-0x0000FE7F QLogic Fibre Channel Adapter
 Memory Address 0xF8E00000-0xFFFFFFFF PCI bus
 Memory Address 0xF8E00000-0xFFFFFFFF PCI-to-PCI bridge
 I/O Port 0x00004200-0x0000FE7F PCI bus
 I/O Port 0x00004200-0x0000FE7F QLogic Fibre Channel Adapter
 I/O Port 0x00002400-0x0000FE7F PCI bus
 I/O Port 0x00002400-0x0000FE7F QLogic Fibre Channel Adapter

I/O Port 0x00004800-0x0000FE7F	PCI bus	0x000003F8-0x000003FF	Communications Port (COM1)	OK
I/O Port 0x00004800-0x0000FE7F	QLogic Fibre Channel Adapter	0x000002F8-0x000002FF	Communications Port (COM2)	OK
IRQ 90	NEC PCI to USB Open Host Controller	0x00000020-0x00000021	Advanced programmable interrupt controller	OK
IRQ 90	NEC PCI to USB Open Host Controller	OK		
IRQ 90	Standard Enhanced PCI to USB Host Controller	0x000000A0-0x000000A1	Advanced programmable interrupt controller	OK
Memory Address 0xA0000-0xBFFFF	PCI bus	OK		
Memory Address 0xA0000-0xBFFFF	Radeon 7000 / RADEON VE Family (Microsoft Corporation)	0x00000080-0x0000008F	Direct memory access controller	OK
Memory Address 0xF9A00000-0xFFFFFFFF	PCI bus	0x000000C0-0x000000DF	Direct memory access controller	OK
Memory Address 0xF9A00000-0xFFFFFFFF	PCI standard	0x00000040-0x00000043	System timer	OK
Memory Address 0xF9A00000-0xFFFFFFFF	PCI-to-PCI bridge	0x00000070-0x00000073	System CMOS/real time clock	OK
Memory Address 0xF9400000-0xFFFFFFFF	PCI bus	0x00000061-0x00000061	System speaker	OK
Memory Address 0xF9400000-0xFFFFFFFF	NEC PCI to USB Open Host Controller	0x000000F0-0x000000FF	Numeric data processor	OK
[DMA]		0x0000002E-0x0000002F	Motherboard resources	OK
Resource Device Status		0x0000004E-0x0000004F	Motherboard resources	OK
Channel 4 Direct memory access controller	OK	0x00000052-0x00000053	Motherboard resources	OK
[Forced Hardware]		0x00000092-0x00000092	Motherboard resources	OK
Device PNP Device ID		0x00000094-0x0000009F	Motherboard resources	OK
[I/O]		0x000000A8-0x000000A9	Motherboard resources	OK
Resource Device Status		0x000000400-0x00000047F	Motherboard resources	OK
0x00000000-0x0000E07F	PCI bus OK	0x000000480-0x0000004FF	Motherboard resources	OK
0x00000000-0x0000E07F	Direct memory access controller OK	0x000000500-0x00000055F	Motherboard resources	OK
0x00001800-0x000018FF	Radeon 7000 / RADEON VE Family (Microsoft Corporation) OK	0x00000600-0x00000600	Motherboard resources	OK
0x000003B0-0x000003BB	Radeon 7000 / RADEON VE Family (Microsoft Corporation) OK	0x00000800-0x00000803	Motherboard resources	OK
0x000003C0-0x000003DF	Radeon 7000 / RADEON VE Family (Microsoft Corporation) OK	0x00000C00-0x00000CDF	Motherboard resources	OK
0x00000700-0x0000070F	Standard Dual Channel PCI IDE Controller OK	0x00000F50-0x00000F5F	Motherboard resources	OK
0x000001F0-0x000001F7	Primary IDE Channel OK	0x00002000-0x0000FE7F	PCI bus OK	
0x000003F6-0x000003F6	Primary IDE Channel OK	0x00002000-0x0000FE7F	QLogic Fibre Channel Adapter	OK
0x00000170-0x00000177	Secondary IDE Channel OK	0x00002200-0x0000FE7F	PCI bus OK	
0x00000376-0x00000376	Secondary IDE Channel OK	0x00002200-0x0000FE7F	QLogic Fibre Channel Adapter	OK
0x00000060-0x00000060	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard OK	0x00002400-0x0000FE7F	PCI bus OK	
0x00000064-0x00000064	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard OK	0x00002400-0x0000FE7F	QLogic Fibre Channel Adapter	OK
		0x00002600-0x0000FE7F	PCI bus OK	
		0x00002600-0x0000FE7F	QLogic Fibre Channel Adapter	OK
		0x00003000-0x0000F07F	PCI bus OK	
		0x00004000-0x0000FE7F	PCI bus OK	

0x00004200-0x0000FE7F	PCI bus	OK
0x00004200-0x0000FE7F	QLogic Fibre Channel Adapter	OK
0x00004400-0x0000FE7F	PCI bus	OK
0x00004400-0x0000FE7F	QLogic Fibre Channel Adapter	OK
0x00004600-0x0000FE7F	PCI bus	OK
0x00004600-0x0000FE7F	QLogic Fibre Channel Adapter	OK
0x00004800-0x0000FE7F	PCI bus	OK
0x00004800-0x0000FE7F	QLogic Fibre Channel Adapter	OK
0x00005000-0x0000F07F	PCI bus	OK

[IRQs]

Resource	Device	Status
IRQ 9	Microsoft ACPI-Compliant System	OK
IRQ 16	Radeon 7000 / RADEON VE Family (Microsoft Corporation)	OK
IRQ 20	NEC PCI to USB Open Host Controller	OK
IRQ 20	NEC PCI to USB Open Host Controller	OK
IRQ 20	Standard Enhanced PCI to USB Host Controller	OK
IRQ 14	Primary IDE Channel	OK
IRQ 1	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard	OK
IRQ 12	PS/2 Compatible Mouse	OK
IRQ 4	Communications Port (COM1)	OK
IRQ 3	Communications Port (COM2)	OK
IRQ 0	System timer	OK
IRQ 8	System CMOS/real time clock	OK
IRQ 13	Numeric data processor	OK
IRQ 24	Broadcom NetXtreme Gigabit Ethernet #2	OK
IRQ 28	Broadcom NetXtreme Gigabit Ethernet	OK
IRQ 18	QLogic Fibre Channel Adapter	OK
IRQ 19	QLogic Fibre Channel Adapter	OK
IRQ 51	QLogic Fibre Channel Adapter	OK
IRQ 52	QLogic Fibre Channel Adapter	OK
IRQ 54	IBM ServeRAID 6M Controller	OK
IRQ 90	NEC PCI to USB Open Host Controller	OK
IRQ 90	NEC PCI to USB Open Host Controller	OK

IRQ 90	Standard Enhanced PCI to USB Host Controller	OK
IRQ 94	Broadcom NetXtreme Gigabit Ethernet #3	OK
IRQ 98	Broadcom NetXtreme Gigabit Ethernet #4	OK
IRQ 88	QLogic Fibre Channel Adapter	OK
IRQ 89	QLogic Fibre Channel Adapter	OK
IRQ 121	QLogic Fibre Channel Adapter	OK
IRQ 122	QLogic Fibre Channel Adapter	OK
IRQ 124	IBM ServeRAID 6M Controller	OK

[Memory]

Resource	Device	Status
0xA0000-0xBFFFF	PCI bus	OK
0xA0000-0xBFFFF	Radeon 7000 / RADEON VE Family (Microsoft Corporation)	OK
0xF0000000-0xF80FFFFFF	PCI bus	OK
0xF0000000-0xF80FFFFFF	Radeon 7000 / RADEON VE Family (Microsoft Corporation)	OK
0xF8800000-0xFFFFFFFF	PCI bus	OK
0xF8800000-0xFFFFFFFF	Radeon 7000 / RADEON VE Family (Microsoft Corporation)	OK
0xF8810000-0xF8810FFF	NEC PCI to USB Open Host Controller	OK
0xF8811000-0xF8811FFF	NEC PCI to USB Open Host Controller	OK
0xF8812000-0xF88120FF	Standard Enhanced PCI to USB Host Controller	OK
0x0400-0x04FF	System board	OK
0x100000-0x7FFFFFFF	Memory Module	OK
0xF8900000-0xFFFFFFFF	PCI bus	OK
0xF8900000-0xFFFFFFFF	Broadcom NetXtreme Gigabit Ethernet #2	OK
0xF8910000-0xF891FFFF	Broadcom NetXtreme Gigabit Ethernet	OK
0xF8A00000-0xFFFFFFFF	PCI bus	OK
0xF8A20000-0xF8A20FFF	QLogic Fibre Channel Adapter	OK
0xF8B00000-0xFFFFFFFF	PCI bus	OK
0xF8B20000-0xF8B20FFF	QLogic Fibre Channel Adapter	OK
0xF8C00000-0xFFFFFFFF	PCI bus	OK
0xF8C20000-0xF8C20FFF	QLogic Fibre Channel Adapter	OK

0xF8D00000-0xFFFFFFFF PCI bus OK

0xF8D20000-0xF8D20FFF QLogic Fibre Channel Adapter OK

0xF8000000-0xFFCFFFFFF PCI bus OK

0xF8400000-0xFFCFFFFFF PCI bus OK

0xF8E00000-0xFFFFFFFF PCI bus OK

0xF8E00000-0xFFFFFFFF PCI standard PCI-to-PCI bridge OK

0xF8E08000-0xF8E08FFF IBM ServeRAID 6M Controller OK

0xF9400000-0xFFFFFFFF PCI bus OK

0xF9400000-0xFFFFFFFF NEC PCI to USB Open Host Controller OK

0xF9401000-0xF9401FFF NEC PCI to USB Open Host Controller OK

0xF9402000-0xF9402FFF Standard Enhanced PCI to USB Host Controller OK

0xF9500000-0xFFFFFFFF PCI bus OK

0xF9500000-0xFFFFFFFF Broadcom NetXtreme Gigabit Ethernet #3 OK

0xF9510000-0xF951FFFF Broadcom NetXtreme Gigabit Ethernet #4 OK

0xF9600000-0xFFFFFFFF PCI bus OK

0xF9620000-0xF9620FFF QLogic Fibre Channel Adapter OK

0xF9700000-0xFFFFFFFF PCI bus OK

0xF9720000-0xF9720FFF QLogic Fibre Channel Adapter OK

0xF9800000-0xFFFFFFFF PCI bus OK

0xF9820000-0xF9820FFF QLogic Fibre Channel Adapter OK

0xF9900000-0xFFFFFFFF PCI bus OK

0xF9920000-0xF9920FFF QLogic Fibre Channel Adapter OK

0xF9000000-0xFFCFFFFFF PCI bus OK

0xF9C00000-0xFFCFFFFFF PCI bus OK

0xF9A00000-0xFFFFFFFF PCI bus OK

0xF9A00000-0xFFFFFFFF PCI standard PCI-to-PCI bridge OK

0xF9A08000-0xF9A08FFF IBM ServeRAID 6M Controller OK

[Components]

[Multimedia]

[Audio Codecs]

CODEC Version	Manufacturer Size	Description Creation Date	Status	File
---------------	-------------------	---------------------------	--------	------

c:\windows\system32\msg711.acm Microsoft Corporation
 OK C:\WINDOWS\system32\MSG711.ACM 5.2.3790.1830
 (srv03_sp1_rtm.050324-1447) 13.50 KB (13,824 bytes) 3/25/2005
 7:00 AM

c:\windows\system32\tssoft32.acm DSP GROUP, INC.
 OK C:\WINDOWS\system32\TSSOFT32.ACM 1.01 13.50 KB
 (13,824 bytes) 3/25/2005 7:00 AM

c:\windows\system32\msgsm32.acm Microsoft Corporation
 OK C:\WINDOWS\system32\MSGSM32.ACM 5.2.3790.1830
 (srv03_sp1_rtm.050324-1447) 34.50 KB (35,328 bytes) 3/25/2005
 7:00 AM

c:\windows\system32\imaadp32.acm Microsoft Corporation
 OK C:\WINDOWS\system32\IMAADP32.ACM 5.2.3790.1830
 (srv03_sp1_rtm.050324-1447) 24.00 KB (24,576 bytes) 3/25/2005
 7:00 AM

c:\windows\system32\msadp32.acm Microsoft Corporation
 OK C:\WINDOWS\system32\MSADP32.ACM 5.2.3790.1830
 (srv03_sp1_rtm.050324-1447) 23.50 KB (24,064 bytes) 3/25/2005
 7:00 AM

[Video Codecs]

CODEC Version	Manufacturer Size	Description Creation Date	Status	File
---------------	-------------------	---------------------------	--------	------

c:\windows\system32\msrle32.dll Microsoft Corporation
 OK C:\WINDOWS\system32\MSRLE32.DLL 5.2.3790.1830
 (srv03_sp1_rtm.050324-1447) 15.50 KB (15,872 bytes) 3/25/2005
 7:00 AM

c:\windows\system32\iyuv_32.dll Microsoft Corporation
 OK C:\WINDOWS\system32\IYUV_32.DLL 5.2.3790.1830
 (srv03_sp1_rtm.050324-1447) 52.50 KB (53,760 bytes) 3/24/2005
 12:19 PM

c:\windows\system32\msvidc32.dll Microsoft Corporation
 OK C:\WINDOWS\system32\MSVIDC32.DLL 5.2.3790.1830
 (srv03_sp1_rtm.050324-1447) 43.00 KB (44,032 bytes) 3/25/2005
 7:00 AM

c:\windows\system32\msyuv.dll Microsoft Corporation
 OK C:\WINDOWS\system32\MSYUV.DLL 5.2.3790.1830
 (srv03_sp1_rtm.050324-1447) 21.00 KB (21,504 bytes) 3/24/2005
 12:21 PM

c:\windows\system32\tsbyuv.dll Microsoft Corporation
 OK C:\WINDOWS\system32\TSBYUV.DLL 5.2.3790.1830
 (srv03_sp1_rtm.050324-1447) 12.50 KB (12,800 bytes) 3/24/2005
 12:34 PM

[CD-ROM]

Item	Value
Drive	D:
Description	CD-ROM Drive
Media Loaded	No
Media Type	CD-ROM
Name	HL-DT-ST DVD-ROM GDR8082N

Manufacturer (Standard CD-ROM drives)
 Status OK
 Transfer Rate Not Available
 SCSI Target ID 0
 PNP Device ID
 IDE\CDROMHL-DT-ST_DVD-ROM_GDR8082N_____0L03____
 _\5&A8D2D22&0&0.0.0

[Sound Device]

Item Value

[Display]

Item Value

Name Radeon 7000 / RADEON VE Family (Microsoft Corporation)

PNP Device ID
 PCI\VEN_1002&DEV_5159&SUBSYS_02C81014&REV_00\3&267A616A&
 0&08

Adapter Type ATI display adapter (0x5159), ATI Technologies Inc.
 compatible

Adapter Description Radeon 7000 / RADEON VE Family (Microsoft
 Corporation)

Adapter RAM 16.00 MB (16,777,216 bytes)

Installed Drivers ati2dvag.dll

Driver Version 6.14.10.6509

INF File atiixpag.inf (ati2mtag_RV100 section)

Color Planes 1

Color Table Entries 4294967296

Resolution 1024 x 768 x 60 hertz

Bits/Pixel 32

Memory Address 0xF0000000-0xF80FFFFF

I/O Port 0x00001800-0x000018FF

Memory Address 0xF8800000-0xFFFFFFFF

IRQ Channel IRQ 16

I/O Port 0x000003B0-0x000003BB

I/O Port 0x000003C0-0x000003DF

Memory Address 0xA0000-0xBFFFF

[Infrared]

Item Value

[Input]

[Keyboard]

Item Value

Description USB Human Interface Device

Name Enhanced (101- or 102-key)

Layout 00000409

PNP Device ID
 USB\VID_04B3&PID_4001&MI_00\6&18E8217A&0&0000

Number of Function Keys 12

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&13245C1&0

Number of Function Keys 12

I/O Port 0x00000060-0x00000060

I/O Port 0x00000064-0x00000064

IRQ Channel IRQ 1

Description USB Human Interface Device

Name Enhanced (101- or 102-key)

Layout 00000409

PNP Device ID
 USB\VID_04B3&PID_4001&MI_00\6&18E8217A&0&0000

Number of Function Keys 12

[Pointing Device]

Item Value

Hardware Type USB Human Interface Device

Number of Buttons 3

Status OK

PNP Device ID
 USB\VID_04B3&PID_4001&MI_01\6&18E8217A&0&0001

Power Management Supported No

Double Click Threshold 6

Handedness Right Handed Operation

Hardware Type PS/2 Compatible Mouse

Number of Buttons 3

Status OK

PNP Device ID ACPI\PNP0F13\4&13245C1&0
 Power Management Supported No
 Double Click Threshold 6
 Handedness Right Handed Operation
 IRQ Channel IRQ 12
 Hardware Type USB Human Interface Device
 Number of Buttons 3
 Status OK

PNP Device ID
 USB\VID_04B3&PID_4001&MI_01\6&18E8217A&0&0001

Power Management Supported No
 Double Click Threshold 6
 Handedness Right Handed Operation

[Modem]

Item Value

[Network]

[Adapter]

Item Value

Name [00000001] Broadcom NetXtreme Gigabit Ethernet

Adapter Type Ethernet 802.3
 Product Type Broadcom NetXtreme Gigabit Ethernet
 Installed Yes

PNP Device ID
 PCI\VEN_14E4&DEV_1648&SUBSYS_02E71014&REV_10\3&13C0B0C5&0&09

Last Reset 5/20/2005 9:04 AM

Index 1

Service Name b57nd

IP Address 192.168.20.234

IP Subnet 255.255.255.0

Default IP Gateway Not Available

DHCP Enabled No

DHCP Server Not Available

DHCP Lease Expires Not Available

DHCP Lease Obtained Not Available

MAC Address 00:0D:60:98:0B:E5

Memory Address 0xF8910000-0xF891FFFF

IRQ Channel IRQ 28

Name [00000002] Broadcom NetXtreme Gigabit Ethernet

Adapter Type Ethernet 802.3

Product Type Broadcom NetXtreme Gigabit Ethernet

Installed Yes

PNP Device ID
 PCI\VEN_14E4&DEV_1648&SUBSYS_02E71014&REV_10\3&13C0B0C5&0&08

Last Reset 5/20/2005 9:04 AM

Index 2

Service Name b57nd

IP Address 10.0.0.249

IP Subnet 255.0.0.0

Default IP Gateway Not Available

DHCP Enabled No

DHCP Server Not Available

DHCP Lease Expires Not Available

DHCP Lease Obtained Not Available

MAC Address 00:0D:60:98:0B:E4

Memory Address 0xF8900000-0xFFFFFFFF

IRQ Channel IRQ 24

Name [00000003] RAS Async Adapter

Adapter Type Not Available

Product Type RAS Async Adapter

Installed Yes

PNP Device ID Not Available

Last Reset 5/20/2005 9:04 AM

Index 3

Service Name AsyncMac

IP Address Not Available

IP Subnet Not Available

Default IP Gateway Not Available

DHCP Enabled No

DHCP Server Not Available
 DHCP Lease Expires Not Available
 DHCP Lease Obtained Not Available
 MAC Address Not Available
 Name [00000004] WAN Miniport (L2TP)
 Adapter Type Not Available
 Product Type WAN Miniport (L2TP)
 Installed Yes
 PNP Device ID ROOT\MS_L2TPMINIPOINT\0000
 Last Reset 5/20/2005 9:04 AM
 Index 4
 Service Name Rasl2tp
 IP Address Not Available
 IP Subnet Not Available
 Default IP Gateway Not Available
 DHCP Enabled No
 DHCP Server Not Available
 DHCP Lease Expires Not Available
 DHCP Lease Obtained Not Available
 MAC Address Not Available
 Name [00000005] WAN Miniport (PPTP)
 Adapter Type Wide Area Network (WAN)
 Product Type WAN Miniport (PPTP)
 Installed Yes
 PNP Device ID ROOT\MS_PPTPMINIPOINT\0000
 Last Reset 5/20/2005 9:04 AM
 Index 5
 Service Name PptpMiniport
 IP Address Not Available
 IP Subnet Not Available
 Default IP Gateway Not Available
 DHCP Enabled No
 DHCP Server Not Available
 DHCP Lease Expires Not Available

DHCP Lease Obtained Not Available
 MAC Address 50:50:54:50:30:30
 Name [00000006] WAN Miniport (PPPOE)
 Adapter Type Wide Area Network (WAN)
 Product Type WAN Miniport (PPPOE)
 Installed Yes
 PNP Device ID ROOT\MS_PPPOEMINIPOINT\0000
 Last Reset 5/20/2005 9:04 AM
 Index 6
 Service Name RasPppoe
 IP Address Not Available
 IP Subnet Not Available
 Default IP Gateway Not Available
 DHCP Enabled No
 DHCP Server Not Available
 DHCP Lease Expires Not Available
 DHCP Lease Obtained Not Available
 MAC Address 33:50:6F:45:30:30
 Name [00000007] Direct Parallel
 Adapter Type Not Available
 Product Type Direct Parallel
 Installed Yes
 PNP Device ID ROOT\MS_PTIMINIPOINT\0000
 Last Reset 5/20/2005 9:04 AM
 Index 7
 Service Name Raspti
 IP Address Not Available
 IP Subnet Not Available
 Default IP Gateway Not Available
 DHCP Enabled No
 DHCP Server Not Available
 DHCP Lease Expires Not Available
 DHCP Lease Obtained Not Available
 MAC Address Not Available

Name [00000008] WAN Miniport (IP)

Adapter Type Not Available

Product Type WAN Miniport (IP)

Installed Yes

PNP Device ID ROOT\MS_NDISWANIP\0000

Last Reset 5/20/2005 9:04 AM

Index 8

Service Name NdisWan

IP Address Not Available

IP Subnet Not Available

Default IP Gateway Not Available

DHCP Enabled No

DHCP Server Not Available

DHCP Lease Expires Not Available

DHCP Lease Obtained Not Available

MAC Address Not Available

Name [00000009] Broadcom NetXtreme Gigabit Ethernet

Adapter Type Ethernet 802.3

Product Type Broadcom NetXtreme Gigabit Ethernet

Installed Yes

PNP Device ID
PCI\VEN_14E4&DEV_1648&SUBSYS_02E71014&REV_10\3&1D521019&0&08

Last Reset 5/20/2005 9:04 AM

Index 9

Service Name b57nd

IP Address 192.168.122.200

IP Subnet 255.255.255.0

Default IP Gateway Not Available

DHCP Enabled No

DHCP Server Not Available

DHCP Lease Expires Not Available

DHCP Lease Obtained Not Available

MAC Address 00:0D:60:98:0C:52

Memory Address 0xF9500000-0xFFFFFFFF

IRQ Channel IRQ 94

Name [00000010] Broadcom NetXtreme Gigabit Ethernet

Adapter Type Ethernet 802.3

Product Type Broadcom NetXtreme Gigabit Ethernet

Installed Yes

PNP Device ID
PCI\VEN_14E4&DEV_1648&SUBSYS_02E71014&REV_10\3&1D521019&0&09

Last Reset 5/20/2005 9:04 AM

Index 10

Service Name b57nd

IP Address 10.0.0.250

IP Subnet 255.0.0.0

Default IP Gateway Not Available

DHCP Enabled No

DHCP Server Not Available

DHCP Lease Expires Not Available

DHCP Lease Obtained Not Available

MAC Address 00:0D:60:98:0C:53

Memory Address 0xF9510000-0xF951FFFF

IRQ Channel IRQ 98

[Protocol]

Item	Value
Name	MSAFD Tcpip [TCP/IP]
Connectionless Service	No
Guarantees Delivery	Yes
Guarantees Sequencing	Yes
Maximum Address Size	16 bytes
Maximum Message Size	0 bytes
Message Oriented	No
Minimum Address Size	16 bytes
Pseudo Stream Oriented	No
Supports Broadcasting	No
Supports Connect Data	No
Supports Disconnect Data	No

Supports Encryption	No
Supports Expedited Data	Yes
Supports Graceful Closing	Yes
Supports Guaranteed Bandwidth	No
Supports Multicasting	No
Name	MSAFD Tcpip [UDP/IP]
Connectionless Service	Yes
Guarantees Delivery	No
Guarantees Sequencing	No
Maximum Address Size	16 bytes
Maximum Message Size	63.93 KB (65,467 bytes)
Message Oriented	Yes
Minimum Address Size	16 bytes
Pseudo Stream Oriented	No
Supports Broadcasting	Yes
Supports Connect Data	No
Supports Disconnect Data	No
Supports Encryption	No
Supports Expedited Data	No
Supports Graceful Closing	No
Supports Guaranteed Bandwidth	No
Supports Multicasting	Yes
Name	RSVP UDP Service Provider
Connectionless Service	Yes
Guarantees Delivery	No
Guarantees Sequencing	No
Maximum Address Size	16 bytes
Maximum Message Size	63.93 KB (65,467 bytes)
Message Oriented	Yes
Minimum Address Size	16 bytes
Pseudo Stream Oriented	No
Supports Broadcasting	Yes
Supports Connect Data	No
Supports Disconnect Data	No

Supports Encryption	Yes
Supports Expedited Data	No
Supports Graceful Closing	No
Supports Guaranteed Bandwidth	No
Supports Multicasting	Yes
Name	RSVP TCP Service Provider
Connectionless Service	No
Guarantees Delivery	Yes
Guarantees Sequencing	Yes
Maximum Address Size	16 bytes
Maximum Message Size	0 bytes
Message Oriented	No
Minimum Address Size	16 bytes
Pseudo Stream Oriented	No
Supports Broadcasting	No
Supports Connect Data	No
Supports Disconnect Data	No
Supports Encryption	Yes
Supports Expedited Data	Yes
Supports Graceful Closing	Yes
Supports Guaranteed Bandwidth	No
Supports Multicasting	No
[WinSock]	
Item	Value
File	c:\windows\system32\wsock32.dll
Size	24.50 KB (25,088 bytes)
Version	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)
[Ports]	
[Serial]	
Item	Value
Name	Communications Port (COM1)
Status	OK
PNP Device ID	ACPI\PNP0501\1
Maximum Input Buffer Size	0

Maximum Output Buffer Size	No	XOnXOff InFlow Control	0
Settable Baud Rate	Yes	XOnXOff OutFlow Control	0
Settable Data Bits	Yes	I/O Port	0x000003F8-0x000003FF
Settable Flow Control	Yes	IRQ Channel	IRQ 4
Settable Parity	Yes	Name	Communications Port (COM2)
Settable Parity Check	Yes	Status	OK
Settable Stop Bits	Yes	PNP Device ID	ACPI\PNP0501\2
Settable RLSD	Yes	Maximum Input Buffer Size	0
Supports RLSD	Yes	Maximum Output Buffer Size	No
Supports 16 Bit Mode	No	Settable Baud Rate	Yes
Supports Special Characters	No	Settable Data Bits	Yes
Baud Rate	9600	Settable Flow Control	Yes
Bits/Byte	8	Settable Parity	Yes
Stop Bits	1	Settable Parity Check	Yes
Parity	None	Settable Stop Bits	Yes
Busy	No	Settable RLSD	Yes
Abort Read/Write on Error	No	Supports RLSD	Yes
Binary Mode Enabled	Yes	Supports 16 Bit Mode	No
Continue XMit on XOff	No	Supports Special Characters	No
CTS Outflow Control	No	Baud Rate	9600
Discard NULL Bytes	No	Bits/Byte	8
DSR Outflow Control	0	Stop Bits	1
DSR Sensitivity	0	Parity	None
DTR Flow Control Type	Enable	Busy	No
EOF Character	0	Abort Read/Write on Error	No
Error Replace Character	0	Binary Mode Enabled	Yes
Error Replacement Enabled	No	Continue XMit on XOff	No
Event Character	0	CTS Outflow Control	No
Parity Check Enabled	No	Discard NULL Bytes	No
RTS Flow Control Type	Enable	DSR Outflow Control	0
XOff Character	19	DSR Sensitivity	0
XOffXMit Threshold	512	DTR Flow Control Type	Enable
XOn Character	17	EOF Character	0
XOnXMit Threshold	2048	Error Replace Character	0

Error Replacement Enabled No
 Event Character 0
 Parity Check Enabled No
 RTS Flow Control Type Enable
 XOff Character 19
 XOffXMit Threshold 512
 XOn Character 17
 XOnXMit Threshold 2048
 XOnXOff InFlow Control 0
 XOnXOff OutFlow Control 0

I/O Port 0x000002F8-0x000002FF

IRQ Channel IRQ 3

[Parallel]

Item Value

[Storage]

[Drives]

Item Value

Drive C:

Description Local Fixed Disk

Compressed No

File System NTFS

Size 33.90 GB (36,398,149,632 bytes)

Free Space 22.00 GB (23,620,890,624 bytes)

Volume Name

Volume Serial Number 401F2D61

Drive D:

Description CD-ROM Disc

Drive E:

Description Local Fixed Disk

Compressed No

File System NTFS

Size 242.08 GB (259,927,040,000 bytes)

Free Space 116.01 GB (124,561,178,624 bytes)

Volume Name Virtual Space

Volume Serial Number 5CC71915

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 Z:

Description Network Connection

Provider Name \\fsserv\ddrive

[Disks]

Item Value

Description Disk drive

Manufacturer (Standard disk drives)

Model IBM 1742-900 SCSI Disk Device

Bytes/Sector 512

Media Loaded Yes

Media Type Fixed hard disk

Partitions 13

SCSI Bus 0

SCSI Logical Unit 0

SCSI Port 6

SCSI Target ID 0

Sectors/Track 63

Size 334.01 GB (358,638,658,560 bytes)
 Total Cylinders 43,602
 Total Sectors 700,466,130
 Total Tracks 11,118,510
 Tracks/Cylinder 255
 Partition Disk #28, Partition #0
 Partition Size 334.00 GB (358,630,433,280 bytes)
 Partition Starting Offset 8,225,280 bytes
 Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 13
 SCSI Bus 0
 SCSI Logical Unit 2
 SCSI Port 6
 SCSI Target ID 0
 Sectors/Track 63
 Size 334.01 GB (358,638,658,560 bytes)
 Total Cylinders 43,602
 Total Sectors 700,466,130
 Total Tracks 11,118,510
 Tracks/Cylinder 255
 Partition Disk #29, Partition #0
 Partition Size 334.00 GB (358,630,433,280 bytes)
 Partition Starting Offset 8,225,280 bytes
 Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk

Partitions 13
 SCSI Bus 0
 SCSI Logical Unit 4
 SCSI Port 6
 SCSI Target ID 0
 Sectors/Track 63
 Size 334.01 GB (358,638,658,560 bytes)
 Total Cylinders 43,602
 Total Sectors 700,466,130
 Total Tracks 11,118,510
 Tracks/Cylinder 255
 Partition Disk #30, Partition #0
 Partition Size 334.00 GB (358,630,433,280 bytes)
 Partition Starting Offset 8,225,280 bytes
 Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 13
 SCSI Bus 0
 SCSI Logical Unit 6
 SCSI Port 6
 SCSI Target ID 0
 Sectors/Track 63
 Size 334.01 GB (358,638,658,560 bytes)
 Total Cylinders 43,602
 Total Sectors 700,466,130
 Total Tracks 11,118,510
 Tracks/Cylinder 255
 Partition Disk #31, Partition #0
 Partition Size 334.00 GB (358,630,433,280 bytes)
 Partition Starting Offset 8,225,280 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 13
 SCSI Bus 0
 SCSI Logical Unit 8
 SCSI Port 6
 SCSI Target ID 0
 Sectors/Track 63
 Size 334.01 GB (358,638,658,560 bytes)
 Total Cylinders 43,602
 Total Sectors 700,466,130
 Total Tracks 11,118,510
 Tracks/Cylinder 255
 Partition Disk #32, Partition #0
 Partition Size 334.00 GB (358,630,433,280 bytes)
 Partition Starting Offset 8,225,280 bytes
 Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 13
 SCSI Bus 0
 SCSI Logical Unit 10
 SCSI Port 6
 SCSI Target ID 0
 Sectors/Track 63
 Size 334.01 GB (358,638,658,560 bytes)
 Total Cylinders 43,602

Total Sectors 700,466,130
 Total Tracks 11,118,510
 Tracks/Cylinder 255
 Partition Disk #33, Partition #0
 Partition Size 334.00 GB (358,630,433,280 bytes)
 Partition Starting Offset 8,225,280 bytes
 Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 1
 SCSI Bus 0
 SCSI Logical Unit 12
 SCSI Port 6
 SCSI Target ID 0
 Sectors/Track 63
 Size 300.61 GB (322,776,437,760 bytes)
 Total Cylinders 39,242
 Total Sectors 630,422,730
 Total Tracks 10,006,710
 Tracks/Cylinder 255
 Partition Disk #34, Partition #0
 Partition Size 300.60 GB (322,768,212,480 bytes)
 Partition Starting Offset 8,225,280 bytes
 Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 13
 SCSI Bus 0

SCSI Logical Unit 1
 SCSI Port 8
 SCSI Target ID 0
 Sectors/Track 63
 Size 334.01 GB (358,638,658,560 bytes)
 Total Cylinders 43,602
 Total Sectors 700,466,130
 Total Tracks 11,118,510
 Tracks/Cylinder 255
 Partition Disk #42, Partition #0
 Partition Size 334.00 GB (358,630,433,280 bytes)
 Partition Starting Offset 8,225,280 bytes
 Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 13
 SCSI Bus 0
 SCSI Logical Unit 3
 SCSI Port 8
 SCSI Target ID 0
 Sectors/Track 63
 Size 334.01 GB (358,638,658,560 bytes)
 Total Cylinders 43,602
 Total Sectors 700,466,130
 Total Tracks 11,118,510
 Tracks/Cylinder 255
 Partition Disk #43, Partition #0
 Partition Size 334.00 GB (358,630,433,280 bytes)
 Partition Starting Offset 8,225,280 bytes
 Description Disk drive
 Manufacturer (Standard disk drives)

Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 13
 SCSI Bus 0
 SCSI Logical Unit 5
 SCSI Port 8
 SCSI Target ID 0
 Sectors/Track 63
 Size 334.01 GB (358,638,658,560 bytes)
 Total Cylinders 43,602
 Total Sectors 700,466,130
 Total Tracks 11,118,510
 Tracks/Cylinder 255
 Partition Disk #44, Partition #0
 Partition Size 334.00 GB (358,630,433,280 bytes)
 Partition Starting Offset 8,225,280 bytes
 Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 13
 SCSI Bus 0
 SCSI Logical Unit 7
 SCSI Port 8
 SCSI Target ID 0
 Sectors/Track 63
 Size 334.01 GB (358,638,658,560 bytes)
 Total Cylinders 43,602
 Total Sectors 700,466,130
 Total Tracks 11,118,510

Tracks/Cylinder 255
 Partition Disk #45, Partition #0
 Partition Size 334.00 GB (358,630,433,280 bytes)
 Partition Starting Offset 8,225,280 bytes
 Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 13
 SCSI Bus 0
 SCSI Logical Unit 9
 SCSI Port 8
 SCSI Target ID 0
 Sectors/Track 63
 Size 334.01 GB (358,638,658,560 bytes)
 Total Cylinders 43,602
 Total Sectors 700,466,130
 Total Tracks 11,118,510
 Tracks/Cylinder 255
 Partition Disk #46, Partition #0
 Partition Size 334.00 GB (358,630,433,280 bytes)
 Partition Starting Offset 8,225,280 bytes
 Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 13
 SCSI Bus 0
 SCSI Logical Unit 11
 SCSI Port 8

SCSI Target ID 0
 Sectors/Track 63
 Size 334.01 GB (358,638,658,560 bytes)
 Total Cylinders 43,602
 Total Sectors 700,466,130
 Total Tracks 11,118,510
 Tracks/Cylinder 255
 Partition Disk #47, Partition #0
 Partition Size 334.00 GB (358,630,433,280 bytes)
 Partition Starting Offset 8,225,280 bytes
 Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 1
 SCSI Bus 0
 SCSI Logical Unit 13
 SCSI Port 8
 SCSI Target ID 0
 Sectors/Track 63
 Size 300.61 GB (322,776,437,760 bytes)
 Total Cylinders 39,242
 Total Sectors 630,422,730
 Total Tracks 10,006,710
 Tracks/Cylinder 255
 Partition Disk #48, Partition #0
 Partition Size 300.60 GB (322,768,212,480 bytes)
 Partition Starting Offset 8,225,280 bytes
 Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512

Media Loaded Yes

Media Type Fixed hard disk

Partitions 13

SCSI Bus 0

SCSI Logical Unit 1

SCSI Port 4

SCSI Target ID 0

Sectors/Track 63

Size 334.01 GB (358,638,658,560 bytes)

Total Cylinders 43,602

Total Sectors 700,466,130

Total Tracks 11,118,510

Tracks/Cylinder 255

Partition Disk #14, Partition #0

Partition Size 334.00 GB (358,630,433,280 bytes)

Partition Starting Offset 8,225,280 bytes

Description Disk drive

Manufacturer (Standard disk drives)

Model IBM 1742-900 SCSI Disk Device

Bytes/Sector 512

Media Loaded Yes

Media Type Fixed hard disk

Partitions 13

SCSI Bus 0

SCSI Logical Unit 3

SCSI Port 4

SCSI Target ID 0

Sectors/Track 63

Size 334.01 GB (358,638,658,560 bytes)

Total Cylinders 43,602

Total Sectors 700,466,130

Total Tracks 11,118,510

Tracks/Cylinder 255

Partition Disk #15, Partition #0

Partition Size 334.00 GB (358,630,433,280 bytes)

Partition Starting Offset 8,225,280 bytes

Description Disk drive

Manufacturer (Standard disk drives)

Model IBM 1742-900 SCSI Disk Device

Bytes/Sector 512

Media Loaded Yes

Media Type Fixed hard disk

Partitions 13

SCSI Bus 0

SCSI Logical Unit 5

SCSI Port 4

SCSI Target ID 0

Sectors/Track 63

Size 334.01 GB (358,638,658,560 bytes)

Total Cylinders 43,602

Total Sectors 700,466,130

Total Tracks 11,118,510

Tracks/Cylinder 255

Partition Disk #16, Partition #0

Partition Size 334.00 GB (358,630,433,280 bytes)

Partition Starting Offset 8,225,280 bytes

Description Disk drive

Manufacturer (Standard disk drives)

Model IBM 1742-900 SCSI Disk Device

Bytes/Sector 512

Media Loaded Yes

Media Type Fixed hard disk

Partitions 13

SCSI Bus 0

SCSI Logical Unit 7

SCSI Port 4

SCSI Target ID 0

Sectors/Track 63

Size 334.01 GB (358,638,658,560 bytes)

Total Cylinders 43,602

Total Sectors 700,466,130

Total Tracks 11,118,510

Tracks/Cylinder 255

Partition Disk #17, Partition #0

Partition Size 334.00 GB (358,630,433,280 bytes)

Partition Starting Offset 8,225,280 bytes

Description Disk drive

Manufacturer (Standard disk drives)

Model IBM 1742-900 SCSI Disk Device

Bytes/Sector 512

Media Loaded Yes

Media Type Fixed hard disk

Partitions 13

SCSI Bus 0

SCSI Logical Unit 9

SCSI Port 4

SCSI Target ID 0

Sectors/Track 63

Size 334.01 GB (358,638,658,560 bytes)

Total Cylinders 43,602

Total Sectors 700,466,130

Total Tracks 11,118,510

Tracks/Cylinder 255

Partition Disk #18, Partition #0

Partition Size 334.00 GB (358,630,433,280 bytes)

Partition Starting Offset 8,225,280 bytes

Description Disk drive

Manufacturer (Standard disk drives)

Model IBM 1742-900 SCSI Disk Device

Bytes/Sector 512

Media Loaded Yes

Media Type Fixed hard disk

Partitions 13

SCSI Bus 0

SCSI Logical Unit 11

SCSI Port 4

SCSI Target ID 0

Sectors/Track 63

Size 334.01 GB (358,638,658,560 bytes)

Total Cylinders 43,602

Total Sectors 700,466,130

Total Tracks 11,118,510

Tracks/Cylinder 255

Partition Disk #19, Partition #0

Partition Size 334.00 GB (358,630,433,280 bytes)

Partition Starting Offset 8,225,280 bytes

Description Disk drive

Manufacturer (Standard disk drives)

Model IBM 1742-900 SCSI Disk Device

Bytes/Sector 512

Media Loaded Yes

Media Type Fixed hard disk

Partitions 1

SCSI Bus 0

SCSI Logical Unit 13

SCSI Port 4

SCSI Target ID 0

Sectors/Track 63

Size 300.61 GB (322,776,437,760 bytes)

Total Cylinders 39,242

Total Sectors 630,422,730

Total Tracks 10,006,710

Tracks/Cylinder 255

Partition Disk #20, Partition #0

Partition Size 300.60 GB (322,768,212,480 bytes)

Partition Starting Offset 8,225,280 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 13
 SCSI Bus 0
 SCSI Logical Unit 0
 SCSI Port 9
 SCSI Target ID 0
 Sectors/Track 63
 Size 334.01 GB (358,638,658,560 bytes)
 Total Cylinders 43,602
 Total Sectors 700,466,130
 Total Tracks 11,118,510
 Tracks/Cylinder 255
 Partition Disk #49, Partition #0
 Partition Size 334.00 GB (358,630,433,280 bytes)
 Partition Starting Offset 8,225,280 bytes
 Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 13
 SCSI Bus 0
 SCSI Logical Unit 2
 SCSI Port 9
 SCSI Target ID 0
 Sectors/Track 63
 Size 334.01 GB (358,638,658,560 bytes)
 Total Cylinders 43,602

Total Sectors 700,466,130
 Total Tracks 11,118,510
 Tracks/Cylinder 255
 Partition Disk #50, Partition #0
 Partition Size 334.00 GB (358,630,433,280 bytes)
 Partition Starting Offset 8,225,280 bytes
 Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 13
 SCSI Bus 0
 SCSI Logical Unit 4
 SCSI Port 9
 SCSI Target ID 0
 Sectors/Track 63
 Size 334.01 GB (358,638,658,560 bytes)
 Total Cylinders 43,602
 Total Sectors 700,466,130
 Total Tracks 11,118,510
 Tracks/Cylinder 255
 Partition Disk #51, Partition #0
 Partition Size 334.00 GB (358,630,433,280 bytes)
 Partition Starting Offset 8,225,280 bytes
 Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 13
 SCSI Bus 0

SCSI Logical Unit 6
 SCSI Port 9
 SCSI Target ID 0
 Sectors/Track 63
 Size 334.01 GB (358,638,658,560 bytes)
 Total Cylinders 43,602
 Total Sectors 700,466,130
 Total Tracks 11,118,510
 Tracks/Cylinder 255
 Partition Disk #52, Partition #0
 Partition Size 334.00 GB (358,630,433,280 bytes)
 Partition Starting Offset 8,225,280 bytes
 Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 13
 SCSI Bus 0
 SCSI Logical Unit 8
 SCSI Port 9
 SCSI Target ID 0
 Sectors/Track 63
 Size 334.01 GB (358,638,658,560 bytes)
 Total Cylinders 43,602
 Total Sectors 700,466,130
 Total Tracks 11,118,510
 Tracks/Cylinder 255
 Partition Disk #53, Partition #0
 Partition Size 334.00 GB (358,630,433,280 bytes)
 Partition Starting Offset 8,225,280 bytes
 Description Disk drive
 Manufacturer (Standard disk drives)

Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 13
 SCSI Bus 0
 SCSI Logical Unit 10
 SCSI Port 9
 SCSI Target ID 0
 Sectors/Track 63
 Size 334.01 GB (358,638,658,560 bytes)
 Total Cylinders 43,602
 Total Sectors 700,466,130
 Total Tracks 11,118,510
 Tracks/Cylinder 255
 Partition Disk #54, Partition #0
 Partition Size 334.00 GB (358,630,433,280 bytes)
 Partition Starting Offset 8,225,280 bytes
 Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 13
 SCSI Bus 0
 SCSI Logical Unit 12
 SCSI Port 9
 SCSI Target ID 0
 Sectors/Track 63
 Size 334.01 GB (358,638,658,560 bytes)
 Total Cylinders 43,602
 Total Sectors 700,466,130
 Total Tracks 11,118,510

Tracks/Cylinder 255
 Partition Disk #55, Partition #0
 Partition Size 334.00 GB (358,630,433,280 bytes)
 Partition Starting Offset 8,225,280 bytes
 Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 13
 SCSI Bus 0
 SCSI Logical Unit 1
 SCSI Port 3
 SCSI Target ID 0
 Sectors/Track 63
 Size 334.01 GB (358,638,658,560 bytes)
 Total Cylinders 43,602
 Total Sectors 700,466,130
 Total Tracks 11,118,510
 Tracks/Cylinder 255
 Partition Disk #7, Partition #0
 Partition Size 334.00 GB (358,630,433,280 bytes)
 Partition Starting Offset 8,225,280 bytes
 Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 13
 SCSI Bus 0
 SCSI Logical Unit 3
 SCSI Port 3

SCSI Target ID 0
 Sectors/Track 63
 Size 334.01 GB (358,638,658,560 bytes)
 Total Cylinders 43,602
 Total Sectors 700,466,130
 Total Tracks 11,118,510
 Tracks/Cylinder 255
 Partition Disk #8, Partition #0
 Partition Size 334.00 GB (358,630,433,280 bytes)
 Partition Starting Offset 8,225,280 bytes
 Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 13
 SCSI Bus 0
 SCSI Logical Unit 5
 SCSI Port 3
 SCSI Target ID 0
 Sectors/Track 63
 Size 334.01 GB (358,638,658,560 bytes)
 Total Cylinders 43,602
 Total Sectors 700,466,130
 Total Tracks 11,118,510
 Tracks/Cylinder 255
 Partition Disk #9, Partition #0
 Partition Size 334.00 GB (358,630,433,280 bytes)
 Partition Starting Offset 8,225,280 bytes
 Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512

Media Loaded Yes

Media Type Fixed hard disk

Partitions 13

SCSI Bus 0

SCSI Logical Unit 7

SCSI Port 3

SCSI Target ID 0

Sectors/Track 63

Size 334.01 GB (358,638,658,560 bytes)

Total Cylinders 43,602

Total Sectors 700,466,130

Total Tracks 11,118,510

Tracks/Cylinder 255

Partition Disk #10, Partition #0

Partition Size 334.00 GB (358,630,433,280 bytes)

Partition Starting Offset 8,225,280 bytes

Description Disk drive

Manufacturer (Standard disk drives)

Model IBM 1742-900 SCSI Disk Device

Bytes/Sector 512

Media Loaded Yes

Media Type Fixed hard disk

Partitions 13

SCSI Bus 0

SCSI Logical Unit 9

SCSI Port 3

SCSI Target ID 0

Sectors/Track 63

Size 334.01 GB (358,638,658,560 bytes)

Total Cylinders 43,602

Total Sectors 700,466,130

Total Tracks 11,118,510

Tracks/Cylinder 255

Partition Disk #11, Partition #0

Partition Size 334.00 GB (358,630,433,280 bytes)

Partition Starting Offset 8,225,280 bytes

Description Disk drive

Manufacturer (Standard disk drives)

Model IBM 1742-900 SCSI Disk Device

Bytes/Sector 512

Media Loaded Yes

Media Type Fixed hard disk

Partitions 13

SCSI Bus 0

SCSI Logical Unit 11

SCSI Port 3

SCSI Target ID 0

Sectors/Track 63

Size 334.01 GB (358,638,658,560 bytes)

Total Cylinders 43,602

Total Sectors 700,466,130

Total Tracks 11,118,510

Tracks/Cylinder 255

Partition Disk #12, Partition #0

Partition Size 334.00 GB (358,630,433,280 bytes)

Partition Starting Offset 8,225,280 bytes

Description Disk drive

Manufacturer (Standard disk drives)

Model IBM 1742-900 SCSI Disk Device

Bytes/Sector 512

Media Loaded Yes

Media Type Fixed hard disk

Partitions 1

SCSI Bus 0

SCSI Logical Unit 13

SCSI Port 3

SCSI Target ID 0

Sectors/Track 63

Size 300.61 GB (322,776,437,760 bytes)
 Total Cylinders 39,242
 Total Sectors 630,422,730
 Total Tracks 10,006,710
 Tracks/Cylinder 255
 Partition Disk #13, Partition #0
 Partition Size 300.60 GB (322,768,212,480 bytes)
 Partition Starting Offset 8,225,280 bytes
 Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 13
 SCSI Bus 0
 SCSI Logical Unit 0
 SCSI Port 5
 SCSI Target ID 0
 Sectors/Track 63
 Size 334.01 GB (358,638,658,560 bytes)
 Total Cylinders 43,602
 Total Sectors 700,466,130
 Total Tracks 11,118,510
 Tracks/Cylinder 255
 Partition Disk #21, Partition #0
 Partition Size 334.00 GB (358,630,433,280 bytes)
 Partition Starting Offset 8,225,280 bytes
 Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk

Partitions 13
 SCSI Bus 0
 SCSI Logical Unit 2
 SCSI Port 5
 SCSI Target ID 0
 Sectors/Track 63
 Size 334.01 GB (358,638,658,560 bytes)
 Total Cylinders 43,602
 Total Sectors 700,466,130
 Total Tracks 11,118,510
 Tracks/Cylinder 255
 Partition Disk #22, Partition #0
 Partition Size 334.00 GB (358,630,433,280 bytes)
 Partition Starting Offset 8,225,280 bytes
 Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 13
 SCSI Bus 0
 SCSI Logical Unit 4
 SCSI Port 5
 SCSI Target ID 0
 Sectors/Track 63
 Size 334.01 GB (358,638,658,560 bytes)
 Total Cylinders 43,602
 Total Sectors 700,466,130
 Total Tracks 11,118,510
 Tracks/Cylinder 255
 Partition Disk #23, Partition #0
 Partition Size 334.00 GB (358,630,433,280 bytes)
 Partition Starting Offset 8,225,280 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 12
 SCSI Bus 0
 SCSI Logical Unit 6
 SCSI Port 5
 SCSI Target ID 0
 Sectors/Track 63
 Size 334.01 GB (358,638,658,560 bytes)
 Total Cylinders 43,602
 Total Sectors 700,466,130
 Total Tracks 11,118,510
 Tracks/Cylinder 255
 Partition Disk #24, Partition #0
 Partition Size 334.00 GB (358,630,433,280 bytes)
 Partition Starting Offset 8,225,280 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 13
 SCSI Bus 0
 SCSI Logical Unit 8
 SCSI Port 5
 SCSI Target ID 0
 Sectors/Track 63

Size 334.01 GB (358,638,658,560 bytes)
 Total Cylinders 43,602
 Total Sectors 700,466,130
 Total Tracks 11,118,510
 Tracks/Cylinder 255
 Partition Disk #25, Partition #0
 Partition Size 334.00 GB (358,630,433,280 bytes)
 Partition Starting Offset 8,225,280 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 13
 SCSI Bus 0
 SCSI Logical Unit 10
 SCSI Port 5
 SCSI Target ID 0
 Sectors/Track 63
 Size 334.01 GB (358,638,658,560 bytes)

Total Cylinders 43,602
 Total Sectors 700,466,130
 Total Tracks 11,118,510
 Tracks/Cylinder 255
 Partition Disk #26, Partition #0
 Partition Size 334.00 GB (358,630,433,280 bytes)
 Partition Starting Offset 8,225,280 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512

Media Loaded Yes

Media Type Fixed hard disk

Partitions 13

SCSI Bus 0

SCSI Logical Unit 12

SCSI Port 5

SCSI Target ID 0

Sectors/Track 63

Size 334.01 GB (358,638,658,560 bytes)

Total Cylinders 43,602

Total Sectors 700,466,130

Total Tracks 11,118,510

Tracks/Cylinder 255

Partition Disk #27, Partition #0

Partition Size 334.00 GB (358,630,433,280 bytes)

Partition Starting Offset 8,225,280 bytes

Description Disk drive

Manufacturer (Standard disk drives)

Model IBM 1742-900 SCSI Disk Device

Bytes/Sector 512

Media Loaded Yes

Media Type Fixed hard disk

Partitions 13

SCSI Bus 0

SCSI Logical Unit 0

SCSI Port 2

SCSI Target ID 0

Sectors/Track 63

Size 334.01 GB (358,638,658,560 bytes)

Total Cylinders 43,602

Total Sectors 700,466,130

Total Tracks 11,118,510

Tracks/Cylinder 255

Partition Disk #0, Partition #0

Partition Size 334.00 GB (358,630,433,280 bytes)

Partition Starting Offset 8,225,280 bytes

Description Disk drive

Manufacturer (Standard disk drives)

Model IBM 1742-900 SCSI Disk Device

Bytes/Sector 512

Media Loaded Yes

Media Type Fixed hard disk

Partitions 13

SCSI Bus 0

SCSI Logical Unit 2

SCSI Port 2

SCSI Target ID 0

Sectors/Track 63

Size 334.01 GB (358,638,658,560 bytes)

Total Cylinders 43,602

Total Sectors 700,466,130

Total Tracks 11,118,510

Tracks/Cylinder 255

Partition Disk #1, Partition #0

Partition Size 334.00 GB (358,630,433,280 bytes)

Partition Starting Offset 8,225,280 bytes

Description Disk drive

Manufacturer (Standard disk drives)

Model IBM 1742-900 SCSI Disk Device

Bytes/Sector 512

Media Loaded Yes

Media Type Fixed hard disk

Partitions 13

SCSI Bus 0

SCSI Logical Unit 4

SCSI Port 2
 SCSI Target ID 0
 Sectors/Track 63
 Size 334.01 GB (358,638,658,560 bytes)
 Total Cylinders 43,602
 Total Sectors 700,466,130
 Total Tracks 11,118,510
 Tracks/Cylinder 255
 Partition Disk #2, Partition #0
 Partition Size 334.00 GB (358,630,433,280 bytes)
 Partition Starting Offset 8,225,280 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 13
 SCSI Bus 0

SCSI Logical Unit 6
 SCSI Port 2
 SCSI Target ID 0
 Sectors/Track 63
 Size 334.01 GB (358,638,658,560 bytes)
 Total Cylinders 43,602
 Total Sectors 700,466,130
 Total Tracks 11,118,510
 Tracks/Cylinder 255
 Partition Disk #3, Partition #0
 Partition Size 334.00 GB (358,630,433,280 bytes)
 Partition Starting Offset 8,225,280 bytes

Description Disk drive

Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 14
 SCSI Bus 0
 SCSI Logical Unit 8
 SCSI Port 2
 SCSI Target ID 0
 Sectors/Track 63
 Size 334.01 GB (358,638,658,560 bytes)

Total Cylinders 43,602
 Total Sectors 700,466,130
 Total Tracks 11,118,510
 Tracks/Cylinder 255
 Partition Disk #4, Partition #0
 Partition Size 334.00 GB (358,630,433,280 bytes)
 Partition Starting Offset 8,225,280 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 14
 SCSI Bus 0
 SCSI Logical Unit 10
 SCSI Port 2
 SCSI Target ID 0
 Sectors/Track 63
 Size 334.01 GB (358,638,658,560 bytes)
 Total Cylinders 43,602

Total Sectors 700,466,130
 Total Tracks 11,118,510
 Tracks/Cylinder 255
 Partition Disk #5, Partition #0
 Partition Size 334.00 GB (358,630,433,280 bytes)
 Partition Starting Offset 8,225,280 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 1

SCSI Bus 0
 SCSI Logical Unit 12
 SCSI Port 2
 SCSI Target ID 0
 Sectors/Track 63
 Size 300.61 GB (322,776,437,760 bytes)
 Total Cylinders 39,242
 Total Sectors 630,422,730
 Total Tracks 10,006,710
 Tracks/Cylinder 255

Partition Disk #6, Partition #0
 Partition Size 300.60 GB (322,768,212,480 bytes)
 Partition Starting Offset 8,225,280 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk

Partitions 13
 SCSI Bus 0
 SCSI Logical Unit 1
 SCSI Port 7
 SCSI Target ID 0
 Sectors/Track 63
 Size 334.01 GB (358,638,658,560 bytes)
 Total Cylinders 43,602
 Total Sectors 700,466,130
 Total Tracks 11,118,510
 Tracks/Cylinder 255
 Partition Disk #35, Partition #0
 Partition Size 334.00 GB (358,630,433,280 bytes)
 Partition Starting Offset 8,225,280 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk

Partitions 13
 SCSI Bus 0
 SCSI Logical Unit 3
 SCSI Port 7
 SCSI Target ID 0
 Sectors/Track 63
 Size 334.01 GB (358,638,658,560 bytes)
 Total Cylinders 43,602
 Total Sectors 700,466,130
 Total Tracks 11,118,510
 Tracks/Cylinder 255
 Partition Disk #36, Partition #0
 Partition Size 334.00 GB (358,630,433,280 bytes)

Partition Starting Offset 8,225,280 bytes

Description Disk drive

Manufacturer (Standard disk drives)

Model IBM 1742-900 SCSI Disk Device

Bytes/Sector 512

Media Loaded Yes

Media Type Fixed hard disk

Partitions 13

SCSI Bus 0

SCSI Logical Unit 5

SCSI Port 7

SCSI Target ID 0

Sectors/Track 63

Size 334.01 GB (358,638,658,560 bytes)

Total Cylinders 43,602

Total Sectors 700,466,130

Total Tracks 11,118,510

Tracks/Cylinder 255

Partition Disk #37, Partition #0

Partition Size 334.00 GB (358,630,433,280 bytes)

Partition Starting Offset 8,225,280 bytes

Description Disk drive

Manufacturer (Standard disk drives)

Model IBM 1742-900 SCSI Disk Device

Bytes/Sector 512

Media Loaded Yes

Media Type Fixed hard disk

Partitions 13

SCSI Bus 0

SCSI Logical Unit 7

SCSI Port 7

SCSI Target ID 0

Sectors/Track 63

Size 334.01 GB (358,638,658,560 bytes)

Total Cylinders 43,602

Total Sectors 700,466,130

Total Tracks 11,118,510

Tracks/Cylinder 255

Partition Disk #38, Partition #0

Partition Size 334.00 GB (358,630,433,280 bytes)

Partition Starting Offset 8,225,280 bytes

Description Disk drive

Manufacturer (Standard disk drives)

Model IBM 1742-900 SCSI Disk Device

Bytes/Sector 512

Media Loaded Yes

Media Type Fixed hard disk

Partitions 13

SCSI Bus 0

SCSI Logical Unit 9

SCSI Port 7

SCSI Target ID 0

Sectors/Track 63

Size 334.01 GB (358,638,658,560 bytes)

Total Cylinders 43,602

Total Sectors 700,466,130

Total Tracks 11,118,510

Tracks/Cylinder 255

Partition Disk #39, Partition #0

Partition Size 334.00 GB (358,630,433,280 bytes)

Partition Starting Offset 8,225,280 bytes

Description Disk drive

Manufacturer (Standard disk drives)

Model IBM 1742-900 SCSI Disk Device

Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 13
 SCSI Bus 0
 SCSI Logical Unit 11
 SCSI Port 7
 SCSI Target ID 0
 Sectors/Track 63
 Size 334.01 GB (358,638,658,560 bytes)
 Total Cylinders 43,602
 Total Sectors 700,466,130
 Total Tracks 11,118,510
 Tracks/Cylinder 255
 Partition Disk #40, Partition #0
 Partition Size 334.00 GB (358,630,433,280 bytes)
 Partition Starting Offset 8,225,280 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM 1742-900 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 1
 SCSI Bus 0
 SCSI Logical Unit 13
 SCSI Port 7
 SCSI Target ID 0
 Sectors/Track 63
 Size 300.61 GB (322,776,437,760 bytes)
 Total Cylinders 39,242
 Total Sectors 630,422,730
 Total Tracks 10,006,710

Tracks/Cylinder 255
 Partition Disk #41, Partition #0
 Partition Size 300.60 GB (322,768,212,480 bytes)
 Partition Starting Offset 8,225,280 bytes
 Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM ServeRAID SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 1
 SCSI Bus 0
 SCSI Logical Unit 0
 SCSI Port 11
 SCSI Target ID 0
 Sectors/Track 32
 Size 339.01 GB (364,013,158,400 bytes)
 Total Cylinders 173,575
 Total Sectors 710,963,200
 Total Tracks 22,217,600
 Tracks/Cylinder 128
 Partition Disk #59, Partition #0
 Partition Size 339.01 GB (364,011,044,864 bytes)
 Partition Starting Offset 16,384 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM ServeRAID SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 1
 SCSI Bus 0

SCSI Logical Unit 0
 SCSI Port 10
 SCSI Target ID 0
 Sectors/Track 32
 Size 33.90 GB (36,400,267,264 bytes)
 Total Cylinders 17,357
 Total Sectors 71,094,272
 Total Tracks 2,221,696
 Tracks/Cylinder 128
 Partition Disk #56, Partition #0
 Partition Size 33.90 GB (36,398,153,728 bytes)
 Partition Starting Offset 16,384 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM ServeRAID SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 1
 SCSI Bus 0

SCSI Logical Unit 0
 SCSI Port 10
 SCSI Target ID 1
 Sectors/Track 32
 Size 339.01 GB (364,013,158,400 bytes)
 Total Cylinders 173,575
 Total Sectors 710,963,200
 Total Tracks 22,217,600
 Tracks/Cylinder 128
 Partition Disk #57, Partition #0
 Partition Size 339.01 GB (364,011,044,864 bytes)
 Partition Starting Offset 16,384 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model IBM ServeRAID SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 1
 SCSI Bus 0
 SCSI Logical Unit 0
 SCSI Port 10
 SCSI Target ID 2
 Sectors/Track 32
 Size 410.19 GB (440,439,668,736 bytes)

Total Cylinders 210,018
 Total Sectors 860,233,728
 Total Tracks 26,882,304
 Tracks/Cylinder 128
 Partition Disk #58, Partition #0
 Partition Size 410.19 GB (440,435,474,432 bytes)
 Partition Starting Offset 2,097,152 bytes

[SCSI]

Item	Value
Name	QLogic Fibre Channel Adapter
Manufacturer	QLogic
Status	OK
PNP Device ID	PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&1070020&0&08
I/O Port	0x00002000-0x0000FE7F
Memory Address	0xF8A20000-0xF8A20FFF
IRQ Channel	IRQ 18
Name	QLogic Fibre Channel Adapter

Manufacturer QLogic
Status OK
PNP Device ID
PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&29E81982&0
&08
I/O Port 0x00002200-0x0000FE7F
Memory Address 0xF8B20000-0xF8B20FFF
IRQ Channel IRQ 19

Name QLogic Fibre Channel Adapter

Manufacturer QLogic
Status OK
PNP Device ID
PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&172E68DD&
0&08
I/O Port 0x00002400-0x0000FE7F
Memory Address 0xF8C20000-0xF8C20FFF
IRQ Channel IRQ 51

Name QLogic Fibre Channel Adapter

Manufacturer QLogic
Status OK
PNP Device ID
PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&474B838&0
&08
I/O Port 0x00002600-0x0000FE7F
Memory Address 0xF8D20000-0xF8D20FFF
IRQ Channel IRQ 52

Name IBM ServeRAID 6M Controller

Manufacturer IBM Corporation
Status OK
PNP Device ID
PCI\VEN_9005&DEV_0250&SUBSYS_02791014&REV_02\4&29C8B970&
0&4008
Memory Address 0xF8E08000-0xF8E08FFF
IRQ Channel IRQ 54

Name QLogic Fibre Channel Adapter

Manufacturer QLogic
Status OK
PNP Device ID
PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&300BC0BE&
0&08
I/O Port 0x00004200-0x0000FE7F
Memory Address 0xF9620000-0xF9620FFF
IRQ Channel IRQ 88

Name QLogic Fibre Channel Adapter

Manufacturer QLogic
Status OK
PNP Device ID
PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&72AA75C&0
&08
I/O Port 0x00004400-0x0000FE7F
Memory Address 0xF9720000-0xF9720FFF
IRQ Channel IRQ 89

Name QLogic Fibre Channel Adapter

Manufacturer QLogic
Status OK
PNP Device ID
PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&19E45801&0
&08
I/O Port 0x00004600-0x0000FE7F
Memory Address 0xF9820000-0xF9820FFF
IRQ Channel IRQ 121

Name QLogic Fibre Channel Adapter

Manufacturer QLogic
Status OK
PNP Device ID
PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&2C9E08A6&
0&08
I/O Port 0x00004800-0x0000FE7F
Memory Address 0xF9920000-0xF9920FFF

IRQ Channel IRQ 122

Name IBM ServeRAID 6M Controller

Manufacturer IBM Corporation

Status OK

PNP Device ID
PCI\VEN_9005&DEV_0250&SUBSYS_02791014&REV_02\4&3A1539FE&0&4008

Memory Address 0xF9A08000-0xF9A08FFF

IRQ Channel IRQ 124

[IDE]

Item Value

Name Standard Dual Channel PCI IDE Controller

Manufacturer (Standard IDE ATA/ATAPI controllers)

Status OK

PNP Device ID
PCI\VEN_1166&DEV_0213&SUBSYS_02121166&REV_A0\3&267A616A&0&79

I/O Port 0x00000700-0x0000070F

Name Primary IDE Channel

Manufacturer (Standard IDE ATA/ATAPI controllers)

Status OK

PNP Device ID PCI\IDE\DECHANNEL\4&101988B2&0&0

I/O Port 0x000001F0-0x000001F7

I/O Port 0x000003F6-0x000003F6

IRQ Channel IRQ 14

Name Secondary IDE Channel

Manufacturer (Standard IDE ATA/ATAPI controllers)

Status OK

PNP Device ID PCI\IDE\DECHANNEL\4&101988B2&0&1

I/O Port 0x00000170-0x00000177

I/O Port 0x00000376-0x00000376

[Printing]

Name Driver Port Name Server Name

[Problem Devices]

Device PNP Device ID Error Code

HID SYS USB\VID_04B3&PID_4001&MI_02\6&18E8217A&0&0002
The drivers for this device are not installed.

Not Available ACPI\IBM37D4\2&DABA3FF&0 The drivers for this device are not installed.

HID SYS USB\VID_04B3&PID_4001&MI_02\6&37646A5&0&0002
The drivers for this device are not installed.

[USB]

Device PNP Device ID

NEC PCI to USB Open Host Controller
PCI\VEN_1033&DEV_0035&SUBSYS_00351033&REV_43\3&267A616A&0&18

NEC PCI to USB Open Host Controller
PCI\VEN_1033&DEV_0035&SUBSYS_00351033&REV_43\3&267A616A&0&19

Standard Enhanced PCI to USB Host Controller
PCI\VEN_1033&DEV_00E0&SUBSYS_00E01033&REV_04\3&267A616A&0&1A

NEC PCI to USB Open Host Controller
PCI\VEN_1033&DEV_0035&SUBSYS_00351033&REV_43\3&A985F74&0&18

NEC PCI to USB Open Host Controller
PCI\VEN_1033&DEV_0035&SUBSYS_00351033&REV_43\3&A985F74&0&19

Standard Enhanced PCI to USB Host Controller
PCI\VEN_1033&DEV_00E0&SUBSYS_00E01033&REV_04\3&A985F74&0&1A

[Software Environment]

[System Drivers]

Name	Description	File	Type	Started	Start Mode	State	Status	Error Control	Accept	Pause	Accept	Stop
beep	Beep Driver	c:\windows\system32\drivers\beep.sys	Kernel Driver	Yes	System	Running	OK	Normal	No			
abiosdsk	Abiosdsk	Not Available	Kernel Driver	No	Auto	Running	OK	Normal	No			
acpi	Microsoft ACPI Driver	c:\windows\system32\drivers\acpi.sys	Kernel Driver	Yes	Auto	Running	OK	Normal	No			
acpiec	ACPIEC Driver	c:\windows\system32\drivers\acpiec.sys	Kernel Driver	No	Disabled	Running	OK	Normal	No			
adpu160m	adpu160m	Not Available	Kernel Driver	No	System	Running	OK	Normal	No			
adpu320	adpu320	Not Available	Kernel Driver	No	System	Running	OK	Normal	No			
afd	AFD Driver	c:\windows\system32\drivers\afd.sys	Kernel Driver	Yes	System	Running	OK	Normal	No			
aic78u2	aic78u2	Not Available	Kernel Driver	No	System	Running	OK	Normal	No			
aic78xx	aic78xx	Not Available	Kernel Driver	No	System	Running	OK	Normal	No			
aliide	Alilide	Not Available	Kernel Driver	No	System	Running	OK	Normal	No			
amdide	Amdlde	Not Available	Kernel Driver	No	System	Running	OK	Normal	No			
arc	arc	Not Available	Kernel Driver	No	System	Running	OK	Normal	No			
asynmac	RAS Asynchronous Media Driver	c:\windows\system32\drivers\asynmac.sys	Kernel Driver	No	System	Running	OK	Normal	No			
atapi	Standard IDE/ESDI Hard Disk Controller	c:\windows\system32\drivers\atapi.sys	Kernel Driver	Yes	System	Running	OK	Normal	No			
atdisk	Atdisk	Not Available	Kernel Driver	No	System	Running	OK	Normal	No			
ati2mtag	ati2mtag	c:\windows\system32\drivers\ati2mtag.sys	Kernel Driver	No	System	Running	OK	Normal	No			
atmarpc	ATM ARP Client Protocol	c:\windows\system32\drivers\atmarpc.sys	Kernel Driver	No	System	Running	OK	Normal	No			
audstub	Audio Stub Driver	c:\windows\system32\drivers\audstub.sys	Kernel Driver	No	System	Running	OK	Normal	No			
b57nd	Broadcom NetXtreme Gigabit Ethernet	c:\windows\system32\drivers\b57amd64.sys	Kernel Driver	Yes	System	Running	OK	Normal	No			
cdac15ba	CdaC15BA	c:\windows\system32\drivers\cdac15ba.sys	Kernel Driver	No	Auto	Running	OK	Normal	No			
cdad10ba	CdaD10BA	c:\windows\system32\drivers\cdad10ba.sys	Kernel Driver	No	Auto	Running	OK	Normal	No			
cdrom	CD-ROM Driver	c:\windows\system32\drivers\cdrom.sys	Kernel Driver	No	System	Running	OK	Normal	No			
changer	Changer System	Not Available	Kernel Driver	No	Ignore	Running	OK	Normal	No			
clusdisk	Cluster Disk Driver	c:\windows\system32\drivers\clusdisk.sys	Kernel Driver	No	Disabled	Running	OK	Normal	No			
cmdide	CmdIde	Not Available	Kernel Driver	No	System	Running	OK	Normal	No			
cpqciissm	cpqciissm	Not Available	Kernel Driver	No	System	Running	OK	Normal	No			
credisk	CRC Disk Filter Driver	c:\windows\system32\drivers\credisk.sys	Kernel Driver	Yes	System	Running	OK	Normal	No			
dfsdriver	DfsDriver	c:\windows\system32\drivers\dfs.sys	Kernel Driver	Yes	System	Running	OK	Normal	No			
disk	Disk Driver	c:\windows\system32\drivers\disk.sys	Kernel Driver	Yes	System	Running	OK	Normal	No			
dmboot	dmboot	c:\windows\system32\drivers\dmboot.sys	Kernel Driver	No	Disabled	Running	OK	Normal	No			
dmio	Logical Disk Manager Driver	c:\windows\system32\drivers\dmio.sys	Kernel Driver	Yes	System	Running	OK	Normal	No			
dmload	dmload	c:\windows\system32\drivers\dmload.sys	Kernel Driver	Yes	System	Running	OK	Normal	No			
dpti2o	dpti2o	Not Available	Kernel Driver	No	System	Running	OK	Normal	No			
elxstor	elxstor	Not Available	Kernel Driver	No	System	Running	OK	Normal	No			
em	em	???\c:\windows\system32\drivers\em.sys	Kernel Driver	No	Manual	Running	OK	Normal	No			

fastfat Driver No	Fastfat No	c:\windows\system32\drivers\fastfat.sys Disabled Stopped OK Normal	File System No	ipinip Kernel Driver No	IP in IP Tunnel Driver: No Manual Stopped OK Normal	c:\windows\system32\drivers\ipinip.sys
fdc Driver No	Fdc No	c:\windows\system32\drivers\fdc.sys System Stopped OK Ignore	Kernel No	ipnat Manual	IP Network Address Translator Stopped OK Normal	c:\windows\system32\drivers\ipnat.sys Kernel Driver No
fips Driver Yes	Fips Yes	c:\windows\system32\drivers\fips.sys System Running OK Normal	Kernel No	ipsec Kernel Driver No	IPSEC driver Yes System Running OK Normal	c:\windows\system32\drivers\ipsec.sys
flpydisk Driver No	Flpydisk No	c:\windows\system32\drivers\flpydisk.sys System Stopped OK Ignore	Kernel No	isapnp Boot	PnP ISA/EISA Bus Driver Running OK Critical	c:\windows\system32\drivers\isapnp.sys Kernel Driver Yes
fltmgr Driver Yes	FltMgr Yes	c:\windows\system32\drivers\fltmgr.sys Boot Running OK Normal	File System No	kbdclass System	Keyboard Class Driver Running OK Normal	c:\windows\system32\drivers\kbdclass.sys Kernel Driver Yes
ftdisk Boot	Volume Manager Driver Running OK	c:\windows\system32\drivers\ftdisk.sys Normal	Kernel Driver Yes	kbdhid Kernel Driver No	Keyboard HID Driver Yes System Running OK Ignore	c:\windows\system32\drivers\kbdhid.sys
gpc Manual	Generic Packet Classifier Running OK	c:\windows\system32\drivers\msgpc.sys Normal	Kernel Driver Yes	ksecdd Driver Yes	KSecDD Boot Running OK Normal	c:\windows\system32\drivers\ksecdd.sys Kernel No
hidusb Manual	Microsoft HID Class Driver Running OK	c:\windows\system32\drivers\hidusb.sys Ignore	Kernel Driver Yes	ksthunk Manual	Kernel Streaming WOW64 Thunk Service Running OK Normal	c:\windows\system32\drivers\ksthunk.sys Kernel Driver Yes
hpcisss Disabled	hpcisss Stopped	Not Available OK Normal	Kernel Driver No	lp6nds35 Disabled	lp6nds35 Not Available OK Normal	Kernel Driver No
http Driver No	HTTP No	c:\windows\system32\drivers\http.sys Manual Stopped OK Normal	Kernel No	mnmd Driver Yes	mnmd System Running OK Ignore	c:\windows\system32\drivers\mnmd.sys Kernel No
i2omgmt System	i2omgmt Stopped	Not Available OK Normal	Kernel Driver No	modem Driver No	Modem Manual Stopped OK Ignore	c:\windows\system32\drivers\modem.sys Kernel No
i804prt System	i804 Keyboard and PS/2 Mouse Running OK	c:\windows\system32\drivers\i804prt.sys Normal	Kernel Driver Yes	mouclass Kernel Driver No	Mouse Class Driver Yes System Running OK Normal	c:\windows\system32\drivers\mouclass.sys
iirsp Disabled	iirsp Stopped	Not Available OK Normal	Kernel Driver No	mouhid Kernel Driver No	Mouse HID Driver Yes Manual Running OK Ignore	c:\windows\system32\drivers\mouhid.sys
imapi System	CD-Burning Filter Driver Stopped OK	c:\windows\system32\drivers\imapi.sys Normal	Kernel Driver No	mountmgr Kernel Driver No	Mount Point Manager Yes Boot Running OK Normal	c:\windows\system32\drivers\mountmgr.sys
intelide Disabled	IntelIde Stopped	Not Available OK Normal	Kernel Driver No	mraid35x Disabled	mraid35x Not Available OK Normal	Kernel Driver No
intelppm Kernel Driver No	Intel Processor Driver Yes	c:\windows\system32\drivers\intelppm.sys Manual Running OK Normal	Kernel Driver Normal	mrxdav Manual	WebDav Client Redirector Stopped OK Normal	c:\windows\system32\drivers\mrxdav.sys File System Driver No
ip6fw Manual	IPv6 Windows Firewall Driver Stopped OK	c:\windows\system32\drivers\ip6fw.sys Normal	Kernel Driver No	mrxsmb Driver Yes	MRXSMB System Running OK Normal	c:\windows\system32\drivers\mrxsmb.sys File System No
ipfilterdriver Manual	IP Traffic Filter Driver Stopped OK	c:\windows\system32\drivers\ipfltdrv.sys Normal	Kernel Driver No			

msfs Driver Yes	Msfs Yes	c:\windows\system32\drivers\msfs.sys System Running OK Normal	File System No	pciide Driver Yes	PCIIde Yes	c:\windows\system32\drivers\pciide.sys Boot Running OK Normal	Kernel No
mssmbios Manual	Microsoft System Management BIOS Driver Running OK Normal No Yes	c:\windows\system32\drivers\mssmbios.sys Kernel Driver Yes	File System Yes	pcmcia Driver No	Pcmcia No	c:\windows\system32\drivers\pcmcia.sys Disabled Stopped OK Normal	Kernel No
mup Driver Yes	Mup Yes	c:\windows\system32\drivers\mup.sys Boot Running OK Normal	File System No	pdcomp Manual	PDCOMP Stopped OK	Not Available Ignore Kernel Driver No No	No
ndis Kernel Driver No	NDIS System Driver Yes	c:\windows\system32\drivers\ndis.sys Boot Running OK Normal	File System Normal	pdframe No	PDFRAME Manual Stopped	Not Available OK Ignore Kernel Driver No No	No
ndistapi Manual	Remote Access NDIS TAPI Driver Running OK Normal No Yes	c:\windows\system32\drivers\ndistapi.sys Kernel Driver Yes	File System Yes	pdreli Manual	PDRELI Stopped OK	Not Available Ignore Kernel Driver No No	No
ndisuio Manual	NDIS Usermode I/O Protocol Stopped OK Normal No No	c:\windows\system32\drivers\ndisuio.sys Kernel Driver No No	File System No	pdframe No	PDRFRAME Manual Stopped	Not Available OK Ignore Kernel Driver No No	No
ndiswan Manual	Remote Access NDIS WAN Driver Running OK Normal No Yes	c:\windows\system32\drivers\ndiswan.sys Kernel Driver Yes	File System Yes	pnpmem Manual	Microsoft Memory Module Driver Running OK Normal	c:\windows\system32\drivers\pnpmem.sys Kernel Driver No Yes	Yes
ndproxy Kernel Driver No	NDIS Proxy Yes	c:\windows\system32\drivers\ndproxy.sys Manual Running OK Normal	File System Normal	pptpminiport Manual	WAN Miniport (PPTP) Running OK Normal	c:\windows\system32\drivers\rasppptp.sys Kernel Driver No Yes	Yes
netbios File System No	NetBIOS Interface Yes	c:\windows\system32\drivers\netbios.sys System Running OK Normal	File System Normal	ptilink Manual	Direct Parallel Link Driver Running OK Normal	c:\windows\system32\drivers\ptilink.sys Kernel Driver No Yes	Yes
netbt Kernel Driver No	NetBios over Tcpip Yes	c:\windows\system32\drivers\netbt.sys System Running OK Normal	File System Normal	ql2300 Driver Yes	ql2300 Yes	c:\windows\system32\drivers\ql2300.sys Boot Running OK Normal	Kernel No
nfrd960 Driver Yes	nfrd960 Yes	c:\windows\system32\drivers\nfrd960.sys Boot Running OK Normal	Kernel No	qldirect Driver Yes	qldirect Yes	c:\windows\system32\drivers\qldirect.sys Auto Running OK Normal	Kernel No
npfs Driver Yes	Npfs Yes	c:\windows\system32\drivers\npfs.sys System Running OK Normal	File System No	rasacd System	Remote Access Auto Connection Driver Running OK Normal	c:\windows\system32\drivers\rasacd.sys Kernel Driver No Yes	Yes
ntfs Driver Yes	Ntfs Yes	c:\windows\system32\drivers\ntfs.sys Disabled Running OK Normal	File System No	rasl2tp Manual	WAN Miniport (L2TP) Running OK Normal	c:\windows\system32\drivers\rasl2tp.sys Kernel Driver No Yes	Yes
null Driver Yes	Null Yes	c:\windows\system32\drivers\null.sys System Running OK Normal	Kernel No	rasppoe Manual	Remote Access PPPOE Driver Running OK Normal	c:\windows\system32\drivers\rasppoe.sys Kernel Driver No Yes	Yes
parport Driver No	Parport No	c:\windows\system32\drivers\parport.sys Manual Stopped OK Ignore	Kernel No	raspti Kernel Driver No	Direct Parallel Yes	c:\windows\system32\drivers\raspti.sys Manual Running OK Normal	Kernel Normal
partmgr Kernel Driver No	Partition Manager Yes	c:\windows\system32\drivers\partmgr.sys Boot Running OK Normal	File System Normal	rdbss Driver Yes	Rdbss Yes	c:\windows\system32\drivers\rdbss.sys System Running OK Normal	File System No
pci Kernel Driver No	PCI Bus Driver Yes	c:\windows\system32\drivers\pci.sys Boot Running OK Critical	File System Critical	rdpcdd Driver Yes	RDPCDD Yes	c:\windows\system32\drivers\rdpcdd.sys System Running OK Ignore	Kernel No
				rdpdr Manual	Terminal Server Device Redirector Driver Running OK Normal	c:\windows\system32\drivers\rdpdr.sys Kernel Driver No Yes	Yes

rdpwd	RDPWD	c:\windows\system32\drivers\rdpwd.sys	Kernel Driver	Yes	Manual	Running	OK	Ignore	No	udfs	Udfs	c:\windows\system32\drivers\udfs.sys	File System	Driver	No	Disabled	Stopped	OK	Normal	No	
redbook	Digital CD Audio Playback Filter Driver	c:\windows\system32\drivers\redbook.sys	Kernel Driver	Yes	Running	OK	Normal	No	Yes	ultra	ultra	Not Available	Kernel Driver	Disabled	Stopped	OK	Normal	No	No	No	
secdrv	Security Driver	c:\windows\system32\drivers\secdrv.sys	Kernel Driver	No	Yes	Auto	Running	OK	Normal	update	Microcode Update Driver	c:\windows\system32\drivers\update.sys	Kernel Driver	Manual	Running	OK	Normal	No	Yes	Yes	
serenum	Serenum Filter Driver	c:\windows\system32\drivers\serenum.sys	Kernel Driver	No	Yes	Manual	Running	OK	Normal	usbccgp	Microsoft USB Generic Parent Driver	c:\windows\system32\drivers\usbccgp.sys	Kernel Driver	Manual	Running	OK	Normal	No	Yes	Yes	
serial	Serial port driver	c:\windows\system32\drivers\serial.sys	Kernel Driver	No	Yes	System	Running	OK	Ignore	usbehci	Microsoft USB 2.0 Enhanced Host Controller Miniport Driver	c:\windows\system32\drivers\usbehci.sys	Kernel Driver	Manual	Running	OK	Normal	No	Yes	Yes	
sfloppy	High-Capacity Floppy Disk Drive	c:\windows\system32\drivers\sfloppy.sys	Kernel Driver	No	Manual	Stopped	OK	Normal	No	usbhub	USB2 Enabled Hub	c:\windows\system32\drivers\usbhub.sys	Kernel Driver	No	Yes	Yes	Manual	Running	OK	Normal	
simbad	Simbad	Not Available	Kernel Driver	No	Disabled	Stopped	OK	Normal	No	usbohci	Microsoft USB Open Host Controller Miniport Driver	c:\windows\system32\drivers\usbohci.sys	Kernel Driver	Manual	Running	OK	Normal	No	Yes	Yes	
srv	Srv	c:\windows\system32\drivers\srv.sys	File System	Driver	Yes	Manual	Running	OK	Normal	usbstor	USB Mass Storage Driver	c:\windows\system32\drivers\usbstor.sys	Kernel Driver	Manual	Stopped	OK	Normal	No	No	No	
swenum	Software Bus Driver	c:\windows\system32\drivers\swenum.sys	Kernel Driver	No	Yes	Manual	Running	OK	Normal	vgasave	VGA Display Controller.	c:\windows\system32\drivers\vga.sys	Kernel Driver	System	Running	OK	Ignore	No	Yes	Yes	
symc8xx	symc8xx	Not Available	Kernel Driver	No	Disabled	Stopped	OK	Normal	No	viaide	ViaIde	Not Available	Kernel Driver	Disabled	Stopped	OK	Normal	No	No	No	
symmpi	symmpi	Not Available	Kernel Driver	No	Disabled	Stopped	OK	Normal	No	volsnap	Storage volumes	c:\windows\system32\drivers\volsnap.sys	Kernel Driver	No	Yes	Yes	Boot	Running	OK	Normal	
sym_hi	sym_hi	Not Available	Kernel Driver	No	Disabled	Stopped	OK	Normal	No	wanarp	Remote Access IP ARP Driver	c:\windows\system32\drivers\wanarp.sys	Kernel Driver	Manual	Running	OK	Normal	No	Yes	Yes	
sym_u3	sym_u3	Not Available	Kernel Driver	No	Disabled	Stopped	OK	Normal	No	wdica	WDICA	Not Available	Kernel Driver	Manual	Stopped	OK	Ignore	No	No	No	
tcpip	TCP/IP Protocol Driver	c:\windows\system32\drivers\tcpip.sys	Kernel Driver	System	Running	OK	Normal	No	Yes	wlbs	Network Load Balancing	c:\windows\system32\drivers\wlbs.sys	Kernel Driver	Manual	Stopped	OK	Normal	No	No	No	
tdpipe	TDPIPE	c:\windows\system32\drivers\tdpipe.sys	Kernel Driver	No	No	Manual	Stopped	OK	Ignore	[Signed Drivers]											
tdtcp	TDTCP	c:\windows\system32\drivers\tdtcp.sys	Kernel Driver	Yes	Manual	Running	OK	Ignore	No	Device Name		Signed	Device Class	Driver Version							
termdd	Terminal Device Driver	c:\windows\system32\drivers\termdd.sys	Kernel Driver	System	Running	OK	Normal	No	Yes	Driver Date		Manufacturer	INF Name	Driver Name							
toside	TosIde	Not Available	Kernel Driver	No	Disabled	Stopped	OK	Normal	No	Device ID											
											Microsoft System Management BIOS Driver					No	SYSTEM				
											5.2.3790.1830					10/1/2002 (Standard system devices)			machine.inf		
											Not Available					ROOT\SYSTEM\0002					

Microcode Update Device	No	SYSTEM	5.2.3790.1830		VGA Display Controller.	Not Available	LEGACYDRIVER	
10/1/2002 (Standard system devices)		machine.inf	Not Available		Not Available	Not Available	Not Available	Not Available
Available		ROOT\SYSTEM\0001			Available	Not Available	ROOT\LEGACY_VGASAVE\0000	
Plug and Play Software Device Enumerator	No	SYSTEM	5.2.3790.1830		TDTCP	Not Available	LEGACYDRIVER	Not Available
5.2.3790.1830	10/1/2002 (Standard system devices)	machine.inf	Not Available		Not Available	Not Available	Not Available	Not Available
Not Available		ROOT\SYSTEM\0000			Available	ROOT\LEGACY_TDTCP\0000		
Terminal Server Mouse Driver	No	SYSTEM	5.2.3790.1830		TCP/IP Protocol Driver	Not Available	LEGACYDRIVER	
10/1/2002 (Standard system devices)		machine.inf	Not Available		Not Available	Not Available	Not Available	Not Available
Available		ROOT\RDP_MOUSE\0000			Available	Not Available	ROOT\LEGACY_TCPIP\0000	
Terminal Server Keyboard Driver	No	SYSTEM	5.2.3790.1830		Security Driver	Not Available	LEGACYDRIVER	Not Available
5.2.3790.1830	10/1/2002 (Standard system devices)	machine.inf	Not Available		Available	Not Available	Not Available	Not Available
Not Available		ROOT\RDP_KEYBOARD\0000			Not Available	ROOT\LEGACY_SECDRV\0000		
Terminal Server Device Redirector	No	SYSTEM	5.2.3790.1830		RDPWD	Not Available	LEGACYDRIVER	Not Available
5.2.3790.1830	10/1/2002 (Standard system devices)	machine.inf	Not Available		Not Available	Not Available	Not Available	Not Available
Not Available		ROOT\RDPDR\0000			Available	ROOT\LEGACY_RDPWD\0000		
Direct Parallel	No	NET	5.2.3790.1830	10/1/2002	RDPCDD	Not Available	LEGACYDRIVER	Not Available
Microsoft netrasa.inf	Not Available		ROOT\MS_PTMINIPORT\0000		Not Available	Not Available	Not Available	Not Available
WAN Miniport (PPTP)	No	NET	5.2.3790.1830		Available	ROOT\LEGACY_RDPCDD\0000		
10/1/2002 Microsoft netrasa.inf	Not Available		ROOT\MS_PPTPMINIPORT\0000					
WAN Miniport (PPPOE)	No	NET	5.2.3790.1830		Remote Access Auto Connection Driver	Not Available	LEGACYDRIVER	Not Available
10/1/2002 Microsoft netrasa.inf	Not Available		ROOT\MS_PPPOEMINIPORT\0000		Available	Not Available	Not Available	Not Available
WAN Miniport (IP)	No	NET	5.2.3790.1830	10/1/2002	LEGACYDRIVER	Not Available	Not Available	Not Available
Microsoft netrasa.inf	Not Available		ROOT\MS_NDISWANIP\0000		Available	Not Available	Not Available	Not Available
WAN Miniport (L2TP)	No	NET	5.2.3790.1830		ROOT\LEGACY_RASACD\0000			
10/1/2002 Microsoft netrasa.inf	Not Available		ROOT\MS_L2TPMINIPORT\0000					
Video Codecs	No	MEDIA	5.2.3790.1830	10/1/2002	qldirect	Not Available	LEGACYDRIVER	Not Available
(Standard system devices)		wave.inf	Not Available		Not Available	Not Available	Not Available	Not Available
Available		ROOT\MEDIA\MS_MMVID			Available	ROOT\LEGACY_QLDIRECT\0000		
Legacy Video Capture Devices	No	MEDIA	5.2.3790.1830		Partition Manager	Not Available	LEGACYDRIVER	Not Available
10/1/2002 (Standard system devices)		wave.inf	Not Available		Available	Not Available	Not Available	Not Available
Available		ROOT\MEDIA\MS_MMVCD			Not Available	ROOT\LEGACY_PARTMGR\0000		
Media Control Devices	No	MEDIA	5.2.3790.1830		Null	Not Available	LEGACYDRIVER	Not Available
10/1/2002 (Standard system devices)		wave.inf	Not Available		Not Available	Not Available	Not Available	Not Available
Available		ROOT\MEDIA\MS_MMMCI			Available	ROOT\LEGACY_NULL\0000		
Legacy Audio Drivers	No	MEDIA	5.2.3790.1830	10/1/2002	NetBios over Tcpi	Not Available	LEGACYDRIVER	Not Available
(Standard system devices)		wave.inf	Not Available		Available	Not Available	Not Available	Not Available
Available		ROOT\MEDIA\MS_MMDRV			Not Available	ROOT\LEGACY_NETBT\0000		
Audio Codecs	No	MEDIA	5.2.3790.1830	10/1/2002	NDProxy	Not Available	LEGACYDRIVER	Not Available
(Standard system devices)		wave.inf	Not Available		Not Available	Not Available	Not Available	Not Available
Available		ROOT\MEDIA\MS_MMACM			Available	ROOT\LEGACY_NDPROXY\0000		
Remote Access IP ARP Driver	Not Available				NDIS Usermode I/O Protocol	Not Available	LEGACYDRIVER	
Not Available	Not Available				Not Available	Not Available	Not Available	Not Available
Available		ROOT\LEGACY_WANARP\0000			Available	Not Available	ROOT\LEGACY_NDISUIO\0000	
volsnap	Not Available	LEGACYDRIVER	Not Available		Remote Access NDIS TAPI Driver	Not Available	LEGACYDRIVER	Not Available
Not Available	Not Available				Available	Not Available	Not Available	Not Available
Available		ROOT\LEGACY_VOLSNAP\0000			Available	Not Available	Not Available	Not Available
					ROOT\LEGACY_NDISTAPI\0000			
					NDIS System Driver	Not Available	LEGACYDRIVER	Not Available
					Available	Not Available	Not Available	Not Available
					Not Available	ROOT\LEGACY_NDIS\0000		
					mountmgr	Not Available	LEGACYDRIVER	Not Available
					Not Available	Not Available	Not Available	Not Available
					Available	Not Available	Not Available	Not Available
					Available	ROOT\LEGACY_MOUNTMGR\0000		

mmdd Not Available LEGACYDRIVER Not Available
 Not Available Not Available Not Available Not
 Available ROOT\LEGACY_MMDD\0000

ksecdd Not Available LEGACYDRIVER Not Available
 Not Available Not Available Not Available Not
 Available ROOT\LEGACY_KSECDD\0000

IPSEC driver Not Available LEGACYDRIVER Not
 Available Not Available Not Available Not Available
 Not Available ROOT\LEGACY_IPSEC\0000

IP Network Address Translator Not Available LEGACYDRIVER
 Not Available Not Available Not Available Not
 Available Not Available ROOT\LEGACY_IPNAT\0000

Generic Packet Classifier Not Available LEGACYDRIVER
 Not Available Not Available Not Available Not
 Available Not Available ROOT\LEGACY_GPC\0000

Fips Not Available LEGACYDRIVER Not Available
 Not Available Not Available Not Available Not
 Available ROOT\LEGACY_FIPS\0000

em Not Available LEGACYDRIVER Not Available
 Not Available Not Available Not Available Not
 Available ROOT\LEGACY_EM\0000

dmload Not Available LEGACYDRIVER Not Available
 Not Available Not Available Not Available Not
 Available ROOT\LEGACY_DMLOAD\0000

dmboot Not Available LEGACYDRIVER Not Available
 Not Available Not Available Not Available Not
 Available ROOT\LEGACY_DMBOOT\0000

CRC Disk Filter Driver Not Available LEGACYDRIVER
 Not Available Not Available Not Available Not
 Available Not Available ROOT\LEGACY_CRCDISK\0000

CdaD10BA Not Available LEGACYDRIVER Not
 Available Not Available Not Available Not Available
 Not Available ROOT\LEGACY_CDAD10BA\0000

CdaC15BA Not Available LEGACYDRIVER Not
 Available Not Available Not Available Not Available
 Not Available ROOT\LEGACY_CDAC15BA\0000

Beep Not Available LEGACYDRIVER Not Available
 Not Available Not Available Not Available Not
 Available ROOT\LEGACY_BEEP\0000

AFD Not Available LEGACYDRIVER Not Available
 Not Available Not Available Not Available Not
 Available ROOT\LEGACY_AFD\0000

Generic volume No VOLUME 5.2.3790.1830 10/1/2002
 Microsoft volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURE7673F100OFFSET400
 0LENGTH54C0BFC000

Generic volume No VOLUME 5.2.3790.1830 10/1/2002
 Microsoft volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF45OFFSET7E
 0000LENGTH4B267B9400

Generic volume No VOLUME 5.2.3790.1830 10/1/2002
 Microsoft volume.inf Not Available

STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF44OFFSET16
 FBABC000LENGTH3C84DBE400

Generic volume No VOLUME 5.2.3790.1830 10/1/2002
 Microsoft volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF44OFFSET16
 E7146C00LENGTH1496D600

Generic volume No VOLUME 5.2.3790.1830 10/1/2002
 Microsoft volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF44OFFSET16
 4AAF6E00LENGTH9C648000

Generic volume No VOLUME 5.2.3790.1830 10/1/2002
 Microsoft volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF44OFFSET15
 FAC5A800LENGTH4FE94800

Generic volume No VOLUME 5.2.3790.1830 10/1/2002
 Microsoft volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF44OFFSET15
 B3AF0600LENGTH47162400

Generic volume No VOLUME 5.2.3790.1830 10/1/2002
 Microsoft volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF44OFFSET10
 4B2CF600LENGTH568819200

Generic volume No VOLUME 5.2.3790.1830 10/1/2002
 Microsoft volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF44OFFSET8C
 85B5200LENGTH782D12600

Generic volume No VOLUME 5.2.3790.1830 10/1/2002
 Microsoft volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF44OFFSET6E
 4FC000LENGTH85A0B1400

Generic volume No VOLUME 5.2.3790.1830 10/1/2002
 Microsoft volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF44OFFSET59
 B86C00LENGTH1496D600

Generic volume No VOLUME 5.2.3790.1830 10/1/2002
 Microsoft volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF44OFFSET1F
 68600LENGTH57C16800

Generic volume No VOLUME 5.2.3790.1830 10/1/2002
 Microsoft volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF44OFFSET17
 90400LENGTH7D0400

Generic volume No VOLUME 5.2.3790.1830 10/1/2002
 Microsoft volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF44OFFSETFB
 8200LENGTH7D0400

Generic volume No VOLUME 5.2.3790.1830 10/1/2002
 Microsoft volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF44OFFSET7E
 0000LENGTH7D0400

Generic volume No VOLUME 5.2.3790.1830 10/1/2002
 Microsoft volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF43OFFSET16
 FBABC000LENGTH3C84DBE400

Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF42OFFSET16E7146C00LENGTH1496D600
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF43OFFSET16E7146C00LENGTH1496D600
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF43OFFSET164AAF6E00LENGTH9C648000
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF42OFFSET15FAC5A800LENGTH4FE94800
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF42OFFSET15B3AF0600LENGTH47162400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF42OFFSET104B2CF600LENGTH568819200
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF42OFFSET8C85B5200LENGTH782D12600
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF42OFFSET6E4FC000LENGTH85A0B1400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF42OFFSET59B86C00LENGTH1496D600
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF42OFFSET1F68600LENGTH57C16800
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF42OFFSET1790400LENGTH7D0400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF42OFFSETFB8200LENGTH7D0400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF42OFFSET7E0000LENGTH7D0400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF41OFFSET16FBABC000LENGTH3C84DBE400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF41OFFSET16E7146C00LENGTH1496D600
Microsoft	volume.inf	Not Available		

Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE43987A02OFFSET164 AAAF6E00LENGTH9C648000
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF41OFFSET16 4AAF6E00LENGTH9C648000
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE43987A02OFFSET15F AC5A800LENGTH4FE94800
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF41OFFSET15 FAC5A800LENGTH4FE94800
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE43987A02OFFSET15B 3AF0600LENGTH47162400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF41OFFSET15 B3AF0600LENGTH47162400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE43987A02OFFSET104 B2CF600LENGTH568819200
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE43987A02OFFSET8C8 5B5200LENGTH782D12600
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE43987A02OFFSET6E4 FC000LENGTH85A0B1400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE43987A02OFFSET59B 86C00LENGTH1496D600
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE43987A02OFFSET1F6 8600LENGTH57C16800
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE43987A02OFFSET179 0400LENGTH7D0400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE43987A02OFFSETFB 8200LENGTH7D0400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE43987A02OFFSET7E0 000LENGTH7D0400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF3FOFFSET16 FBABC000LENGTH3C84DBE400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE43987A02OFFSET16E 7146C00LENGTH1496D600
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF3FOFFSET16 4AAF6E00LENGTH9C648000
Microsoft volume.inf	Not Available			

Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF4COFFSET15 FAC5A800LENGTH4FE94800
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF3FOFFSET15 FAC5A800LENGTH4FE94800
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF4COFFSET15 B3AF0600LENGTH47162400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF3FOFFSET10 4B2CF600LENGTH568819200
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF3FOFFSET8C 85B5200LENGTH782D12600
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF3FOFFSET6E 4FC000LENGTH85A0B1400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF4COFFSET59 B86C00LENGTH1496D600
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF3FOFFSET1F 68600LENGTH57C16800
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF4COFFSET17 90400LENGTH7D0400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF3FOFFSETFB 8200LENGTH7D0400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF4COFFSET7E 0000LENGTH7D0400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF4BOFFSET16 FBABC000LENGTH1D4C120600
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF4BOFFSET16 E7146C00LENGTH1496D600
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF4BOFFSET16 4AAF6E00LENGTH9C648000
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF4BOFFSET15 FAC5A800LENGTH4FE94800
Microsoft	volume.inf	Not Available		

Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF4AOFFSET15B3AF0600LENGTH47162400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF4BOFFSET15B3AF0600LENGTH47162400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF4BOFFSET104B2CF600LENGTH568819200
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF4BOFFSET8C85B5200LENGTH782D12600
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF4BOFFSET6E4FC000LENGTH85A0B1400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF4BOFFSET59B86C00LENGTH1496D600
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF4BOFFSET1F68600LENGTH57C16800
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF4BOFFSET1790400LENGTH7D0400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF4BOFFSETFB8200LENGTH7D0400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF4BOFFSET7E0000LENGTH7D0400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF4AOFFSET16FBABC000LENGTH1D4C120600
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF4AOFFSET16E7146C00LENGTH1496D600
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF4AOFFSET164AAF6E00LENGTH9C648000
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF4AOFFSET15FAC5A800LENGTH4FE94800
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF4AOFFSET15B3AF0600LENGTH47162400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF4AOFFSET104B2CF600LENGTH568819200
Microsoft volume.inf	Not Available			

Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF48OFFSET8C85B5200LENGTH782D12600
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF49OFFSET8C85B5200LENGTH782D12600
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF49OFFSET6E4FC000LENGTH85A0B1400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF48OFFSET59B86C00LENGTH1496D600
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF49OFFSET1F68600LENGTH57C16800
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF48OFFSET1790400LENGTH7D0400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF48OFFSETFB8200LENGTH7D0400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF48OFFSET7E0000LENGTH7D0400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF47OFFSET16FBABC000LENGTH3C84DBE400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF47OFFSET16E7146C00LENGTH1496D600
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF47OFFSET164AAF6E00LENGTH9C648000
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF47OFFSET15FAC5A800LENGTH4FE94800
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF48OFFSET15B3AF0600LENGTH47162400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF47OFFSET104B2CF600LENGTH568819200
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF47OFFSET8C85B5200LENGTH782D12600
Microsoft volume.inf	Not Available			

Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF46OFFSET6E4FC00LENGTH85A0B1400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF47OFFSET6E4FC00LENGTH85A0B1400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF47OFFSET59B86C00LENGTH1496D600
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF46OFFSET1F68600LENGTH57C16800
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF46OFFSET1790400LENGTH7D0400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF47OFFSETFB8200LENGTH7D0400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF46OFFSET7E0000LENGTH7D0400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B39OFFSET7E0000LENGTH4B267B9400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B38OFFSET16FBABC000LENGTH1D4C120600
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B38OFFSET16E7146C00LENGTH1496D600
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B38OFFSET16E7146C00LENGTH1496D600
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B38OFFSET164AAF6E00LENGTH9C648000
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF46OFFSET15FAC5A800LENGTH4FE94800
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B38OFFSET15B3AF0600LENGTH47162400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC1DF46OFFSET104B2CF600LENGTH568819200
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B38OFFSET104B2CF600LENGTH568819200
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B38OFFSET8C85B5200LENGTH782D12600
Microsoft	volume.inf	Not Available		

Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B37OFFSET6E4FC000LENGTH85A0B1400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B38OFFSET6E4FC000LENGTH85A0B1400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B37OFFSET59B86C00LENGTH1496D600
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B38OFFSET59B86C00LENGTH1496D600
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B37OFFSET1F68600LENGTH57C16800
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B37OFFSET1790400LENGTH7D0400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B38OFFSETFB8200LENGTH7D0400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B37OFFSET7E0000LENGTH7D0400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B36OFFSET16FBABC000LENGTH1D4C120600
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B36OFFSET16E7146C00LENGTH1496D600
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B36OFFSET164AAF6E00LENGTH9C648000
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B36OFFSET15FAC5A800LENGTH4FE94800
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B36OFFSET15B3AF0600LENGTH47162400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B37OFFSET104B2CF600LENGTH568819200
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B36OFFSET8C85B5200LENGTH782D12600
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B36OFFSET6E4FC000LENGTH85A0B1400
Microsoft	volume.inf	Not Available		

Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B35OFFSET59B86C00LENGTH1496D600
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B36OFFSET59B86C00LENGTH1496D600
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B35OFFSET1F68600LENGTH57C16800
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B35OFFSET1790400LENGTH7D0400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B36OFFSETFB8200LENGTH7D0400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B35OFFSET7E0000LENGTH7D0400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B34OFFSET16FBABC00LENGTH1D4C120600
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B34OFFSET16E7146C00LENGTH1496D600
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B34OFFSET164AAF6E00LENGTH9C648000
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B34OFFSET15FAC5A800LENGTH4FE94800
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B34OFFSET15B3AF0600LENGTH47162400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B34OFFSET104B2CF600LENGTH568819200
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B35OFFSET8C85B5200LENGTH782D12600
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B34OFFSET6E4FC00LENGTH85A0B1400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B34OFFSET59B86C00LENGTH1496D600
Microsoft volume.inf	Not Available			

Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B33OFFSET1F68600LENGTH57C16800
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B34OFFSET1F68600LENGTH57C16800
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B33OFFSET1790400LENGTH7D0400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B34OFFSET1790400LENGTH7D0400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B34OFFSETFB8200LENGTH7D0400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B33OFFSET7E0000LENGTH7D0400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B32OFFSET16FBABC000LENGTH1D4C120600
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B33OFFSET16E7146C00LENGTH1496D600
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B33OFFSET164AAF6E00LENGTH9C648000
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B32OFFSET15FAC5A800LENGTH4FE94800
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B32OFFSET15B3AF0600LENGTH47162400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B32OFFSET104B2CF600LENGTH568819200
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B32OFFSET8C85B5200LENGTH782D12600
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B32OFFSET6E4FC000LENGTH85A0B1400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B32OFFSET59B86C00LENGTH1496D600
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B32OFFSET1F68600LENGTH57C16800
Microsoft volume.inf	Not Available			

Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B31OFFSET1790400LENGTH7D0400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B32OFFSET1790400LENGTH7D0400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B32OFFSETFB8200LENGTH7D0400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B31OFFSET7E0000LENGTH7D0400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B30OFFSET16FBABC000LENGTH3C84DBE400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B30OFFSET16E7146C00LENGTH1496D600
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B30OFFSET164AAF6E00LENGTH9C648000
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B30OFFSET15FAC5A800LENGTH4FE94800
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B30OFFSET15B3AF0600LENGTH47162400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B30OFFSET104B2CF600LENGTH568819200
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B30OFFSET8C85B5200LENGTH782D12600
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B30OFFSET6E4FC000LENGTH85A0B1400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B30OFFSET59B86C00LENGTH1496D600
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B30OFFSET1F68600LENGTH57C16800
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B30OFFSET1790400LENGTH7D0400
Microsoft	volume.inf	Not Available		

Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B2FOFFSETFB8200LENGTH7D0400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B30OFFSETFB0000LENGTH7D0400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B2FOFFSET7E0000LENGTH7D0400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B2FOFFSET16FBABC000LENGTH3C84DBE400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B2FOFFSET16E7146C00LENGTH1496D600
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B2FOFFSET164AAF6E00LENGTH9C648000
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B2FOFFSET15FAC5A800LENGTH4FE94800
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B2FOFFSET15B3AF0600LENGTH47162400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B2FOFFSET104B2CF600LENGTH568819200
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B2FOFFSET8C85B5200LENGTH782D12600
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B2FOFFSET6E4FC000LENGTH85A0B1400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B2FOFFSET59B86C00LENGTH1496D600
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B2FOFFSET1F68600LENGTH57C16800
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B2FOFFSET1790400LENGTH7D0400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B2FOFFSETFB8200LENGTH7D0400
Microsoft volume.inf	Not Available			

Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE549D037DOFFSET7E0000LENGTH7D0400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B2EOFFSET7E0000LENGTH7D0400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE549D037DOFFSET16FBABC000LENGTH3C84DBE400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE549D037DOFFSET16E7146C00LENGTH1496D600
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE549D037DOFFSET164AAF6E00LENGTH9C648000
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B2COFFSET15FAC5A800LENGTH4FE94800
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE549D037DOFFSET15B3AF0600LENGTH47162400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE549D037DOFFSET104B2CF600LENGTH568819200
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE549D037DOFFSET8C85B5200LENGTH782D12600
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE549D037DOFFSET6E4FC000LENGTH85A0B1400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE549D037DOFFSET59B86C00LENGTH1496D600
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE549D037DOFFSET1F68600LENGTH57C16800
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE549D037DOFFSET1790400LENGTH7D0400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE549D037DOFFSETFB8200LENGTH7D0400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AC04B2COFFSET7E0000LENGTH7D0400
Microsoft	volume.inf	Not Available		

Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BACOFFSET1F68600LENGTH57C16800
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE542FB102OFFSET204000LENGTH668BFFC000
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE542F0C01OFFSET4000LENGTH54C0BFC000
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE4D5F8E00OFFSET4000LENGTH8797FC000
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BADOFFSET7E0000LENGTH4B267B9400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BACOFFSET16FBABC000LENGTH3C84DBE400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BACOFFSET16E7146C00LENGTH1496D600
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BACOFFSET164AAF6E00LENGTH9C648000
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BACOFFSET15FAC5A800LENGTH4FE94800
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BACOFFSET15B3AF0600LENGTH47162400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BACOFFSET104B2CF600LENGTH568819200
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BACOFFSET8C85B5200LENGTH782D12600
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BACOFFSET6E4FC000LENGTH85A0B1400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BACOFFSET59B86C00LENGTH1496D600
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BACOFFSET1F68600LENGTH57C16800
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BACOFFSET1790400LENGTH7D0400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BACOFFSETB8200LENGTH7D0400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BACOFFSET7E0000LENGTH7D0400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BACOFFSET16FBABC000LENGTH3C84DBE400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BACOFFSET16E7146C00LENGTH1496D600
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BACOFFSET164AAF6E00LENGTH9C648000
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BACOFFSET15FAC5A800LENGTH4FE94800
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BACOFFSET15B3AF0600LENGTH47162400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BACOFFSET104B2CF600LENGTH568819200
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BACOFFSET8C85B5200LENGTH782D12600
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BACOFFSET6E4FC000LENGTH85A0B1400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BACOFFSET59B86C00LENGTH1496D600
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BACOFFSET1F68600LENGTH57C16800
Microsoft volume.inf	Not Available			

Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BAAOFFSET1790400LENGTH7D0400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BABOFFSET1790400LENGTH7D0400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BABOFFSETB8200LENGTH7D0400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BAAOFFSET7E0000LENGTH7D0400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA9OFFSET16FBABC000LENGTH3C84DBE400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA9OFFSET16E7146C00LENGTH1496D600
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA9OFFSET164AAF6E00LENGTH9C648000
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA9OFFSET15FAC5A800LENGTH4FE94800
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA9OFFSET15B3AF0600LENGTH47162400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BAAOFFSET104B2CF600LENGTH568819200
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA9OFFSET8C85B5200LENGTH782D12600
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA9OFFSET6E4FC000LENGTH85A0B1400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA9OFFSET59B86C00LENGTH1496D600
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA9OFFSET1F68600LENGTH57C16800
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA9OFFSET1790400LENGTH7D0400
Microsoft	volume.inf	Not Available		

Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA5OFFSETF B8200LENGTH7D0400
Microsoft volume.inf	Not Available			
STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA7OFFSET7				
E0000LENGTH7D0400				
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA5OFFSET7 E0000LENGTH7D0400
Microsoft volume.inf	Not Available			
STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA6OFFSET7				
E0000LENGTH4B267B9400				
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE50224415OFFSET16F BABC000LENGTH3C84DBE400
Microsoft volume.inf	Not Available			
STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA5OFFSET16				
FBABC000LENGTH1D4C120600				
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE50224415OFFSET16E 7146C00LENGTH1496D600
Microsoft volume.inf	Not Available			
STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA5OFFSET16				
E7146C00LENGTH1496D600				
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE50224415OFFSET164 AAF6E00LENGTH9C648000
Microsoft volume.inf	Not Available			
STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA5OFFSET16				
4AAF6E00LENGTH9C648000				
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE50224415OFFSET15F AC5A800LENGTH4FE94800
Microsoft volume.inf	Not Available			
STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA5OFFSET15				
FAC5A800LENGTH4FE94800				
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE50224415OFFSET15B 3AF0600LENGTH47162400
Microsoft volume.inf	Not Available			
STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA5OFFSET15				
B3AF0600LENGTH47162400				
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE50224415OFFSET104 B2CF600LENGTH568819200
Microsoft volume.inf	Not Available			
STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA5OFFSET10				
4B2CF600LENGTH568819200				
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE50224415OFFSET8C8 5B5200LENGTH782D12600
Microsoft volume.inf	Not Available			
STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA5OFFSET8				
C85B5200LENGTH782D12600				
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE50224415OFFSET6E4 FC000LENGTH85A0B1400
Microsoft volume.inf	Not Available			
STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA5OFFSET6				
E4FC000LENGTH85A0B1400				
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE50224415OFFSET59B 86C00LENGTH1496D600
Microsoft volume.inf	Not Available			
STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA5OFFSET59				
B86C00LENGTH1496D600				
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE50224415OFFSET1F6 8600LENGTH57C16800
Microsoft volume.inf	Not Available			
STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA5OFFSET1F				
68600LENGTH57C16800				
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE50224415OFFSET179 0400LENGTH7D0400
Microsoft volume.inf	Not Available			
STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA5OFFSET17				
90400LENGTH7D0400				
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE50224415OFFSETFB8 200LENGTH7D0400
Microsoft volume.inf	Not Available			

Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA3OFFSET7E0000LENGTH7D0400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE50224415OFFSET7E0000LENGTH7D0400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA2OFFSET16FBABC000LENGTH1D4C120600
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA3OFFSET16E7146C00LENGTH1496D600
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA2OFFSET164AAF6E00LENGTH9C648000
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA2OFFSET15FAC5A800LENGTH4FE94800
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA2OFFSET15B3AF0600LENGTH47162400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA2OFFSET104B2CF600LENGTH568819200
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA2OFFSET8C85B5200LENGTH782D12600
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA2OFFSET6E4FC000LENGTH85A0B1400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA2OFFSET59B86C00LENGTH1496D600
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA2OFFSET1F68600LENGTH57C16800
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA2OFFSET1790400LENGTH7D0400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA2OFFSETFB8200LENGTH7D0400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA2OFFSET7E0000LENGTH7D0400
Microsoft	volume.inf	Not Available		

Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA0OFFSET16 FBABC000LENGTH1D4C120600
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA1OFFSET16 E7146C00LENGTH1496D600
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA0OFFSET16 4AAF6E00LENGTH9C648000
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA0OFFSET15 FAC5A800LENGTH4FE94800
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA0OFFSET15 B3AF0600LENGTH47162400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA0OFFSET10 4B2CF600LENGTH568819200
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA0OFFSET8 C85B5200LENGTH782D12600
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA0OFFSET6 E4FC000LENGTH85A0B1400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA0OFFSET59 B86C00LENGTH1496D600
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA0OFFSET1F 68600LENGTH57C16800
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA0OFFSET17 90400LENGTH7D0400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA0OFFSETF B8200LENGTH7D0400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3ABE5BA0OFFSETF E0000LENGTH7D0400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C5OFFSETFE 0000LENGTH4B267B9400
Microsoft	volume.inf	Not Available		

Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C3OFFSET16
Microsoft	volume.inf	Not Available		FBABC000LENGTH3C84DBE400
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C4OFFSET16
Microsoft	volume.inf	Not Available		FBABC000LENGTH3C84DBE400
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C4OFFSET16
Microsoft	volume.inf	Not Available		E7146C00LENGTH1496D600
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C4OFFSET16
Microsoft	volume.inf	Not Available		4AAF6E00LENGTH9C648000
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C3OFFSET15
Microsoft	volume.inf	Not Available		FAC5A800LENGTH4FE94800
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C3OFFSET15
Microsoft	volume.inf	Not Available		B3AF0600LENGTH47162400
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C3OFFSET10
Microsoft	volume.inf	Not Available		4B2CF600LENGTH568819200
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C3OFFSET8C
Microsoft	volume.inf	Not Available		85B5200LENGTH782D12600
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C3OFFSET6E
Microsoft	volume.inf	Not Available		4FC000LENGTH85A0B1400
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C3OFFSET59
Microsoft	volume.inf	Not Available		B86C00LENGTH1496D600
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C3OFFSET1F
Microsoft	volume.inf	Not Available		68600LENGTH57C16800
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C3OFFSET17
Microsoft	volume.inf	Not Available		90400LENGTH7D0400
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C3OFFSETFB
Microsoft	volume.inf	Not Available		8200LENGTH7D0400
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C3OFFSET7E
Microsoft	volume.inf	Not Available		0000LENGTH7D0400
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C2OFFSET16
Microsoft	volume.inf	Not Available		FBABC000LENGTH3C84DBE400

Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C1OFFSET16E7146C00LENGTH1496D600
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C2OFFSET16E7146C00LENGTH1496D600
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C2OFFSET164AAF6E00LENGTH9C648000
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C1OFFSET15FAC5A800LENGTH4FE94800
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C2OFFSET15B3AF0600LENGTH47162400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C2OFFSET104B2CF600LENGTH568819200
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C1OFFSET8C85B5200LENGTH782D12600
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C1OFFSET6E4FC000LENGTH85A0B1400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C1OFFSET59B86C00LENGTH1496D600
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C1OFFSET1F68600LENGTH57C16800
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C1OFFSET1790400LENGTH7D0400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C1OFFSETFB8200LENGTH7D0400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C2OFFSET7E0000LENGTH7D0400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C0OFFSET16FBABC000LENGTH3C84DBE400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C0OFFSET16E7146C00LENGTH1496D600
Microsoft volume.inf	Not Available			

Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126BFOFFSET164AAF6E00LENGTH9C648000
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C0OFFSET1644AF6E00LENGTH9C648000
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C0OFFSET15FAC5A800LENGTH4FE94800
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C0OFFSET15B3AF0600LENGTH47162400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C0OFFSET104B2CF600LENGTH568819200
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C0OFFSET8C85B5200LENGTH782D12600
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C0OFFSET6E4FC000LENGTH85A0B1400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126BFOFFSET59B86C00LENGTH1496D600
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126BFOFFSET1F68600LENGTH57C16800
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C0OFFSET1790400LENGTH7D0400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C0OFFSETFB8200LENGTH7D0400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA126C0OFFSET7E0000LENGTH7D0400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA513A1OFFSET7E0000LENGTH4B267B9400
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA513A0OFFSET3447BE4400LENGTH1F38C96000
Microsoft	volume.inf	Not Available		
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA513A0OFFSET16FBABC000LENGTH1D4C120600
Microsoft	volume.inf	Not Available		

Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139FOFFSET16
Microsoft	volume.inf	Not Available		FBABC000LENGTH1D4C120600
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA513A0OFFSET16
Microsoft	volume.inf	Not Available		E7146C00LENGTH1496D600
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA513A0OFFSET16
Microsoft	volume.inf	Not Available		E7146C00LENGTH1496D600
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139FOFFSET16
Microsoft	volume.inf	Not Available		4AAF6E00LENGTH9C648000
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA513A0OFFSET15
Microsoft	volume.inf	Not Available		FAC5A800LENGTH4FE94800
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA513A0OFFSET15
Microsoft	volume.inf	Not Available		B3AF0600LENGTH47162400
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA513A0OFFSET10
Microsoft	volume.inf	Not Available		4B2CF600LENGTH568819200
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139FOFFSET15
Microsoft	volume.inf	Not Available		B3AF0600LENGTH47162400
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139FOFFSET10
Microsoft	volume.inf	Not Available		4B2CF600LENGTH568819200
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139FOFFSET8C
Microsoft	volume.inf	Not Available		85B5200LENGTH782D12600
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139FOFFSET8C
Microsoft	volume.inf	Not Available		85B5200LENGTH782D12600
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139FOFFSET6E
Microsoft	volume.inf	Not Available		4FC000LENGTH85A0B1400
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA513A0OFFSET59
Microsoft	volume.inf	Not Available		B86C00LENGTH1496D600
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA513A0OFFSET1F
Microsoft	volume.inf	Not Available		68600LENGTH57C16800
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA513A0OFFSET17
Microsoft	volume.inf	Not Available		90400LENGTH7D0400
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA513A0OFFSETFB
Microsoft	volume.inf	Not Available		8200LENGTH7D0400
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA513A0OFFSET7E
Microsoft	volume.inf	Not Available		0000LENGTH7D0400
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139FOFFSET34
Microsoft	volume.inf	Not Available		47BE4400LENGTH1F38C96000
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139FOFFSET16F
Microsoft	volume.inf	Not Available		BABC000LENGTH3C84DBE400

Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139DOFFSET16E7146C00LENGTH1496D600
Microsoft volume.inf	Not Available			
STORAGE\VOLUME\1&30A96598&0&SIGNATUREC7D9CD4OFFSET16E7146C00LENGTH1496D600				
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139DOFFSET1644AAF6E00LENGTH9C648000
Microsoft volume.inf	Not Available			
STORAGE\VOLUME\1&30A96598&0&SIGNATUREC7D9CD4OFFSET164AAF6E00LENGTH9C648000				
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139DOFFSET15FAC5A800LENGTH4FE94800
Microsoft volume.inf	Not Available			
STORAGE\VOLUME\1&30A96598&0&SIGNATUREC7D9CD4OFFSET15FAC5A800LENGTH4FE94800				
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139DOFFSET15B3AF0600LENGTH47162400
Microsoft volume.inf	Not Available			
STORAGE\VOLUME\1&30A96598&0&SIGNATUREC7D9CD4OFFSET15B3AF0600LENGTH47162400				
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139DOFFSET104B2CF600LENGTH568819200
Microsoft volume.inf	Not Available			
STORAGE\VOLUME\1&30A96598&0&SIGNATUREC7D9CD4OFFSET104B2CF600LENGTH568819200				
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139DOFFSET8C85B5200LENGTH782D12600
Microsoft volume.inf	Not Available			
STORAGE\VOLUME\1&30A96598&0&SIGNATUREC7D9CD4OFFSET8C85B5200LENGTH782D12600				
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139DOFFSET6E4FC000LENGTH85A0B1400
Microsoft volume.inf	Not Available			
STORAGE\VOLUME\1&30A96598&0&SIGNATUREC7D9CD4OFFSET6E4FC000LENGTH85A0B1400				
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139DOFFSET59B86C00LENGTH1496D600
Microsoft volume.inf	Not Available			
STORAGE\VOLUME\1&30A96598&0&SIGNATUREC7D9CD4OFFSET59B86C00LENGTH1496D600				
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139DOFFSET1F68600LENGTH57C16800
Microsoft volume.inf	Not Available			
STORAGE\VOLUME\1&30A96598&0&SIGNATUREC7D9CD4OFFSET1F68600LENGTH57C16800				
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139DOFFSET1790400LENGTH7D0400
Microsoft volume.inf	Not Available			
STORAGE\VOLUME\1&30A96598&0&SIGNATUREC7D9CD4OFFSET1790400LENGTH7D0400				
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139DOFFSETFB8200LENGTH7D0400
Microsoft volume.inf	Not Available			
STORAGE\VOLUME\1&30A96598&0&SIGNATUREC7D9CD4OFFSETFB8200LENGTH7D0400				
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139DOFFSET7E0000LENGTH7D0400
Microsoft volume.inf	Not Available			
STORAGE\VOLUME\1&30A96598&0&SIGNATUREC7D9CD4OFFSET7E0000LENGTH7D0400				
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139COFFSET16FBABC000LENGTH3C84DBE400
Microsoft volume.inf	Not Available			
STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139DOFFSET16FBABC000LENGTH3C84DBE400				
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139COFFSET16E7146C00LENGTH1496D600
Microsoft volume.inf	Not Available			
STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139COFFSET16E7146C00LENGTH1496D600				

Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139BOFFSET164AAF6E00LENGTH9C648000
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139COFFSET164AAF6E00LENGTH9C648000
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139BOFFSET15FAC5A800LENGTH4FE94800
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139COFFSET15B3AF0600LENGTH47162400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139BOFFSET104B2CF600LENGTH568819200
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139COFFSET8C85B5200LENGTH782D12600
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139COFFSET6E4FC000LENGTH85A0B1400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139BOFFSET59B86C00LENGTH1496D600
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139BOFFSET1F68600LENGTH57C16800
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139BOFFSET1790400LENGTH7D0400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139BOFFSETFB8200LENGTH7D0400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139BOFFSET7E0000LENGTH7D0400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139BOFFSET16FBABC000LENGTH3C84DBE400
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139BOFFSET16E7146C00LENGTH1496D600
Microsoft volume.inf	Not Available			
Generic volume	No	VOLUME 5.2.3790.1830	10/1/2002	STORAGE\VOLUME\1&30A96598&0&SIGNATURE3AA5139BOFFSET16E7146C00LENGTH1496D600
Microsoft volume.inf	Not Available			
Volume Manager (Standard system devices)	No	SYSTEM 5.2.3790.1830	10/1/2002	machine.inf Not Available
Logical Disk Manager (Standard system devices)	No	SYSTEM 5.2.3790.1830	10/1/2002	machine.inf Not Available
IBM Dummy Device	No	SYSTEM 5.2.3790.1830	10/1/2002	machine.inf Not Available
SCSI BRIDGE	VEN_IBM&PROD_DUMMY_DEVICE&REV_7.10\5&2444047F&0&300			
SCSI Processor Device	No	SYSTEM 5.2.3790.1830	10/1/2002	machine.inf Not Available

SCSI\PROCESSOR&VEN_IBM&PROD_EXP400___S320&REV_D110\5&2444047F&0&2F0

SCSI Processor Device No SYSTEM 5.2.3790.1830
10/1/2002 IBM scsidev.inf Not Available
SCSI\PROCESSOR&VEN_IBM&PROD_EXP400___S320&REV_D110\5&2444047F&0&1F0

Disk drive No DISKDRIVE 5.2.3790.1830 10/1/2002
(Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.10\5&2444047F&0&000

IBM ServeRAID 6M Controller No SCSIADAPTER
5.2.3790.1830 10/1/2002 IBM Corporation pnpscsi.inf Not
Available
PCI\VEN_9005&DEV_0250&SUBSYS_02791014&REV_02\4&3A1539FE&0&4008

PCI standard PCI-to-PCI bridge No SYSTEM 5.2.3790.1830
10/1/2002 (Standard system devices) machine.inf Not
Available
PCI\VEN_1014&DEV_01A7&SUBSYS_00000000&REV_02\3&36B90202&0&08

PCI standard host CPU bridge No SYSTEM 5.2.3790.1830
10/1/2002 (Standard system devices) machine.inf Not
Available
PCI\VEN_1014&DEV_02A1&SUBSYS_00000000&REV_02\3&36B90202&0&00

PCI bus No SYSTEM 5.2.3790.1830 10/1/2002 (Standard
system devices) machine.inf Not Available
ACPI\PNP0A03\17

PCI standard host CPU bridge No SYSTEM 5.2.3790.1830
10/1/2002 (Standard system devices) machine.inf Not
Available
PCI\VEN_1014&DEV_02A1&SUBSYS_00000000&REV_02\3&3BCEF44&0&00

PCI bus No SYSTEM 5.2.3790.1830 10/1/2002 (Standard
system devices) machine.inf Not Available
ACPI\PNP0A03\16

Qlogic processor device No SYSTEM 5.2.3790.1830
10/1/2002 QLOGIC scsidev.inf Not Available
SCSI\PROCESSOR&VEN_QLOGIC&PROD_PSEUDO_DEVICE&REV_4&29C1039C&0&07F0

Disk drive No DISKDRIVE 5.2.3790.1830 10/1/2002
(Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&29C1039C&0&00D

Disk drive No DISKDRIVE 5.2.3790.1830 10/1/2002
(Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&29C1039C&0&00B

Disk drive No DISKDRIVE 5.2.3790.1830 10/1/2002
(Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&29C1039C&0&009

Disk drive No DISKDRIVE 5.2.3790.1830 10/1/2002
(Standard disk drives) disk.inf Not Available

SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&29C1039C&0&007

Disk drive No DISKDRIVE 5.2.3790.1830 10/1/2002
(Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&29C1039C&0&005

Disk drive No DISKDRIVE 5.2.3790.1830 10/1/2002
(Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&29C1039C&0&003

Disk drive No DISKDRIVE 5.2.3790.1830 10/1/2002
(Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&29C1039C&0&001

QLogic Fibre Channel Adapter No SCSIADAPTER 9.0.1.64
2/18/2005 QLogic oem0.inf Not Available
PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&2C9E08A6&0&08

PCI standard host CPU bridge No SYSTEM 5.2.3790.1830
10/1/2002 (Standard system devices) machine.inf Not
Available
PCI\VEN_1014&DEV_02A1&SUBSYS_00000000&REV_02\3&2C9E08A6&0&00

PCI bus No SYSTEM 5.2.3790.1830 10/1/2002 (Standard
system devices) machine.inf Not Available
ACPI\PNP0A03\15

Qlogic processor device No SYSTEM 5.2.3790.1830
10/1/2002 QLOGIC scsidev.inf Not Available
SCSI\PROCESSOR&VEN_QLOGIC&PROD_PSEUDO_DEVICE&REV_4&3343B59&0&07F0

Disk drive No DISKDRIVE 5.2.3790.1830 10/1/2002
(Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&3343B59&0&00C

Disk drive No DISKDRIVE 5.2.3790.1830 10/1/2002
(Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&3343B59&0&00A

Disk drive No DISKDRIVE 5.2.3790.1830 10/1/2002
(Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&3343B59&0&008

Disk drive No DISKDRIVE 5.2.3790.1830 10/1/2002
(Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&3343B59&0&006

Disk drive No DISKDRIVE 5.2.3790.1830 10/1/2002
(Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&3343B59&0&004

Disk drive No DISKDRIVE 5.2.3790.1830 10/1/2002
(Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&3343B59&0&002

Disk drive No DISKDRIVE 5.2.3790.1830 10/1/2002
(Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&3343B59&0&000

QLogic Fibre Channel Adapter No SCSIADAPTER 9.0.1.64
2/18/2005 QLogic oem0.inf Not Available

PCI bus No SYSTEM 5.2.3790.1830 10/1/2002 (Standard system devices) machine.inf Not Available ACPI\PNP0A03\11	PCI\VEN_1033&DEV_0035&SUBSYS_00351033&REV_43\3&A985F74&0&18
Serverworks Champion CSB6 - SouthBridge 6 LPC No SYSTEM 5.2.3790.1830 10/1/2002 ServerWorks (RCC) machine.inf Not Available PCI\VEN_1166&DEV_0227&SUBSYS_00000000&REV_00\3&A985F74&0&7B	PCI standard host CPU bridge No SYSTEM 5.2.3790.1830 10/1/2002 (Standard system devices) machine.inf Not Available PCI\VEN_1014&DEV_02A1&SUBSYS_00000000&REV_02\3&A985F74&0&00
ServerWorks Champion CSB6 - SouthBridge 6 No SYSTEM 5.2.3790.1830 10/1/2002 ServerWorks (RCC) machine.inf Not Available PCI\VEN_1166&DEV_0203&SUBSYS_00000000&REV_A0\3&A985F74&0&78	PCI bus No SYSTEM 5.2.3790.1830 10/1/2002 (Standard system devices) machine.inf Not Available ACPI\PNP0A03\10
USB Root Hub No USB 5.2.3790.1830 10/1/2002 (Standard USB Host Controller) usbport.inf Not Available USB\ROOT_HUB20\4&26176FDA&0	Memory Module No MEMORY5.2.3790.1830 10/1/2002 Microsoft memory.inf Not Available ACPI\PNP0C80\2
Standard Enhanced PCI to USB Host Controller No USB 5.2.3790.1830 10/1/2002 (Standard USB Host Controller) usbport.inf Not Available PCI\VEN_1033&DEV_00E0&SUBSYS_00E01033&REV_04\3&A985F74&0&1A	Intel Processor No PROCESSOR 5.2.3790.1830 10/1/2002 Intel cpu.inf Not Available ACPI\GENUINEINTEL_-_EM64T_FAMILY_15_MODEL_4\15
HID SYS Not Available Not Available Not Available Not Available Not Available Not Available Not Available Available USB\VID_04B3&PID_4001&MI_02\6&37646A5&0&0002	Intel Processor No PROCESSOR 5.2.3790.1830 10/1/2002 Intel cpu.inf Not Available ACPI\GENUINEINTEL_-_EM64T_FAMILY_15_MODEL_4\14
HID-compliant mouseNo MOUSE 5.2.3790.1830 10/1/2002 Microsoft msmouse.inf Not Available HID\VID_04B3&PID_4001&MI_01\7&3BDBD69&0&0000	Intel Processor No PROCESSOR 5.2.3790.1830 10/1/2002 Intel cpu.inf Not Available ACPI\GENUINEINTEL_-_EM64T_FAMILY_15_MODEL_4\13
USB Human Interface Device No HIDCLASS 5.2.3790.1830 10/1/2002 (Standard system devices) input.inf Not Available USB\VID_04B3&PID_4001&MI_01\6&37646A5&0&0001	Intel Processor No PROCESSOR 5.2.3790.1830 10/1/2002 Intel cpu.inf Not Available ACPI\GENUINEINTEL_-_EM64T_FAMILY_15_MODEL_4\11
HID Keyboard Device No KEYBOARD 5.2.3790.1830 10/1/2002 (Standard keyboards) keyboard.inf Not Available HID\VID_04B3&PID_4001&MI_00\7&F725ED5&0&0000	Intel Processor No PROCESSOR 5.2.3790.1830 10/1/2002 Intel cpu.inf Not Available ACPI\GENUINEINTEL_-_EM64T_FAMILY_15_MODEL_4\10
USB Human Interface Device No HIDCLASS 5.2.3790.1830 10/1/2002 (Standard system devices) input.inf Not Available USB\VID_04B3&PID_4001&MI_00\6&37646A5&0&0000	Intel Processor No PROCESSOR 5.2.3790.1830 10/1/2002 Intel cpu.inf Not Available ACPI\GENUINEINTEL_-_EM64T_FAMILY_15_MODEL_4\8
USB Composite Device No USB 5.2.3790.1830 10/1/2002 (Standard USB Host Controller) usb.inf Not Available USB\VID_04B3&PID_4001\000D6046D537	IBM Dummy Device No SYSTEM 5.2.3790.1830 10/1/2002 IBM scsidev.inf Not Available SCSI\BRIDGE&VEN_IBM&PROD_DUMMY_DEVICE&REV_7.10\5&804C5&0&300
USB Root Hub No USB 5.2.3790.1830 10/1/2002 (Standard USB Host Controller) usbport.inf Not Available USB\ROOT_HUB\4&14A79F01&0	SCSI Processor Device No SYSTEM 5.2.3790.1830 10/1/2002 IBM scsidev.inf Not Available SCSI\PROCESSOR&VEN_IBM&PROD_EXP400___S320&REV_D110\5&804C5&0&2F0
NEC PCI to USB Open Host Controller No USB 5.2.3790.1830 10/1/2002 NEC usbport.inf Not Available PCI\VEN_1033&DEV_0035&SUBSYS_00351033&REV_43\3&A985F74&0&19	SCSI Processor Device No SYSTEM 5.2.3790.1830 10/1/2002 IBM scsidev.inf Not Available SCSI\PROCESSOR&VEN_IBM&PROD_EXP400___S320&REV_D110\5&804C5&0&1F0
USB Root Hub No USB 5.2.3790.1830 10/1/2002 (Standard USB Host Controller) usbport.inf Not Available USB\ROOT_HUB\4&27614FA6&0	Disk drive No DISKDRIVE 5.2.3790.1830 10/1/2002 (Standard disk drives) disk.inf Not Available SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.10\5&804C5&0&020
NEC PCI to USB Open Host Controller No USB 5.2.3790.1830 10/1/2002 NEC usbport.inf Not Available	

Disk drive No DISKDRIVE 5.2.3790.1830 10/1/2002
(Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.10\5&804C5&0&010

Disk drive No DISKDRIVE 5.2.3790.1830 10/1/2002
(Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_SERVERAID&REV_6.10\5&804C5&0&00

IBM ServeRAID 6M Controller No SCSIADAPTER
5.2.3790.1830 10/1/2002 IBM Corporation pnpscsi.inf Not Available
PCI\VEN_9005&DEV_0250&SUBSYS_02791014&REV_02\4&29C8B970&0&4008

PCI standard PCI-to-PCI bridge No SYSTEM 5.2.3790.1830
10/1/2002 (Standard system devices) machine.inf Not Available
PCI\VEN_1014&DEV_01A7&SUBSYS_00000000&REV_02\3&20FEA912&0&08

PCI standard host CPU bridge No SYSTEM 5.2.3790.1830
10/1/2002 (Standard system devices) machine.inf Not Available
PCI\VEN_1014&DEV_02A1&SUBSYS_00000000&REV_02\3&20FEA912&0&00

PCI bus No SYSTEM 5.2.3790.1830 10/1/2002 (Standard system devices) machine.inf Not Available
ACPI\PNP0A03\7

PCI standard host CPU bridge No SYSTEM 5.2.3790.1830
10/1/2002 (Standard system devices) machine.inf Not Available
PCI\VEN_1014&DEV_02A1&SUBSYS_00000000&REV_02\3&E44F86D&0&00

PCI bus No SYSTEM 5.2.3790.1830 10/1/2002 (Standard system devices) machine.inf Not Available
ACPI\PNP0A03\6

Qlogic processor device No SYSTEM 5.2.3790.1830
10/1/2002 QLOGIC scsidev.inf Not Available
SCSI\PROCESSOR&VEN_QLOGIC&PROD_PSEUDO_DEVICE&REV_4&1D1C1BB3&0&07F0

Disk drive No DISKDRIVE 5.2.3790.1830 10/1/2002
(Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&1D1C1BB3&0&00D

Disk drive No DISKDRIVE 5.2.3790.1830 10/1/2002
(Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&1D1C1BB3&0&00B

Disk drive No DISKDRIVE 5.2.3790.1830 10/1/2002
(Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&1D1C1BB3&0&009

Disk drive No DISKDRIVE 5.2.3790.1830 10/1/2002
(Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&1D1C1BB3&0&007

Disk drive No DISKDRIVE 5.2.3790.1830 10/1/2002
(Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&1D1C1BB3&0&005

Disk drive No DISKDRIVE 5.2.3790.1830 10/1/2002
(Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&1D1C1BB3&0&003

Disk drive No DISKDRIVE 5.2.3790.1830 10/1/2002
(Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&1D1C1BB3&0&001

QLogic Fibre Channel Adapter No SCSIADAPTER 9.0.1.64
2/18/2005 QLogic oem0.inf Not Available
PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&474B838&0&08

PCI standard host CPU bridge No SYSTEM 5.2.3790.1830
10/1/2002 (Standard system devices) machine.inf Not Available
PCI\VEN_1014&DEV_02A1&SUBSYS_00000000&REV_02\3&474B838&0&00

PCI bus No SYSTEM 5.2.3790.1830 10/1/2002 (Standard system devices) machine.inf Not Available
ACPI\PNP0A03\5

Qlogic processor device No SYSTEM 5.2.3790.1830
10/1/2002 QLOGIC scsidev.inf Not Available
SCSI\PROCESSOR&VEN_QLOGIC&PROD_PSEUDO_DEVICE&REV_4&17455A85&0&07F0

Disk drive No DISKDRIVE 5.2.3790.1830 10/1/2002
(Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&17455A85&0&00C

Disk drive No DISKDRIVE 5.2.3790.1830 10/1/2002
(Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&17455A85&0&00A

Disk drive No DISKDRIVE 5.2.3790.1830 10/1/2002
(Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&17455A85&0&008

Disk drive No DISKDRIVE 5.2.3790.1830 10/1/2002
(Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&17455A85&0&006

Disk drive No DISKDRIVE 5.2.3790.1830 10/1/2002
(Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&17455A85&0&004

Disk drive No DISKDRIVE 5.2.3790.1830 10/1/2002
(Standard disk drives) disk.inf Not Available
SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&17455A85&0&002

Disk drive No DISKDRIVE 5.2.3790.1830 10/1/2002
(Standard disk drives) disk.inf Not Available

SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&17455A85&0&00	Qlogic processor device	No	SYSTEM	5.2.3790.1830	10/1/2002	QLOGIC	scsidev.inf	Not Available
0	SCSI\PROCESSOR&VEN_QLOGIC&PROD_PSEUDO_DEVICE&REV_4&3B4E3515&0&07F0							
QLogic Fibre Channel Adapter	No	SCSIADAPTER	9.0.1.64					
2/18/2005 QLogic	oem0.inf	Not Available						
PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&172E68DD&0&08	Disk drive	No	DISKDRIVE	5.2.3790.1830	10/1/2002	(Standard disk drives)	disk.inf	Not Available
	SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&3B4E3515&0&00C							
PCI standard host CPU bridge	No	SYSTEM	5.2.3790.1830					
10/1/2002 (Standard system devices)	machine.inf	Not Available						
Available								
PCI\VEN_1014&DEV_02A1&SUBSYS_00000000&REV_02\3&172E68DD&0&00	Disk drive	No	DISKDRIVE	5.2.3790.1830	10/1/2002	(Standard disk drives)	disk.inf	Not Available
	SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&3B4E3515&0&00A							
PCI bus	No	SYSTEM	5.2.3790.1830					
10/1/2002 (Standard system devices)	machine.inf	Not Available						
ACPI\PNP0A03\4								
Qlogic processor device	No	SYSTEM	5.2.3790.1830					
10/1/2002 QLOGIC	scsidev.inf	Not Available						
SCSI\PROCESSOR&VEN_QLOGIC&PROD_PSEUDO_DEVICE&REV_4&232FAFE&0&07F0								
Disk drive	No	DISKDRIVE	5.2.3790.1830					
(Standard disk drives)	disk.inf	Not Available						
SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&232FAFE&0&00D								
Disk drive	No	DISKDRIVE	5.2.3790.1830					
(Standard disk drives)	disk.inf	Not Available						
SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&232FAFE&0&00B								
Disk drive	No	DISKDRIVE	5.2.3790.1830					
(Standard disk drives)	disk.inf	Not Available						
SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&232FAFE&0&009								
Disk drive	No	DISKDRIVE	5.2.3790.1830					
(Standard disk drives)	disk.inf	Not Available						
SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&232FAFE&0&007								
Disk drive	No	DISKDRIVE	5.2.3790.1830					
(Standard disk drives)	disk.inf	Not Available						
SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&232FAFE&0&005								
Disk drive	No	DISKDRIVE	5.2.3790.1830					
(Standard disk drives)	disk.inf	Not Available						
SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&232FAFE&0&003								
Disk drive	No	DISKDRIVE	5.2.3790.1830					
(Standard disk drives)	disk.inf	Not Available						
SCSI\DISK&VEN_IBM&PROD_1742-900&REV_0520\4&232FAFE&0&001								
QLogic Fibre Channel Adapter	No	SCSIADAPTER	9.0.1.64					
2/18/2005 QLogic	oem0.inf	Not Available						
PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&29E81982&0&08								
PCI standard host CPU bridge	No	SYSTEM	5.2.3790.1830					
10/1/2002 (Standard system devices)	machine.inf	Not Available						
Available								
PCI\VEN_1014&DEV_02A1&SUBSYS_00000000&REV_02\3&29E81982&0&00								
PCI bus	No	SYSTEM	5.2.3790.1830					
10/1/2002 (Standard system devices)	machine.inf	Not Available						
ACPI\PNP0A03\3								
	Broadcom NetXtreme Gigabit Ethernet	No	NET	7.98.0.0	10/1/2002	Broadcom	net5amd.inf	Not Available
	PCI\VEN_14E4&DEV_1648&SUBSYS_02E71014&REV_10\3&13C0B0C5&0&09							
	Broadcom NetXtreme Gigabit Ethernet	No	NET	7.98.0.0	10/1/2002	Broadcom	net5amd.inf	Not Available
	PCI\VEN_14E4&DEV_1648&SUBSYS_02E71014&REV_10\3&13C0B0C5&0&08							
	PCI standard host CPU bridge	No	SYSTEM	5.2.3790.1830	10/1/2002	(Standard system devices)	machine.inf	Not Available
	Available							

PCI\VEN_1014&DEV_02A1&SUBSYS_00000000&REV_02\3&13C0B0C5&0&00					System CMOS/real time clock	No	SYSTEM	5.2.3790.1830	10/1/2002	(Standard system devices)	machine.inf	Not Available
PCI bus	No	SYSTEM	5.2.3790.1830	10/1/2002	System timer	No	SYSTEM	5.2.3790.1830	10/1/2002	(Standard system devices)	machine.inf	Not Available
(Standard system devices)					Available	ACPI\PNP0B00\4&13245C1&0						
ACPI\PNP0A03\1					Advanced programmable interrupt controller	No	SYSTEM	5.2.3790.1830	10/1/2002	(Standard system devices)	machine.inf	Not Available
Memory Module	No	MEMORY	5.2.3790.1830	10/1/2002	Not Available	ACPI\PNP0100\4&13245C1&0						
Microsoft memory.inf					Direct memory access controller	No	SYSTEM	5.2.3790.1830	10/1/2002	(Standard system devices)	machine.inf	Not Available
Intel Processor	No	PROCESSOR	5.2.3790.1830	10/1/2002	Available	ACPI\PNP0200\4&13245C1&0						
10/1/2002 Intel					Communications Port	No	PORTS	5.2.3790.1830	10/1/2002	(Standard port types)	msports.inf	Not Available
ACPI\GENUINEINTEL_-_EM64T_FAMILY_15_MODEL_4_7					Available	ACPI\PNP0501\2						
Intel Processor	No	PROCESSOR	5.2.3790.1830	10/1/2002	Advanced programmable interrupt controller	No	SYSTEM	5.2.3790.1830	10/1/2002	(Standard system devices)	machine.inf	Not Available
10/1/2002 Intel					Not Available	ACPI\PNP0003\4&13245C1&0						
ACPI\GENUINEINTEL_-_EM64T_FAMILY_15_MODEL_4_6					Communications Port	No	PORTS	5.2.3790.1830	10/1/2002	(Standard port types)	msports.inf	Not Available
Intel Processor	No	PROCESSOR	5.2.3790.1830	10/1/2002	Available	ACPI\PNP0501\1						
10/1/2002 Intel					PS/2 Compatible Mouse	No	MOUSE	5.2.3790.1830	10/1/2002	Microsoft	msmouse.inf	Not Available
ACPI\GENUINEINTEL_-_EM64T_FAMILY_15_MODEL_4_5					Available	ACPI\PNP0F13\4&13245C1&0						
Intel Processor	No	PROCESSOR	5.2.3790.1830	10/1/2002	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard	No	KEYBOARD	5.2.3790.1830	10/1/2002	(Standard keyboards)	keyboard.inf	Not Available
10/1/2002 Intel					Available	ACPI\PNP0303\4&13245C1&0						
ACPI\GENUINEINTEL_-_EM64T_FAMILY_15_MODEL_4_4					Serverworks Champion CSB6 - SouthBridge 6 LPC	No	SYSTEM	5.2.3790.1830	10/1/2002	ServerWorks (RCC)	machine.inf	Not Available
Intel Processor	No	PROCESSOR	5.2.3790.1830	10/1/2002	Not Available	PCI\VEN_1166&DEV_0227&SUBSYS_00000000&REV_00\3&267A616A&0&7B						
10/1/2002 Intel					Secondary IDE Channel	No	HDC	5.2.3790.1830	10/1/2002	(Standard IDE ATA/ATAPI controllers)	mshdc.inf	Not Available
ACPI\GENUINEINTEL_-_EM64T_FAMILY_15_MODEL_4_3					Available	PCI\IDE\IDECHANNEL\4&101988B2&0&1						
Intel Processor	No	PROCESSOR	5.2.3790.1830	10/1/2002	CD-ROM Drive	No	CDROM	5.2.3790.1830	10/1/2002	(Standard CD-ROM drives)	cdrom.inf	Not Available
10/1/2002 Intel					Available	IDE\CDROMHL-DT-ST_DVD-ROM_GDR8082N_____0L03_____\5&A8D2D22&0&0.0.0						
ACPI\GENUINEINTEL_-_EM64T_FAMILY_15_MODEL_4_2					Primary IDE Channel	No	HDC	5.2.3790.1830	10/1/2002	(Standard IDE ATA/ATAPI controllers)	mshdc.inf	Not Available
Intel Processor	No	PROCESSOR	5.2.3790.1830	10/1/2002	Available	PCI\IDE\IDECHANNEL\4&101988B2&0&0						
10/1/2002 Intel					Standard Dual Channel PCI IDE Controller	No	HDC	5.2.3790.1830	10/1/2002	(Standard IDE ATA/ATAPI controllers)	mshdc.inf	Not Available
ACPI\GENUINEINTEL_-_EM64T_FAMILY_15_MODEL_4_0					Available	PCI\VEN_1166&DEV_0213&SUBSYS_02121166&REV_A0\3&267A616A&0&79						
ACPI Fixed Feature Button	No	SYSTEM	5.2.3790.1830	10/1/2002	ServerWorks Champion CSB6 - SouthBridge 6	No	SYSTEM	5.2.3790.1830	10/1/2002	ServerWorks (RCC)	machine.inf	Not Available
10/1/2002 (Standard system devices)					Not Available	PCI\VEN_1166&DEV_0203&SUBSYS_00000000&REV_A0\3&267A616A&0&78						
Available	ACPI\FIXEDBUTTON\2&DABA3FF&0				USB Root Hub	No	USB	5.2.3790.1830	10/1/2002	(Standard USB Host Controller)	usbport.inf	Not Available
System board	No	SYSTEM	5.2.3790.1830	10/1/2002	Available	USB\ROOT_HUB20\4&2B778F81&0						
(Standard system devices)					Not Available							
ACPI\PNP0C01\1					Motherboard resources	No	SYSTEM	5.2.3790.1830	10/1/2002	(Standard system devices)	machine.inf	Not Available
Not Available	Not Available	Not Available	Not Available	Not Available	Available	ACPI\PNP0C02\3						
Not Available	Not Available	Not Available	Not Available	Not Available	Numeric data processor	No	SYSTEM	5.2.3790.1830	10/1/2002	(Standard system devices)	machine.inf	Not Available
Not Available	ACPI\IBM37D4\2&DABA3FF&0				Available	ACPI\PNP0C04\4&13245C1&0						
System speaker	No	SYSTEM	5.2.3790.1830	10/1/2002	System speaker	No	SYSTEM	5.2.3790.1830	10/1/2002	(Standard system devices)	machine.inf	Not Available
(Standard system devices)					Available	ACPI\PNP0800\4&13245C1&0						
ACPI\PNP0800\4&13245C1&0												

Standard Enhanced PCI to USB Host Controller No USB
5.2.3790.1830 10/1/2002 (Standard USB Host Controller) usbport.inf
Not Available
PCI\VEN_1033&DEV_00E0&SUBSYS_00E01033&REV_04\3&267A616A&
0&1A

HID SYS Not Available Not Available Not Available
Not Available Not Available Not Available Not
Available USB\VID_04B3&PID_4001&MI_02\6&18E8217A&0&0002

HID-compliant mouse No MOUSE 5.2.3790.1830 10/1/2002
Microsoft msmouse.inf Not Available
HID\VID_04B3&PID_4001&MI_01\7&2FE08322&0&0000

USB Human Interface Device No HIDCLASS
5.2.3790.1830 10/1/2002 (Standard system devices) input.inf
Not Available
USB\VID_04B3&PID_4001&MI_01\6&18E8217A&0&0001

HID Keyboard Device No KEYBOARD
5.2.3790.1830 10/1/2002 (Standard keyboards) keyboard.inf
Not Available
HID\VID_04B3&PID_4001&MI_00\7&181CF6DD&0&0000

USB Human Interface Device No HIDCLASS
5.2.3790.1830 10/1/2002 (Standard system devices) input.inf
Not Available
USB\VID_04B3&PID_4001&MI_00\6&18E8217A&0&0000

USB Composite Device No USB 5.2.3790.1830
10/1/2002 (Standard USB Host Controller) usb.inf Not Available
USB\VID_04B3&PID_4001\000D6046D4BA

USB Root Hub No USB 5.2.3790.1830 10/1/2002
(Standard USB Host Controller) usbport.inf Not Available
USB\ROOT_HUB\4&2DDBD7B&0

NEC PCI to USB Open Host Controller No USB
5.2.3790.1830 10/1/2002 NEC usbport.inf Not Available
PCI\VEN_1033&DEV_0035&SUBSYS_00351033&REV_43\3&267A616A&
0&19

USB Root Hub No USB 5.2.3790.1830 10/1/2002
(Standard USB Host Controller) usbport.inf Not Available
USB\ROOT_HUB\4&15976E20&0

NEC PCI to USB Open Host Controller No USB
5.2.3790.1830 10/1/2002 NEC usbport.inf Not Available
PCI\VEN_1033&DEV_0035&SUBSYS_00351033&REV_43\3&267A616A&
0&18

Default Monitor No MONITOR 5.2.3790.1830
10/1/2002 (Standard monitor types) monitor.inf Not
Available
DISPLAY\DEFAULT_MONITOR\4&36FA8DD8&0&10000000&00&01

Plug and Play Monitor No MONITOR
5.2.3790.1830 10/1/2002 (Standard monitor types) monitor.inf
Not Available
DISPLAY\IBM029A\4&36FA8DD8&0&10000082&00&01

Radeon 7000 / RADEON VE Family (Microsoft Corporation) No
DISPLAY 6.14.10.6508 12/3/2004 ATI Technologies Inc.
atiixpag.inf Not Available
PCI\VEN_1002&DEV_5159&SUBSYS_02C81014&REV_00\3&267A616A&
0&08

PCI standard host CPU bridge No SYSTEM 5.2.3790.1830
10/1/2002 (Standard system devices) machine.inf Not
Available
PCI\VEN_1014&DEV_02A1&SUBSYS_00000000&REV_02\3&267A616A&
0&00

PCI bus No SYSTEM 5.2.3790.1830 10/1/2002 (Standard
system devices) machine.inf Not Available
ACPI\PNP0A03\0

Microsoft ACPI-Compliant System No SYSTEM
5.2.3790.1830 10/1/2002 Microsoft acpi.inf Not Available
ACPI\HAL\PNP0C08\0

ACPI Multiprocessor x64-based PC No COMPUTER
5.2.3790.1830 10/1/2002 (Standard computers) hal.inf Not
Available ROOT\ACPI_HAL\0000

Not Available Not Available Not Available Not
Available Not Available Not Available Not Available
Not Available HTREE\ROOT\0

[Environment Variables]

Variable Value User Name

CLASSPATH
.;C:\SQLLIB\java\db2java.zip;C:\SQLLIB\java\db2jcc.jar;C:\SQLLIB\java\sqlj.
zip;C:\SQLLIB\java\db2jcc_license_cu.jar;C:\SQLLIB\bin;C:\SQLLIB\java\co
mmon.jar <SYSTEM>

ClusterLog C:\WINDOWS\Cluster\cluster.log <SYSTEM>

ComSpec %SystemRoot%\system32\cmd.exe <SYSTEM>

DB2INSTANCE DB2 <SYSTEM>

DB2TEMPDIR C:\SQLLIB\ <SYSTEM>

FP_NO_HOST_CHECK NO <SYSTEM>

INCLUDE
C:\MsSDKx64\Include;C:\MsSDKx64\Include\crt;C:\MsSDKx64\Include\crt\s
ys;C:\MsSDKx64\Include\mf;C:\MsSDKx64\Include\atl;C:\SQLLIB\INCLUD
E;C:\SQLLIB\LIB <SYSTEM>

LIB
C:\MsSDKx64\Lib\AMD64;C:\MsSDKx64\Lib\AMD64\mf;C:\SQLLIB\LIB
<SYSTEM>

MSSDK C:\MsSDKx64 <SYSTEM>

NUMBER_OF_PROCESSORS 16 <SYSTEM>

OS Windows_NT <SYSTEM>

Path
C:\MsSDKx64\Bin\Win64\x86\AMD64;C:\MsSDKx64\Bin;C:\MsSDKx64\Bin
\Winnt;C:\tpc-c.ibm\utils\Windows;C:\tools;c:\tools\util;C:\Perl\bin;%SystemR
oot%\system32;%SystemRoot%;%SystemRoot%\System32\Wbem;C:\SQLLIB\
BIN;C:\SQLLIB\FUNCTION;C:\SQLLIB\SAMPLES\REPL;C:\vim6\
<SYSTEM>

PATHEXT
 .COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH <SYSTEM>

PROCESSOR_ARCHITECTURE AMD64 <SYSTEM>

PROCESSOR_IDENTIFIER EM64T Family 15 Model 4 Stepping 1, GenuineIntel <SYSTEM>

PROCESSOR_LEVEL 15 <SYSTEM>

PROCESSOR_REVISION 0401 <SYSTEM>

TEMP %SystemRoot%\TEMP <SYSTEM>

TMP %SystemRoot%\TEMP <SYSTEM>

windir %SystemRoot% <SYSTEM>

TEMP %USERPROFILE%\Local Settings\Temp NT AUTHORITY\SYSTEM

TMP %USERPROFILE%\Local Settings\Temp NT AUTHORITY\SYSTEM

TEMP %USERPROFILE%\Local Settings\Temp NT AUTHORITY\LOCAL SERVICE

TMP %USERPROFILE%\Local Settings\Temp NT AUTHORITY\LOCAL SERVICE

TEMP %USERPROFILE%\Local Settings\Temp NT AUTHORITY\NETWORK SERVICE

TMP %USERPROFILE%\Local Settings\Temp NT AUTHORITY\NETWORK SERVICE

TEMP %USERPROFILE%\Local Settings\Temp DB2SERV2\Administrator

TMP %USERPROFILE%\Local Settings\Temp DB2SERV2\Administrator

VCToolkitInstallDir C:\MSVC2003\ DB2SERV2\Administrator

[Print Jobs]

Document Size	Owner	Notify	Status	Time Submitted
Start Time	Until Time	Elapsed Time	Pages Printed	Job ID
Priority	Parameters	Driver	Print Processor	Host
Data Type	Name		Queue	

[Network Connections]

Local Name	Remote Name	Type	Status	User Name
Z:	\\fsserv\ddrive	Disk	Current Connection	
	DB2SERV2\Administrator			

[Running Tasks]

Name	Path	Process ID	Priority	Min Working Set	Max Working Set
Working Set	Start Time	Version	Size	File Date	
system idle process	Not Available	0	0	0	Not Available
system	Not Available	4	8	0	1413120
smss.exe	Not Available	896	11	204800	1413120
csrss.exe	Not Available	928	13	Not Available	Not Available
winlogon.exe	c:\windows\system32\winlogon.exe	13	204800	1413120	5/20/2005 9:16 AM
services.exe	c:\windows\system32\services.exe	204800	1413120	5/20/2005 9:16 AM	5.2.3790.1830
lsass.exe	c:\windows\system32\lsass.exe	1413120	5/20/2005 9:16 AM	5.2.3790.1830	3/25/2005 7:00 AM
svchost.exe	c:\windows\system32\svchost.exe	204800	1413120	5/20/2005 9:16 AM	5.2.3790.1830
svchost.exe	Not Available	1000	8	Not Available	Not Available
svchost.exe	Not Available	216	8	Not Available	Not Available
svchost.exe	c:\windows\system32\svchost.exe	204800	1413120	5/20/2005 9:16 AM	5.2.3790.1830
msdtc.exe	Not Available	1844	8	Not Available	Not Available
wrshdnt.exe	c:\wrshdnt\wrshdnt.exe	204800	1413120	5/20/2005 9:16 AM	2.23.00 92.00 KB (94,208 bytes)
svchost.exe	Not Available	1960	8	Not Available	Not Available

explorer.exe	c:\windows\explorer.exe	748	8						wmiprvse.exe	Not Available	640	8	Not Available						
204800	1413120	5/20/2005 9:16 AM	6.00.3790.1830						Available	Not Available	5/20/2005 10:45 AM	Not Available							
(srv03_sp1_rtm.050324-1447)	1.30 MB (1,364,480 bytes)	3/25/2005 7:00 AM							Not Available	Not Available									
svchost.exe	c:\windows\system32\svchost.exe	1272	8						helpsvc.exe	c:\windows\pchealth\helpctr\binaries\helpsvc.exe	1956	8	204800	1413120	5/20/2005 10:45 AM				
204800	1413120	5/20/2005 9:16 AM	5.2.3790.1830						5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	1.52 MB (1,591,296 bytes)	4/1/2005 10:17 AM								
(srv03_sp1_rtm.050324-1447)	24.50 KB (25,088 bytes)	3/25/2005 7:00 AM																	
wmiprvse.exe	Not Available	1776	8	Not Available					[Loaded Modules]										
Available	Not Available	5/20/2005 9:17 AM	Not Available						Name	Version	Size	File Date	Manufacturer	Path					
Not Available	Not Available								winlogon	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	901.00 KB (922,624 bytes)	3/25/2005 7:00 AM	Microsoft Corporation	c:\windows\system32\winlogon.exe					
wpabaln.exe	c:\windows\system32\wpabaln.exe	1812	8						ntdll	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	1.20 MB (1,257,472 bytes)	3/25/2005 7:00 AM	Microsoft Corporation	c:\windows\system32\ntdll.dll					
204800	1413120	5/20/2005 9:18 AM	5.2.3790.1830						kernel32	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	1.43 MB (1,500,160 bytes)	3/25/2005 7:00 AM	Microsoft Corporation	c:\windows\system32\kernel32.dll					
(srv03_sp1_rtm.050324-1447)	33.50 KB (34,304 bytes)	3/25/2005 7:00 AM							advapi32	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	1.00 MB (1,051,136 bytes)	3/25/2005 7:00 AM	Microsoft Corporation	c:\windows\system32\advapi32.dll					
cmd.exe	c:\windows\system32\cmd.exe	456	8	204800					rpert4	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	1.63 MB (1,714,176 bytes)	3/25/2005 7:00 AM	Microsoft Corporation	c:\windows\system32\rpert4.dll					
1413120	5/20/2005 9:20 AM	5.2.3790.1830	(srv03_sp1_rtm.050324-1447)	538.50 KB (551,424 bytes)	3/25/2005 7:00 AM				crypt32	5.131.3790.1830 (srv03_sp1_rtm.050324-1447)	1.36 MB (1,428,992 bytes)	3/25/2005 7:00 AM	Microsoft Corporation	c:\windows\system32\crypt32.dll					
taskmgr.exe	c:\windows\system32\taskmgr.exe	1108							msasn1	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	152.50 KB (156,160 bytes)	3/25/2005 7:00 AM	Microsoft Corporation	c:\windows\system32\msasn1.dll					
13	204800	1413120	5/20/2005 9:34 AM	5.2.3790.1830					msvcrt	7.0.3790.1830 (srv03_sp1_rtm.050324-1447)	508.00 KB (520,192 bytes)	3/25/2005 7:00 AM	Microsoft Corporation	c:\windows\system32\msvcrt.dll					
(srv03_sp1_rtm.050324-1447)	231.00 KB (236,544 bytes)	3/25/2005 7:00 AM							user32	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	1.04 MB (1,085,952 bytes)	3/25/2005 7:00 AM	Microsoft Corporation	c:\windows\system32\user32.dll					
db2bp.exe	c:\sql\bin\db2bp.exe	992	8	204800					gdi32	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	592.00 KB (606,208 bytes)	3/25/2005 7:00 AM	Microsoft Corporation	c:\windows\system32\gdi32.dll					
1413120	5/20/2005 10:18 AM	8.1.9.671	1.15 MB (1,208,320 bytes)	3/24/2005 11:21 PM					nddeapi	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	25.00 KB (25,600 bytes)	3/25/2005 7:00 AM	Microsoft Corporation	c:\windows\system32\nddeapi.dll					
notepad.exe	c:\windows\system32\notepad.exe	628	8						profmap	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	36.00 KB (36,864 bytes)	3/25/2005 7:00 AM	Microsoft Corporation	c:\windows\system32\profmap.dll					
204800	1413120	5/20/2005 10:19 AM	5.2.3790.1830						netapi32	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	589.00 KB (603,136 bytes)	3/25/2005 7:00 AM	Microsoft Corporation	c:\windows\system32\netapi32.dll					
(srv03_sp1_rtm.050324-1447)	86.00 KB (88,064 bytes)	3/25/2005 7:00 AM																	
javaw.exe	c:\program files (x86)\raidman\jre\bin\javaw.exe	888	8	24.09 KB															
204800	1413120	5/20/2005 10:23 AM	Not Available	(24,671 bytes)	8/29/2002 9:10 AM														
mmc.exe	c:\windows\system32\mmc.exe	576	8	204800															
1413120	5/20/2005 10:28 AM	5.2.3790.1830	(srv03_sp1_rtm.050324-1447)	1.83 MB (1,920,512 bytes)	3/25/2005 7:00 AM														
vds.exe	c:\windows\system32\vds.exe	1668	8	204800															
1413120	5/20/2005 10:28 AM	5.2.3790.1830	(srv03_sp1_rtm.050324-1447)	418.00 KB (428,032 bytes)	3/25/2005 7:00 AM														
dmadmin.exe	c:\windows\system32\dmadmin.exe	1512	8																
204800	1413120	5/20/2005 10:28 AM	5.2.3790.1830																
(srv03_sp1_rtm.050324-1447)	389.50 KB (398,848 bytes)	3/25/2005 7:00 AM																	
wordpad.exe	c:\program files\windows nt\accessories\wordpad.exe	1320	8	204800															
5.2.3790.1830	(srv03_sp1_rtm.050324-1447)	333.50 KB (341,504 bytes)	4/1/2005 10:15 AM																
mmc.exe	c:\windows\system32\mmc.exe	1660	8	204800															
1413120	5/20/2005 10:44 AM	5.2.3790.1830	(srv03_sp1_rtm.050324-1447)	1.83 MB (1,920,512 bytes)	3/25/2005 7:00 AM														
helpctr.exe	c:\windows\pchealth\helpctr\binaries\helpctr.exe	1764	8																
204800	1413120	5/20/2005 10:45 AM	5.2.3790.1830																
(srv03_sp1_rtm.050324-1447)	1.30 MB (1,363,456 bytes)	4/1/2005 10:17 AM																	

userenv 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 1.02 MB (1,069,056 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\userenv.dll	comctl32 6.0 (srv03_sp1_rtm.050324-1447) 1.51 MB (1,584,128 bytes) 4/1/2005 5:03 AM Microsoft Corporation c:\windows\winsxs\amd64_microsoft.windows.common-controls_6595b64144c cf1df_6.0.3790.1830_x-ww_aced72af\comctl32.dll
psapi 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 29.00 KB (29,696 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\psapi.dll	winscard 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 230.00 KB (235,520 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\winscard.dll
regapi 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 108.50 KB (111,104 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\regapi.dll	wtsapi32 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 29.00 KB (29,696 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\wtsapi32.dll
secur32 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 120.00 KB (122,880 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\secur32.dll	winmm 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 303.50 KB (310,784 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\winmm.dll
setupapi 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 1.45 MB (1,523,200 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\setupapi.dll	shell32 6.0.3790.1830 (srv03_sp1_rtm.050324-1447) 10.01 MB (10,492,416 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\shell32.dll
version 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 28.00 KB (28,672 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\version.dll	sxs 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 1.91 MB (2,003,968 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\sxs.dll
winsta 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 89.00 KB (91,136 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\winsta.dll	rsaenh 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 241.96 KB (247,768 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\rsaenh.dll
ws2_32 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 176.50 KB (180,736 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\ws2_32.dll	wldap32 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 390.00 KB (399,360 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\wldap32.dll
ws2help 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 30.50 KB (31,232 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\ws2help.dll	csdll 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 151.50 KB (155,136 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\csdll.dll
msgina 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 1.14 MB (1,193,472 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\msgina.dll	dimsntfy 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 28.00 KB (28,672 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\dimsntfy.dll
shsvcs 6.0.3790.1830 (srv03_sp1_rtm.050324-1447) 193.50 KB (198,144 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\shsvcs.dll	wlnotify 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 148.00 KB (151,552 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\wlnotify.dll
shlwapi 6.0.3790.1830 (srv03_sp1_rtm.050324-1447) 606.50 KB (621,056 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\shlwapi.dll	mpr 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 115.00 KB (117,760 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\mpr.dll
sfc 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 6.00 KB (6,144 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\sfc.dll	oleaut32 5.2.3790.1830 1.06 MB (1,116,160 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\oleaut32.dll
sfc_os 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 183.50 KB (187,904 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\sfc_os.dll	winspool 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 247.00 KB (252,928 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\winspool.drv
wintrust 5.131.3790.1830 (srv03_sp1_rtm.050324-1447) 297.50 KB (304,640 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\wintrust.dll	comctl32 5.82 (srv03_sp1_rtm.050324-1447) 934.50 KB (956,928 bytes) 4/1/2005 5:03 AM Microsoft Corporation c:\windows\winsxs\amd64_microsoft.windows.common-controls_6595b64144c cf1df_5.82.3790.1830_x-ww_4d792d2a\comctl32.dll
imagehlp 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 57.50 KB (58,880 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\imagehlp.dll	uxtheme 6.0.3790.1830 (srv03_sp1_rtm.050324-1447) 494.50 KB (506,368 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\uxtheme.dll
ole32 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 2.43 MB (2,543,616 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\ole32.dll	

samlib 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 69.00 KB (70,656 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\samlib.dll	umpnpmgr 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 205.00 KB (209,920 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\umpnpmgr.dll
cscui 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 441.00 KB (451,584 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\cscui.dll	eventlog 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 127.00 KB (130,048 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\eventlog.dll
mprapi 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 154.50 KB (158,208 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\mprapi.dll	lsass 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 14.00 KB (14,336 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\lsass.exe
activeds 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 348.50 KB (356,864 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\activeds.dll	lsasrv 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 1.50 MB (1,568,256 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\lsasrv.dll
adslrpc 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 240.50 KB (246,272 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\adslrpc.dll	ntdsapi 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 127.50 KB (130,560 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\ntdsapi.dll
credui 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 202.00 KB (206,848 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\credui.dll	dnsapi 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 297.50 KB (304,640 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\dnsapi.dll
atl 3.05.2284 96.50 KB (98,816 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\atl.dll	samsrv 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 1.01 MB (1,059,328 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\samsrv.dll
rtutils 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 66.00 KB (67,584 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\rtutils.dll	cryptdll 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 47.00 KB (48,128 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\cryptdll.dll
ntmarta 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 222.50 KB (227,840 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\ntmarta.dll	msprivs 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 47.50 KB (48,640 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\msprivs.dll
clbcatq 2001.12.4720.1830 (srv03_sp1_rtm.050324-1447) 865.00 KB (885,760 bytes) 4/1/2005 10:15 AM Microsoft Corporation c:\windows\system32\clbcatq.dll	kerberos 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 698.00 KB (714,752 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\kerberos.dll
comres 2001.12.4720.1830 (srv03_sp1_rtm.050324-1447) 779.50 KB (798,208 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\comres.dll	msv1_0 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 253.00 KB (259,072 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\msv1_0.dll
xpsp2res 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 2.77 MB (2,899,456 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\xpsp2res.dll	iphlpapi 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 177.00 KB (181,248 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\iphlpapi.dll
services 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 216.50 KB (221,696 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\services.exe	netlogon 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 666.00 KB (681,984 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\netlogon.dll
ncobjapi 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 80.00 KB (81,920 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\ncobjapi.dll	w32time 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 400.50 KB (410,112 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\w32time.dll
msvcp60 7.0.3790.1830 (srv03_sp1_rtm.050324-1447) 919.50 KB (941,568 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\msvcp60.dll	schannel 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 248.00 KB (253,952 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\schannel.dll
scesrv 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 594.50 KB (608,768 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\scesrv.dll	wdigest 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 130.50 KB (133,632 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\wdigest.dll
authz 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 167.00 KB (171,008 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\authz.dll	rassfm 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 36.00 KB (36,864 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\rassfm.dll

kdcsvc 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 409.00 KB (418,816 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\kdcsvc.dll	aelupsvc 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 31.50 KB (32,256 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\aelupsvc.dll
ntdsa 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 2.81 MB (2,948,096 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\ntdsa.dll	apphelp 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 241.00 KB (246,784 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\apphelp.dll
esent 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 2.26 MB (2,366,976 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\esent.dll	dmserver 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 36.50 KB (37,376 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\dmserver.dll
ntdsatq 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 51.00 KB (52,224 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\ntdsatq.dll	es 2001.12.4720.1830 (srv03_sp1_rtm.050324-1447) 357.00 KB (365,568 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\es.dll
mwssock 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 478.00 KB (489,472 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\mwssock.dll	srvsvc 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 156.50 KB (160,256 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\srvsvc.dll
scecli 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 308.00 KB (315,392 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\scecli.dll	seclogon 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 27.50 KB (28,160 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\seclogon.dll
ws03res 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 794.00 KB (813,056 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\ws03res.dll	sens 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 63.50 KB (65,024 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\sens.dll
hnetcfg 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 561.00 KB (574,464 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\hnetcfg.dll	wmisvc 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 227.00 KB (232,448 bytes) 4/1/2005 10:14 AM Microsoft Corporation c:\windows\system32\wbem\wmisvc.dll
wshtcpip 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 29.00 KB (29,696 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\wshtcpip.dll	vssapi 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 1.26 MB (1,320,960 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\vssapi.dll
pstorsvc 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 36.00 KB (36,864 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\pstorsvc.dll	comsvcs 2001.12.4720.1830 (srv03_sp1_rtm.050324-1447) 2.06 MB (2,156,544 bytes) 4/1/2005 10:15 AM Microsoft Corporation c:\windows\system32\comsvcs.dll
psbase 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 124.00 KB (126,976 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\psbase.dll	wiarpc 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 57.00 KB (58,368 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\wiarpc.dll
dssenh 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 226.96 KB (232,408 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\dssenh.dll	netman 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 457.00 KB (467,968 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\netman.dll
svchost 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 24.50 KB (25,088 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\svchost.exe	netshell 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 2.32 MB (2,437,120 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\netshell.dll
rpss 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 672.00 KB (688,128 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\rpss.dll	clusapi 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 127.00 KB (130,048 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\clusapi.dll
schedsvc 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 308.50 KB (315,904 bytes) 4/1/2005 10:17 AM Microsoft Corporation c:\windows\system32\schedsvc.dll	rasapi32 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 410.00 KB (419,840 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\rasapi32.dll
msidle 6.00.3790.1830 (srv03_sp1_rtm.050324-1447) 9.00 KB (9,216 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\msidle.dll	rasman 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 95.50 KB (97,792 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\rasman.dll
wkssvc 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 221.00 KB (226,304 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\wkssvc.dll	tapi32 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 332.50 KB (340,480 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\tapi32.dll

wininet 6.00.3790.1830 (srv03_sp1_rtm.050324-1447) 1.13 MB (1,186,304 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\wininet.dll	wbemcons 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 65.50 KB (67,072 bytes) 4/1/2005 10:14 AM Microsoft Corporation c:\windows\system32\wbem\wbemcons.dll
wzcsapi 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 49.00 KB (50,176 bytes) 3/24/2005 12:35 PM Microsoft Corporation c:\windows\system32\wzcsapi.dll	pchsvc 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 76.00 KB (77,824 bytes) 4/1/2005 10:17 AM Microsoft Corporation c:\windows\pchealth\helpctr\binaries\pchsvc.dll
wzcsvc 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 492.00 KB (503,808 bytes) 3/24/2005 12:35 PM Microsoft Corporation c:\windows\system32\wzcsvc.dll	wrshdnt 2.23.00 92.00 KB (94,208 bytes) 4/4/2005 10:02 AM Denicomp Systems c:\wrshdnt\wrshdnt.exe
wmi 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 5.50 KB (5,632 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\wmi.dll	wow64 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 245.00 KB (250,880 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\wow64.dll
dhcpcsvc 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 219.00 KB (224,256 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\dhcpcsvc.dll	wow64win 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 280.00 KB (286,720 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\wow64win.dll
wbemcomn 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 524.00 KB (536,576 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\wbem\wbemcomn.dll	wow64cpu 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 18.50 KB (18,944 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\wow64cpu.dll
wbemcore 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 1.24 MB (1,299,968 bytes) 4/1/2005 10:14 AM Microsoft Corporation c:\windows\system32\wbem\wbemcore.dll	explorer 6.00.3790.1830 (srv03_sp1_rtm.050324-1447) 1.30 MB (1,364,480 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\explorer.exe
esscli 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 626.50 KB (641,536 bytes) 4/1/2005 10:14 AM Microsoft Corporation c:\windows\system32\wbem\esscli.dll	browseui 6.00.3790.1830 (srv03_sp1_rtm.050324-1447) 1.53 MB (1,601,536 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\browseui.dll
fastprox 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 866.50 KB (887,296 bytes) 4/1/2005 10:14 AM Microsoft Corporation c:\windows\system32\wbem\fastprox.dll	shdocvw 6.00.3790.1830 (srv03_sp1_rtm.050324-1447) 2.30 MB (2,416,128 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\shdocvw.dll
wbemsvc 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 58.00 KB (59,392 bytes) 4/1/2005 10:14 AM Microsoft Corporation c:\windows\system32\wbem\wbemsvc.dll	cryptui 5.131.3790.1830 (srv03_sp1_rtm.050324-1447) 705.50 KB (722,432 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\cryptui.dll
wmiutils 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 171.00 KB (175,104 bytes) 4/1/2005 10:14 AM Microsoft Corporation c:\windows\system32\wbem\wmiutils.dll	themeui 6.00.3790.1830 (srv03_sp1_rtm.050324-1447) 530.50 KB (543,232 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\themeui.dll
repdrvfs 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 353.50 KB (361,984 bytes) 4/1/2005 10:14 AM Microsoft Corporation c:\windows\system32\wbem\repdrvfs.dll	msimg32 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 6.50 KB (6,656 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\msimg32.dll
wmiprvsd 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 743.00 KB (760,832 bytes) 4/1/2005 10:14 AM Microsoft Corporation c:\windows\system32\wbem\wmiprvsd.dll	actxprxy 6.00.3790.1830 (srv03_sp1_rtm.050324-1447) 220.50 KB (225,792 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\actxprxy.dll
wbemess 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 532.50 KB (545,280 bytes) 4/1/2005 10:14 AM Microsoft Corporation c:\windows\system32\wbem\wbemess.dll	linkinfo 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 30.00 KB (30,720 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\linkinfo.dll
rasdlg 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 859.50 KB (880,128 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\rasdlg.dll	ntshrui 6.00.3790.1830 (srv03_sp1_rtm.050324-1447) 184.00 KB (188,416 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\ntshrui.dll
ncprov 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 73.00 KB (74,752 bytes) 4/1/2005 10:14 AM Microsoft Corporation c:\windows\system32\wbem\ncprov.dll	webcheck 6.00.3790.1830 (srv03_sp1_rtm.050324-1447) 439.00 KB (449,536 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\webcheck.dll
rasadhlp 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 12.00 KB (12,288 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\rasadhlp.dll	wsock32 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 24.50 KB (25,088 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\wsock32.dll

stobject 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 142.50 KB (145,920 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\stobject.dll	zipfldr 6.00.3790.1830 (srv03_sp1_rtm.050324-1447) 449.50 KB (460,288 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\zipfldr.dll
batmeter 6.00.3790.1830 (srv03_sp1_rtm.050324-1447) 41.50 KB (42,496 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\batmeter.dll	termsrv 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 354.50 KB (363,008 bytes) 4/1/2005 10:15 AM Microsoft Corporation c:\windows\system32\termsrv.dll
powrprof 6.00.3790.1830 (srv03_sp1_rtm.050324-1447) 32.50 KB (33,280 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\powrprof.dll	icaapi 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 27.50 KB (28,160 bytes) 4/1/2005 10:15 AM Microsoft Corporation c:\windows\system32\icaapi.dll
drprov 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 24.00 KB (24,576 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\drprov.dll	mstlsapi 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 187.00 KB (191,488 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\mstlsapi.dll
ntlanman 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 71.50 KB (73,216 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\ntlanman.dll	rdpwsx 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 170.13 KB (174,216 bytes) 4/1/2005 10:15 AM Microsoft Corporation c:\windows\system32\rdpwsx.dll
netui0 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 130.00 KB (133,120 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\netui0.dll	wpabaln 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 33.50 KB (34,304 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\wpabaln.exe
netui1 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 338.50 KB (346,624 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\netui1.dll	cmd 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 538.50 KB (551,424 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\cmd.exe
davclnt 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 38.00 KB (38,912 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\davclnt.dll	taskmgr 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 231.00 KB (236,544 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\taskmgr.exe
urlmon 6.00.3790.1830 (srv03_sp1_rtm.050324-1447) 1.02 MB (1,074,176 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\urlmon.dll	utildll 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 37.00 KB (37,888 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\utildll.dll
browsec 6.00.3790.1830 (srv03_sp1_rtm.050324-1447) 63.00 KB (64,512 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\browsec.dll	db2bp 8.1.9.671 1.15 MB (1,208,320 bytes) 3/24/2005 11:21 PM International Business Machines Corporation c:\sqlib\bin\db2bp.exe
shdoclc 6.00.3790.1830 (srv03_sp1_rtm.050324-1447) 589.50 KB (603,648 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\shdoclc.dll	db2wint64 8.1.9.671 36.00 KB (36,864 bytes) 3/24/2005 10:30 PM International Business Machines Corporation c:\sqlib\bin\db2wint64.dll
mprui 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 67.50 KB (69,120 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\mprui.dll	db2osse64 8.1.9.671 1.87 MB (1,961,984 bytes) 3/26/2005 4:04 PM International Business Machines Corporation c:\sqlib\bin\db2osse64.dll
netui2 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 542.00 KB (555,008 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\netui2.dll	db2sys64 8.1.9.671 1.45 MB (1,519,104 bytes) 3/26/2005 4:29 PM International Business Machines Corporation c:\sqlib\bin\db2sys64.dll
comdlg32 6.00.3790.1830 (srv03_sp1_rtm.050324-1447) 446.50 KB (457,216 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\comdlg32.dll	db2sysp64 8.1.9.671 160.00 KB (163,840 bytes) 3/24/2005 11:06 PM International Business Machines Corporation c:\sqlib\bin\db2sysp64.dll
netmsg 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 179.00 KB (183,296 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\netmsg.dll	db2app64 8.1.9.671 6.21 MB (6,508,032 bytes) 3/26/2005 4:11 PM International Business Machines Corporation c:\sqlib\bin\db2app64.dll
netplwiz 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 938.50 KB (961,024 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\netplwiz.dll	db2locale64 8.1.9.671 41.00 KB (41,984 bytes) 3/24/2005 10:29 PM International Business Machines Corporation c:\sqlib\bin\db2locale64.dll
mlang 6.00.3790.1830 (srv03_sp1_rtm.050324-1447) 686.00 KB (702,464 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\mlang.dll	db2g11n64 8.1.9.671 439.00 KB (449,536 bytes) 3/26/2005 4:29 PM International Business Machines Corporation c:\sqlib\bin\db2g11n64.dll

db2genreg64	8.1.9.671	161.50 KB (165,376 bytes)	3/24/2005	10:31 PM	International Business Machines Corporation	c:\sqllib\bin\db2genreg64.dll
db2install64	8.1.9.671	21.00 KB (21,504 bytes)	3/24/2005	11:05 PM	International Business Machines Corporation	c:\sqllib\bin\db2install64.dll
db2dascmn64	8.1.9.671	102.50 KB (104,960 bytes)	3/24/2005	10:34 PM	International Business Machines Corporation	c:\sqllib\bin\db2dascmn64.dll
db2osse_db264	8.1.9.671	65.00 KB (66,560 bytes)	3/26/2005	4:34 PM	International Business Machines Corporation	c:\sqllib\bin\db2osse_db264.dll
db2sec64	8.1.9.671	23.00 KB (23,552 bytes)	3/24/2005	10:31 PM	International Business Machines Corporation	c:\sqllib\bin\db2sec64.dll
db2trcapi64	8.1.9.671	33.00 KB (33,792 bytes)	3/24/2005	10:32 PM	International Business Machines Corporation	c:\sqllib\bin\db2trcapi64.dll
db2clfpf	8.1.9.671	958.50 KB (981,504 bytes)	3/24/2005	10:46 PM	International Business Machines Corporation	c:\sqllib\bin\db2clfpf.dll
db2dasapi64	8.1.9.671	450.50 KB (461,312 bytes)	3/24/2005	10:34 PM	International Business Machines Corporation	c:\sqllib\bin\db2dasapi64.dll
db2daskrb64	8.1.9.671	32.50 KB (33,280 bytes)	3/24/2005	10:34 PM	International Business Machines Corporation	c:\sqllib\bin\db2daskrb64.dll
IBMOSAuthclient64	8.1.9.671	27.00 KB (27,648 bytes)	3/24/2005	10:31 PM	International Business Machines Corporation	c:\sqllib\security\plugin\ibm\client\ibmosauthclient64.dll
notepad	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	86.00 KB (88,064 bytes)	3/25/2005	7:00 AM	Microsoft Corporation	c:\windows\system32\notepad.exe
javaw	Not Available	24.09 KB (24,671 bytes)	8/29/2002	9:10 AM	Not Available	c:\program files (x86)\raidman\jre\bin\javaw.exe
mmc	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	1.83 MB (1,920,512 bytes)	3/25/2005	7:00 AM	Microsoft Corporation	c:\windows\system32\mmc.exe
mfc42u	6.50.9146.0	1.39 MB (1,462,272 bytes)	3/25/2005	7:00 AM	Microsoft Corporation	c:\windows\system32\mfc42u.dll
mmcbase	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	106.50 KB (109,056 bytes)	3/25/2005	7:00 AM	Microsoft Corporation	c:\windows\system32\mmcbase.dll
oleacc	4.2.5406.0 (srv03_sp1_rtm.050324-1447)	374.50 KB (383,488 bytes)	3/25/2005	7:00 AM	Microsoft Corporation	c:\windows\system32\oleacc.dll
mmcmdmgr	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	2.23 MB (2,336,256 bytes)	3/25/2005	7:00 AM	Microsoft Corporation	c:\windows\system32\mmcmdmgr.dll
msxml3	8.70.1104.0	2.04 MB (2,141,184 bytes)	3/25/2005	7:00 AM	Microsoft Corporation	c:\windows\system32\msxml3.dll
cmprops	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	313.00 KB (320,512 bytes)	4/1/2005	10:14 AM	Microsoft Corporation	c:\windows\system32\cmprops.dll
mmfutil	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	19.50 KB (19,968 bytes)	4/1/2005	10:14 AM	Microsoft Corporation	c:\windows\system32\mmfutil.dll
ntmsmgr	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	912.50 KB (934,400 bytes)	3/25/2005	7:00 AM	Microsoft Corporation	c:\windows\system32\ntmsmgr.dll
ntmsapi	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	90.50 KB (92,672 bytes)	3/25/2005	7:00 AM	Microsoft Corporation	c:\windows\system32\ntmsapi.dll
els	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	327.00 KB (334,848 bytes)	3/25/2005	7:00 AM	Microsoft Corporation	c:\windows\system32\els.dll
dfrgsnap	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	51.50 KB (52,736 bytes)	3/25/2005	7:00 AM	Microsoft Corp. and Executive Software International, Inc.	c:\windows\system32\dfrgsnap.dll
dfrgres	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	51.50 KB (52,736 bytes)	3/25/2005	7:00 AM	Microsoft Corp. and Executive Software International, Inc.	c:\windows\system32\dfrgres.dll
mycomput	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	176.50 KB (180,736 bytes)	3/25/2005	7:00 AM	Microsoft Corporation	c:\windows\system32\mycomput.dll
filemgmt	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	610.00 KB (624,640 bytes)	3/25/2005	7:00 AM	Microsoft Corporation	c:\windows\system32\filemgmt.dll
cfgmgr32	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	18.00 KB (18,432 bytes)	3/25/2005	7:00 AM	Microsoft Corporation	c:\windows\system32\cfgmgr32.dll
wbemctl	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	316.50 KB (324,096 bytes)	4/1/2005	10:14 AM	Microsoft Corporation	c:\windows\system32\wbem\wbemctl.dll
localsec	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	536.50 KB (549,376 bytes)	3/25/2005	7:00 AM	Microsoft Corporation	c:\windows\system32\localsec.dll
smlogcfg	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	711.50 KB (728,576 bytes)	3/25/2005	7:00 AM	Microsoft Corporation	c:\windows\system32\smlogcfg.dll
odbc32	3.526.1830.0 (srv03_sp1_rtm.050324-1447)	408.00 KB (417,792 bytes)	3/25/2005	7:00 AM	Microsoft Corporation	c:\windows\system32\odbc32.dll
pdh	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	563.00 KB (576,512 bytes)	3/25/2005	7:00 AM	Microsoft Corporation	c:\windows\system32\pdh.dll
odbcbcpl	2000.086.1830.00 (srv03_sp1_rtm.050324-1447)	32.00 KB (32,768 bytes)	3/25/2005	7:00 AM	Microsoft Corporation	c:\windows\system32\odbcbcpl.dll

odbcint 3.526.1830.0 (srv03_sp1_rtm.050324-1447) 96.00 KB (98,304 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\odbcint.dll	vds_ps 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 28.50 KB (29,184 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\vds_ps.dll
snmpsnap 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 312.50 KB (320,000 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\snmpsnap.dll	dmvdsitf 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 172.00 KB (176,128 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\dmvdsitf.dll
dmmskmgr 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 349.50 KB (357,888 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\dmmskmgr.dll	vds 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 418.00 KB (428,032 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\vds.exe
dmutil 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 57.00 KB (58,368 bytes) 3/24/2005 12:16 PM Microsoft Corporation c:\windows\system32\dmutil.dll	osuninst 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 4.50 KB (4,608 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\osuninst.dll
dmmskres 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 116.50 KB (119,296 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\dmmskres.dll	vdsutil 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 78.50 KB (80,384 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\vdsutil.dll
devmgr 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 456.00 KB (466,944 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\devmgr.dll	vdsbas 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 278.00 KB (284,672 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\vdsbas.dll
rasuser 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 278.50 KB (285,184 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\rasuser.dll	fmifs 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 27.50 KB (28,160 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\fmifs.dll
dsprop 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 213.50 KB (218,624 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\dsprop.dll	ulib 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 342.50 KB (350,720 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\ulib.dll
dsuixt 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 145.50 KB (148,992 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\dsuixt.dll	ifsutil 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 120.50 KB (123,392 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\ifsutil.dll
mprsnap 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 1.62 MB (1,703,424 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\mprsnap.dll	vdsdyndr 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 456.50 KB (467,456 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\vdsdyndr.dll
rtrfiltr 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 134.50 KB (137,728 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\rtrfiltr.dll	dmintf 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 23.00 KB (23,552 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\dmintf.dll
servdeps 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 84.00 KB (86,016 bytes) 4/1/2005 10:14 AM Microsoft Corporation c:\windows\system32\servdeps.dll	dmadmin 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 389.50 KB (398,848 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\dmadmin.exe
riched32 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 7.00 KB (7,168 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\riched32.dll	dmconfig 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 545.00 KB (558,080 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\dmconfig.dll
riched20 5.31.23.1224 1.10 MB (1,157,120 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\riched20.dll	wordpad 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 333.50 KB (341,504 bytes) 4/1/2005 10:15 AM Microsoft Corporation c:\program files\windows nt\accessories\wordpad.exe
adsnt 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 535.50 KB (548,352 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\adsnt.dll	msftedit 5.41.21.2505 968.00 KB (991,232 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\msftedit.dll
dmdlgs 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 429.50 KB (439,808 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\dmdlgs.dll	msctf 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 617.50 KB (632,320 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\msctf.dll
dmview 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 107.00 KB (109,568 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\dmview.ocx	dmocx 5.2.3790.1830 (srv03_sp1_rtm.050324-1447) 27.50 KB (28,160 bytes) 3/25/2005 7:00 AM Microsoft Corporation c:\windows\system32\dmocx.dll

helpctr	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	1.30 MB	(1,363,456 bytes)	4/1/2005 10:17 AM	Microsoft Corporation	c:\windows\pchealth\helpctr\binaries\helpctr.exe	Alerter	Alerter	Running	Auto	Share Process	c:\windows\system32\svchost.exe -k localservice	Normal	NT	AUTHORITY\LocalService	0	
hcappres	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	7.50 KB	(7,680 bytes)	4/1/2005 10:17 AM	Microsoft Corporation	c:\windows\pchealth\helpctr\binaries\hcappres.dll	Application Layer Gateway Service		ALG	Stopped	Manual	c:\windows\system32\alg.exe	Normal	NT	AUTHORITY\LocalService	0	
itss	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	208.00 KB	(212,992 bytes)	3/25/2005 7:00 AM	Microsoft Corporation	c:\windows\system32\itss.dll	Application Management	AppMgmt	Stopped	Manual	Share	c:\windows\system32\svchost.exe -k netsvcs	Normal	LocalSystem	0	0	
pchshell	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	155.00 KB	(158,720 bytes)	4/1/2005 10:17 AM	Microsoft Corporation	c:\windows\pchealth\helpctr\binaries\pchshell.dll	ASP.NET State Service	aspnet_state	Stopped	Manual	Own Process	c:\windows\microsoft.net\framework\v1.1.4322\aspnet_state.exe	Normal	NT	AUTHORITY\NetworkService	0	
mshtml	6.00.3790.1830 (srv03_sp1_rtm.050324-1447)	5.65 MB	(5,928,448 bytes)	3/25/2005 7:00 AM	Microsoft Corporation	c:\windows\system32\mshtml.dll	Windows Audio	AudioSrv	Stopped	Disabled	Share Process	c:\windows\system32\svchost.exe -k netsvcs	Normal	LocalSystem	0	0	
msls31	3.10.349.0	357.00 KB	(365,568 bytes)	3/25/2005 7:00 AM	Microsoft Corporation	c:\windows\system32\msls31.dll	Background Intelligent Transfer Service		BITS	Stopped	Manual	c:\windows\system32\svchost.exe -k netsvcs	Normal	LocalSystem	0	0	
msimtf	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	380.50 KB	(389,632 bytes)	3/25/2005 7:00 AM	Microsoft Corporation	c:\windows\system32\msimtf.dll	Computer Browser	Browser	Stopped	Manual	Share Process	c:\windows\system32\svchost.exe -k netsvcs	Normal	LocalSystem	0	0	
jscrip	5.6.0.8827	974.50 KB	(997,888 bytes)	3/25/2005 7:00 AM	Microsoft Corporation	c:\windows\system32\jscrip.dll	Indexing Service	CiSvc	Stopped	Disabled	Share Process	c:\windows\system32\cisvc.exe	Normal	LocalSystem	0	0	
imm32	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	208.00 KB	(212,992 bytes)	3/25/2005 7:00 AM	Microsoft Corporation	c:\windows\system32\imm32.dll	ClipBook	ClipSrv	Stopped	Disabled	Own Process	c:\windows\system32\clipsrv.exe	Normal	LocalSystem	0	0	
mshtml	6.00.3790.1830 (srv03_sp1_rtm.050324-1447)	905.50 KB	(927,232 bytes)	3/25/2005 7:00 AM	Microsoft Corporation	c:\windows\system32\mshtml.dll	COM+ System Application	COMSysApp	Stopped	Manual	Own Process	c:\windows\system32\dlhhost.exe	/processid:{02d4b3f1-fd88-11d1-960d-00805fc79235}	Normal	LocalSystem	0	0
vbscript	5.6.0.8827	646.50 KB	(662,016 bytes)	3/25/2005 7:00 AM	Microsoft Corporation	c:\windows\system32\vbscript.dll	Cryptographic Services		CryptSvc	Stopped	Manual	c:\windows\system32\svchost.exe -k netsvcs	Normal	LocalSystem	0	0	
msinfo	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	636.00 KB	(651,264 bytes)	4/1/2005 10:17 AM	Microsoft Corporation	c:\windows\pchealth\helpctr\binaries\msinfo.dll	DB2 - DB2	DB2	Stopped	Manual	Own Process	c:\sql\lib\bin\db2syscs.exe	Normal	LocalSystem	0	0	
wbemprox	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	38.00 KB	(38,912 bytes)	4/1/2005 10:14 AM	Microsoft Corporation	c:\windows\system32\wbem\wbemprox.dll	DB2DAS - DB2DAS00		DB2DAS00	Stopped	Manual	c:\sql\lib\bin\db2dasrrm.exe	Normal	LocalSystem	0	0	
helpsvc	5.2.3790.1830 (srv03_sp1_rtm.050324-1447)	1.52 MB	(1,591,296 bytes)	4/1/2005 10:17 AM	Microsoft Corporation	c:\windows\pchealth\helpctr\binaries\helpsvc.exe	DB2 Governor	DB2GOVERNOR	Stopped	Manual	Own Process	c:\sql\lib\bin\db2govds.exe	Normal	LocalSystem	0	0	
[Services]							DB2 JDBC Applet Server	DB2JDS	Stopped	Manual	Own Process	c:\sql\lib\bin\db2jds.exe	Normal	LocalSystem	0	0	
Display Name	Name	State	Start Mode	Service Type			DB2 License Server	DB2LICD	Stopped	Manual	Own Process	c:\sql\lib\bin\db2licd.exe	Ignore	LocalSystem	0	0	
Path	Error Control	Start Name	Tag ID				DB2 Registry Reflector		DB2NTREGREFLECTOR	Stopped	Manual	Own Process	c:\sql\lib\bin\db2reg64.exe	Normal	LocalSystem	0	
Application Experience Lookup	Service	AeLookupSvc	Running	Auto	Share Process	c:\windows\system32\svchost.exe -k netsvcs	DB2 Security Server	DB2NTSECSERVER	Stopped	Manual	Own Process	c:\sql\lib\bin\db2sec.exe	Normal	LocalSystem	0	0	
Normal	LocalSystem	0					DCOM Server Process Launcher	DcomLaunch	Running	Auto	Share Process	c:\windows\system32\svchost.exe -k dcomlaunch	Normal	LocalSystem	0	0	

Distributed File System Process	Dfs	Stopped	Manual	Own	LocalSystem	0	TCP/IP NetBIOS Helper Process	LmHosts	Running	Auto	Share	Normal	0
c:\windows\system32\dfsrv.exe							c:\windows\system32\svchost.exe -k localservice						
DHCP Client	Dhcp	Stopped	Manual	Share	Process	0	Messenger	Messenger	Stopped	Disabled	Share	Process	0
c:\windows\system32\svchost.exe -k networkservice							c:\windows\system32\svchost.exe -k netsvcs						
AUTHORITY\NetworkService							NT AUTHORITY\LocalSystem						
Logical Disk Manager	Administrative Service			dmadmin	Running	0	NetMeeting	Remote Desktop Sharing		mnmsrvc	Stopped	Disabled	0
Manual Share Process							c:\windows\system32\dmadmin.exe /com						
Normal LocalSystem							0						
Logical Disk Manager	dmserver	Running	Auto	Share	Process	0	Distributed Transaction Coordinator		MSDTC	Running	Auto	Normal	0
c:\windows\system32\svchost.exe -k netsvcs							c:\windows\system32\msdtc.exe						
Normal LocalSystem							NT AUTHORITY\NetworkService						
DNS Client	Dnscache	Stopped	Manual	Share	Process	0	Windows Installer	MSIServer	Stopped	Manual	Share	Process	0
c:\windows\system32\svchost.exe -k networkservice							c:\windows\system32\msiexec.exe /v						
AUTHORITY\NetworkService							Normal LocalSystem						
Error Reporting Service	ERSvc	Stopped	Manual	Share	Process	0	Network DDE	NetDDE	Stopped	Disabled	Share	Process	0
c:\windows\system32\svchost.exe -k winerr							c:\windows\system32\netdde.exe						
LocalSystem							Normal LocalSystem						
Event Log	Eventlog	Running	Auto	Share	Process	0	Network DDE	DSDMNetDDEsdm		Stopped	Disabled	Share	0
c:\windows\system32\services.exe							c:\windows\system32\netdde.exe						
Normal LocalSystem							Normal LocalSystem						
COM+ Event System	EventSystem		Running	Auto	Share	0	Net Logon	Netlogon	Stopped	Manual	Share	Process	0
c:\windows\system32\svchost.exe -k netsvcs							c:\windows\system32\lsass.exe						
LocalSystem							Normal LocalSystem						
Help and Support	helpsvc	Running	Manual	Share	Process	0	Network Connections	Netman	Running	Manual	Share	Process	0
c:\windows\system32\svchost.exe -k netsvcs							c:\windows\system32\svchost.exe -k netsvcs						
Normal LocalSystem							Normal LocalSystem						
Human Interface Device Access	HidServ	Stopped	Disabled	Share	Process	0	Network Location Awareness (NLA)		Nla	Running	Manual	Normal	0
c:\windows\system32\svchost.exe -k netsvcs							c:\windows\system32\svchost.exe -k netsvcs						
LocalSystem							Normal LocalSystem						
HTTP SSLHTTPFilter		Stopped	Manual	Share	Process	0	File Replication	NtFrs	Stopped	Manual	Own	Process	0
c:\windows\system32\lsass.exe							c:\windows\system32\ntfrs.exe						
Normal LocalSystem							Ignore LocalSystem						
IAS Jet Database Access	IASJet	Stopped	Manual	Share	Process	0	NT LM Security Support Provider		NtLmSsp	Stopped	Manual	Normal	0
c:\windows\systwow64\svchost.exe -k iasjet							c:\windows\system32\lsass.exe						
LocalSystem							Normal LocalSystem						
IMAPI CD-Burning COM Service				ImapiService	Stopped	0	Removable Storage	NtmsSvc	Stopped	Manual	Share	Process	0
Disabled Own Process							c:\windows\system32\imapi.exe						
LocalSystem							Normal LocalSystem						
Intersite Messaging	IsmServ	Stopped	Disabled	Own	Process	0	Plug and Play	PlugPlay	Running	Auto	Share	Process	0
c:\windows\system32\ismssrv.exe							c:\windows\system32\services.exe						
Normal LocalSystem							Normal LocalSystem						
Kerberos Key Distribution Center			kdc	Stopped	Disabled	0	IPSEC Services	PolicyAgent		Stopped	Manual	Share	0
Share Process							c:\windows\system32\lsass.exe						
LocalSystem							Normal LocalSystem						
Server	lanmanserver	Running	Auto	Share	Process	0	Protected Storage	ProtectedStorage		Running	Auto	Share	0
c:\windows\system32\svchost.exe -k netsvcs							c:\windows\system32\lsass.exe						
Normal LocalSystem							Normal LocalSystem						
Workstation	lanmanworkstation	Running	Auto	Share	Process	0	Remote Access Auto Connection Manager		RasAuto	Stopped	Manual	Normal	0
c:\windows\system32\svchost.exe -k netsvcs							c:\windows\system32\svchost.exe -k netsvcs						
LocalSystem							Normal LocalSystem						
License Logging	LicenseService	Stopped	Disabled	Own	Process	0	Remote Access Connection Manager		RasMan	Stopped	Manual	Normal	0
c:\windows\system32\llssrv.exe							c:\windows\system32\svchost.exe -k netsvcs						
AUTHORITY\NetworkService							Normal LocalSystem						
Remote Desktop Help Session Manager				RDSessMgr	Stopped	0	Remote Shell Daemon		Remote Shell Daemon		Running	Normal	0
Manual Own Process							c:\windows\system32\sessmgr.exe						
Normal LocalSystem							0						
Remote Shell Daemon				Remote Shell Daemon		0	Auto Own Process		c:\wrshdnt\wrshdnt.exe		Running	Normal	0
LocalSystem							0						

Routing and Remote Access Share Process LocalSystem	RemoteAccess c:\windows\system32\svchost.exe -k netshvc	Stopped Normal	Disabled NT					Telephony TapiSrv c:\windows\system32\svchost.exe -k tapisrv	Stopped Normal	Manual LocalSystem	Share Process	0	
Remote Registry Process AUTHORITY\LocalService	RemoteRegistry c:\windows\system32\svchost.exe -k regsvc	Running Normal	Auto NT	Share				Terminal Services Process LocalSystem	TermService c:\windows\system32\svchost.exe -k termshvc	Running 0	Manual Normal	Share	
Remote Procedure Call (RPC) Manual NT AUTHORITY\NetworkService	Locator c:\windows\system32\locator.exe	RpcLocator Normal		Stopped				Themes c:\windows\system32\svchost.exe -k netshvc	Themes Normal	Stopped LocalSystem	Disabled LocalSystem	Share Process	0
Remote Procedure Call (RPC) Process AUTHORITY\NetworkService	RpcSs c:\windows\system32\svchost.exe -k rpsvc	Running Normal	Auto NT	Share				Telnet c:\windows\system32\tlntsvr.exe	TlntSvr Normal	Stopped NT AUTHORITY\LocalService	Disabled 0	Own Process	0
Resultant Set of Policy Provider Process LocalSystem	RSOPProv c:\windows\system32\rsopprov.exe	Stopped Normal	Manual	Share				Distributed Link Tracking Server Process LocalSystem	TrkSvr c:\windows\system32\svchost.exe -k netshvc	Stopped 0	Disabled Normal	Share	
Special Administration Console Helper Share Process LocalSystem	sacsrv c:\windows\system32\svchost.exe -k netshvc	Stopped Normal	Manual	Share				Distributed Link Tracking Client Process LocalSystem	TrkWks c:\windows\system32\svchost.exe -k netshvc	Stopped 0	Manual Normal	Share	
Security Accounts Manager Process LocalSystem	SamSs c:\windows\system32\lsass.exe	Running Normal	Auto LocalSystem	Share	0			Terminal Services Session Directory Own Process LocalSystem	Tssdis c:\windows\system32\tssdis.exe	Stopped 0	Disabled Normal	0	
Smart Card c:\windows\system32\scardsvr.exe AUTHORITY\LocalService	SCardSvr c:\windows\system32\scardsvr.exe	Stopped Ignore	Manual NT	Share Process			0	Windows User Mode Driver Framework Own Process NT AUTHORITY\LocalService	UMWdf c:\windows\system32\wdmfr.exe	Stopped 0	Manual Normal	Share	
Task Scheduler c:\windows\system32\svchost.exe -k netshvc	Schedule c:\windows\system32\svchost.exe -k netshvc	Running Normal	Auto	Share Process LocalSystem	0			Uninterruptible Power Supply Process AUTHORITY\LocalService	UPS c:\windows\system32\ups.exe	Stopped 0	Manual NT	Own	
Secondary Logon c:\windows\system32\svchost.exe -k netshvc	seclogon c:\windows\system32\svchost.exe -k netshvc	Running Ignore	Auto	Share Process LocalSystem	0			Virtual Disk Service c:\windows\system32\vds.exe	vds Normal	Running LocalSystem	Manual 0	Own Process	
System Event Notification Process LocalSystem	SENS c:\windows\system32\svchost.exe -k netshvc	Running Normal	Auto	Share				Volume Shadow Copy Process LocalSystem	VSS c:\windows\system32\vssvc.exe	Stopped Normal	Manual LocalSystem	Own	0
ServeRAID Manager Agent Manual Normal	ServeRAIDManagerAgent "c:\program files (x86)\raidman\raidserv.exe"	Stopped 0		Share				Windows Time c:\windows\system32\svchost.exe -k localservice AUTHORITY\LocalService	W32Time Normal	Running 0	Auto NT	Share Process	
Windows Firewall/Internet Connection Sharing (ICS) Stopped -k netshvc	SharedAccess c:\windows\system32\svchost.exe -k netshvc	Stopped Normal	Disabled LocalSystem	Share Process	0			WebClient c:\windows\system32\svchost.exe -k localservice AUTHORITY\LocalService	WebClient Normal	Stopped 0	Disabled Normal	Share Process	
Shell Hardware Detection Share Process LocalSystem	ShellHWDetection c:\windows\system32\svchost.exe -k netshvc	Running 0	Auto	Ignore				WinHTTP Web Proxy Auto-Discovery Service Stopped -k localservice	WinHttpAutoProxySvc c:\windows\system32\svchost.exe -k localservice	Manual Normal	NT AUTHORITY\LocalService	0	
Print Spooler c:\windows\system32\spoolsv.exe	Spooler c:\windows\system32\spoolsv.exe	Stopped Normal	Manual	Own Process LocalSystem	0			Windows Management Instrumentation Share Process LocalSystem	winmgmt c:\windows\system32\svchost.exe -k netshvc	Running 0	Auto	Ignore	
Windows Image Acquisition (WIA) Share Process NT AUTHORITY\LocalService	stisvc c:\windows\system32\svchost.exe -k imgsvc	Stopped Normal	Disabled					Portable Media Serial Number Service Manual Normal	WmdmPmSN c:\windows\system32\svchost.exe -k netshvc	Stopped 0			
Microsoft Software Shadow Copy Provider Own Process LocalSystem	swprv c:\windows\system32\svchost.exe -k swprv	Stopped Normal	Manual					Windows Management Instrumentation Driver Extensions Stopped -k netshvc	Wmi c:\windows\system32\svchost.exe -k netshvc	Manual Normal	LocalSystem	0	
Performance Logs and Alerts Own Process NT Authority\NetworkService	SysmonLog c:\windows\system32\smlogsvc.exe	Stopped Normal	Auto					WMI Performance Adapter Own Process LocalSystem	WmiApSrv c:\windows\system32\wbem\wmiapsrv.exe	Stopped 0	Manual		

Automatic Updates wuauerv Stopped Manual Share Process
 c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem

Wireless Configuration WZCSVC Stopped Manual Share
 Process c:\windows\system32\svchost.exe -k netsvcs Normal
 LocalSystem 0

Network Provisioning Service xmlprov Stopped Manual Share
 Process c:\windows\system32\svchost.exe -k netsvcs Normal
 LocalSystem 0

[Program Groups]

Group Name	Name	User Name
Accessories	Default User:Accessories	Default User
Accessories\Accessibility	Default User:Accessories\Accessibility	Default User
Accessories\Entertainment	Default User:Accessories\Entertainment	Default User
Startup	Default User:Startup	Default User
Accessories	All Users:Accessories	All Users
Accessories\Accessibility	All Users:Accessories\Accessibility	All Users
Accessories\Communications	All Users:Accessories\Communications	All Users
Accessories\Entertainment	All Users:Accessories\Entertainment	All Users
Accessories\System Tools	All Users:Accessories\System Tools	All Users
ActiveState	ActivePerl 5.8	All Users:ActiveState
Administrative Tools	All Users:Administrative Tools	All Users
IBM DB2	All Users:IBM DB2	All Users
IBM DB2\Command Line Tools	All Users:IBM DB2\Command Line Tools	All Users
IBM DB2\Development Tools	All Users:IBM DB2\Development Tools	All Users
IBM DB2\General Administration Tools	All Users:IBM DB2\General Administration Tools	All Users
IBM DB2\Information	All Users:IBM DB2\Information	All Users
IBM DB2\Monitoring Tools	All Users:IBM DB2\Monitoring Tools	All Users
IBM DB2\Set-up Tools	All Users:IBM DB2\Set-up Tools	All Users
ServeRAID Manager	All Users:ServeRAID Manager	All Users

0	Startup	All Users:Startup	All Users
	WinZip	All Users:WinZip	All Users
	Accessories	NT AUTHORITY\SYSTEM:Accessories	NT AUTHORITY\SYSTEM
	Accessories\Accessibility	NT AUTHORITY\SYSTEM:Accessories\Accessibility	NT AUTHORITY\SYSTEM
	Accessories\Entertainment	NT AUTHORITY\SYSTEM:Accessories\Entertainment	NT AUTHORITY\SYSTEM
	Startup	NT AUTHORITY\SYSTEM:Startup	NT AUTHORITY\SYSTEM
	Accessories	DB2SERV2\Administrator:Accessories	DB2SERV2\Administrator
	Accessories\Accessibility	DB2SERV2\Administrator:Accessories\Accessibility	DB2SERV2\Administrator
	Accessories\Entertainment	DB2SERV2\Administrator:Accessories\Entertainment	DB2SERV2\Administrator
	Microsoft Visual C++ Toolkit 2003	DB2SERV2\Administrator:Microsoft Visual C++ Toolkit 2003	DB2SERV2\Administrator
	Startup	DB2SERV2\Administrator:Startup	DB2SERV2\Administrator
	Winsock RSHD-NT	DB2SERV2\Administrator:Winsock RSHD-NT	DB2SERV2\Administrator

[Startup Programs]

Program	Command	User Name	Location
desktop	desktop.ini	NT AUTHORITY\SYSTEM	Startup
desktop	desktop.ini	DB2SERV2\Administrator	Startup
desktop	desktop.ini	DEFAULT	Startup
desktop	desktop.ini	All Users	Common Startup

[OLE Registration]

Object	Local Server
Sound (OLE2)	sndrec32.exe
Media Clip	mplay32.exe

Video Clip mplay32.exe /avi
MIDI Sequence mplay32.exe /mid
Sound Not Available
Media Clip Not Available
WordPad Document "%programfiles%\windows nt\accessories\wordpad.exe"
Bitmap Image mspaint.exe

[Windows Error Reporting]

Time Type Details

[Internet Settings]

[Internet Explorer]

[Following are sub-categories of this main category]

[Summary]

Item	Value
Version	6.0.3790.1830
Build	63790.1830
Application Path	C:\Program Files\Internet Explorer
Language	English (United States)
Active Printer	Not Available
Cipher Strength	128-bit
Content Advisor	Disabled
IEAK Install	No

[File Versions]

File	Version	Size	Date	Path	Company
------	---------	------	------	------	---------

actxprxy.dll	6.0.3790.1830	221 KB	3/25/2005 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
advpack.dll	6.0.3790.1830	146 KB	3/25/2005 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
asctrls.ocx	6.0.3790.1830	147 KB	3/25/2005 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
browsecl.dll	6.0.3790.1830	63 KB	3/25/2005 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
browseui.dll	6.0.3790.1830	1,564 KB	3/25/2005 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
cdfview.dll	6.0.3790.1830	216 KB	3/25/2005 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
comctl32.dll	5.82.3790.1830	935 KB	3/25/2005 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
dxtrans.dll	6.3.3790.1830	320 KB	3/25/2005 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
dxtmsft.dll	6.3.3790.1830	549 KB	3/25/2005 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
iecont.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
iecontlc.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
iedkcs32.dll	16.0.3790.1830	417 KB	3/25/2005 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
iepeers.dll	6.0.3790.1830	361 KB	3/25/2005 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
iesetup.dll	6.0.3790.1830	71 KB	3/25/2005 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
ieuinit.inf	Not Available	24 KB	3/25/2005 8:00:00 AM	C:\WINDOWS\system32	Not Available
ieexplore.exe	6.0.3790.1830	94 KB	3/25/2005 8:00:00 AM	C:\Program Files\Internet Explorer	Microsoft Corporation
imgutil.dll	6.0.3790.1830	61 KB	3/25/2005 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
inetcp1.cpl	6.0.3790.1830	428 KB	3/25/2005 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
inetcp1c.dll	6.0.3790.1830	110 KB	3/25/2005 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
inseng.dll	6.0.3790.1830	147 KB	3/25/2005 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
mlang.dll	6.0.3790.1830	686 KB	3/25/2005 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation
msencode.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
mshta.exe	6.0.3790.1830	38 KB	3/25/2005 8:00:00 AM	C:\WINDOWS\system32	Microsoft Corporation

mshtml.dll	6.0.3790.1830	5,790 KB	3/25/2005 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation			
mshtml.tlb	6.0.3790.1830	1,320 KB	3/25/2005 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation			
mshtml.dll	6.0.3790.1830	906 KB	3/25/2005 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation			
mshtml.dll	6.0.3790.1830	56 KB	3/25/2005 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation			
msident.dll	6.0.3790.1830	69 KB	3/25/2005 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation			
msident.dll	6.0.3790.1830	16 KB	3/25/2005 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation			
msieftp.dll	6.0.3790.1830	369 KB	3/25/2005 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation			
msrating.dll	6.0.3790.1830	240 KB	3/25/2005 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation			
mstime.dll	6.0.3790.1830	878 KB	3/25/2005 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation			
occache.dll	6.0.3790.1830	126 KB	3/25/2005 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation			
proctexe.ocx	<File Missing>	Not Available	Not Available
Available	Not Available	Not Available	Not Available
sendmail.dll	6.0.3790.1830	64 KB	3/25/2005 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation			
shdoclc.dll	6.0.3790.1830	590 KB	3/25/2005 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation			
shdocvw.dll	6.0.3790.1830	2,360 KB	3/25/2005 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation			
shfolder.dll	6.0.3790.1830	34 KB	3/25/2005 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation			
shlwapi.dll	6.0.3790.1830	607 KB	3/25/2005 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation			
tdc.ocx	1.3.0.3130	91 KB	3/25/2005 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation			
url.dll	6.0.3790.1830	40 KB	3/25/2005 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation			
urlmon.dll	6.0.3790.1830	1,049 KB	3/25/2005 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation			
webcheck.dll	6.0.3790.1830	439 KB	3/25/2005 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation			
wininet.dll	6.0.3790.1830	1,159 KB	3/25/2005 8:00:00 AM
C:\WINDOWS\system32 Microsoft Corporation			

[Connectivity]

Item	Value
------	-------

Connection Preference	Never dial
LAN Settings	
AutoConfigProxy	wininet.dll
AutoProxyDetectMode	Disabled
AutoConfigURL	
Proxy	Disabled
ProxyServer	
ProxyOverride	
[Cache]	
[Following are sub-categories of this main category]	
[Summary]	
Item	Value
Page Refresh Type	Automatic
Temporary Internet Files Folder C:\Documents and Settings\Administrator\Local Settings\Temporary Internet Files	
Total Disk Space	Not Available
Available Disk Space Not Available	
Maximum Cache Size Not Available	
Available Cache Size Not Available	
[List of Objects]	
Program File	Status CodeBase
No cached object information available	
[Content]	
[Following are sub-categories of this main category]	
[Summary]	
Item	Value
Content Advisor	Disabled
[Personal Certificates]	
Issued To	Issued By Validity Signature Algorithm
No personal certificate information available	
[Other People Certificates]	
Issued To	Issued By Validity Signature Algorithm
No other people certificate information available	
[Publishers]	
Name	
No publisher information available	

[Security]

Zone	Security Level
My Computer	Custom
Local intranet	Custom
Trusted sites	Custom
Internet	High
Restricted sites	Custom

ServeRAID-6M Disk Controller Configuration Parameters

May 20, 2005 10:25:55 AM EDT

Configuration summary

```

Server name.....db2serv2
ServeRAID Manager agent.....6.10.26 (1253)
ServeRAID Manager console.....6.10.26 (1253)
Number of controllers.....2
Operating system.....Windows 2003

```

Configuration information for controller 2

```

Controller type.....ServeRAID-6M
SCSI backend type.....AIC-7902
SCSI backend revision.....3
Controller FRU.....02R0985
Battery FRU.....02R0986
Serial number.....60B73A57
Part number.....32P0033
Physical slot.....12
BIOS version.....7.10.23
Firmware version.....7.10.23
Device driver version.....7.10.53
Battery-backup cache.....Installed
Battery temperature.....Normal
Battery charge level.....100 %
Battery-backup cache size.....128 MB
Read-ahead cache mode.....Adaptive
Stripe-unit size.....64 KB
Rebuild rate.....High
Hot-swap rebuild.....Enabled
Copy back.....Enabled
Data scrubbing.....Enabled
Auto-synchronization.....Enabled
Clustering.....Disabled
Unattended mode.....Disabled
BIOS-compatibility mapping.....Limited
Number of arrays.....1
Number of logical drives.....1
Number of hot-spare drives.....0
Number of ready drives.....0

```

Spanned array 1

```

Array identifier.....1
Array size.....694300 MB
Free space.....0 MB

```

```

Number of logical drives.....1
Number of physical drives.....20

```

Arrays in spanned array 1

```

-----
Array identifier.....A
Array size.....347150 MB
Stripe order (channel/device)...1/0 1/1 1/2 1/3 1/4 2/0 2/1 2/2 2/3 2/4
Number of physical drives.....10

```

```

Array identifier.....B
Array size.....347150 MB
Stripe order (channel/device)...2/5 2/6 2/8 2/9 2/10 1/5 1/6 1/8 1/9 1/10
Number of physical drives.....10

```

Physical drives in array A

```

-----
Type.....Hard disk drive
Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0R1KG
Firmware level.....B85B
Channel.....1
SCSI ID.....0
Size.....34715 MB
State.....Online
Array letter.....A
PFA error.....No

```

```

Type.....Hard disk drive
Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0MHQ8
Firmware level.....B85B
Channel.....1
SCSI ID.....1
Size.....34715 MB
State.....Online
Array letter.....A
PFA error.....No

```

```

Type.....Hard disk drive
Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0MHK7
Firmware level.....B85B
Channel.....1
SCSI ID.....2
Size.....34715 MB
State.....Online
Array letter.....A
PFA error.....No

```

```

Type.....Hard disk drive
Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0R1NK
Firmware level.....B85B
Channel.....1
SCSI ID.....3
Size.....34715 MB
State.....Online
Array letter.....A

```

PFA error.....No
Type.....Hard disk drive
Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0R1HT
Firmware level.....B85B
Channel.....1
SCSI ID.....4
Size.....34715 MB
State.....Online
Array letter.....A
PFA error.....No

Type.....Hard disk drive
Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0Q4ZT
Firmware level.....B85B
Channel.....2
SCSI ID.....0
Size.....34715 MB
State.....Online
Array letter.....A
PFA error.....No

Type.....Hard disk drive
Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0R1JG
Firmware level.....B85B
Channel.....2
SCSI ID.....1
Size.....34715 MB
State.....Online
Array letter.....A
PFA error.....No

Type.....Hard disk drive
Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0R1R6
Firmware level.....B85B
Channel.....2
SCSI ID.....2
Size.....34715 MB
State.....Online
Array letter.....A
PFA error.....No

Type.....Hard disk drive
Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0MLFE
Firmware level.....B85B
Channel.....2
SCSI ID.....3
Size.....34715 MB
State.....Online
Array letter.....A
PFA error.....No

Type.....Hard disk drive

Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0PPYX
Firmware level.....B85B
Channel.....2
SCSI ID.....4
Size.....34715 MB
State.....Online
Array letter.....A
PFA error.....No

Physical drives in array B

Type.....Hard disk drive
Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0PF7M
Firmware level.....B85B
Channel.....2
SCSI ID.....5
Size.....34715 MB
State.....Online
Array letter.....B
PFA error.....No

Type.....Hard disk drive
Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0R16T
Firmware level.....B85B
Channel.....2
SCSI ID.....6
Size.....34715 MB
State.....Online
Array letter.....B
PFA error.....No

Type.....Hard disk drive
Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0QPHW
Firmware level.....B85B
Channel.....2
SCSI ID.....8
Size.....34715 MB
State.....Online
Array letter.....B
PFA error.....No

Type.....Hard disk drive
Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0R1GN
Firmware level.....B85B
Channel.....2
SCSI ID.....9
Size.....34715 MB
State.....Online
Array letter.....B
PFA error.....No

Type.....Hard disk drive

Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0R139
Firmware level.....B85B
Channel.....2
SCSI ID.....10
Size.....34715 MB
State.....Online
Array letter.....B
PFA error.....No

Type.....Hard disk drive
Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0QG2J
Firmware level.....B85B
Channel.....1
SCSI ID.....5
Size.....34715 MB
State.....Online
Array letter.....B
PFA error.....No

Type.....Hard disk drive
Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0PFE6
Firmware level.....B85B
Channel.....1
SCSI ID.....6
Size.....34715 MB
State.....Online
Array letter.....B
PFA error.....No

Type.....Hard disk drive
Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0R0XP
Firmware level.....B85B
Channel.....1
SCSI ID.....8
Size.....34715 MB
State.....Online
Array letter.....B
PFA error.....No

Type.....Hard disk drive
Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0MK3Z
Firmware level.....B85B
Channel.....1
SCSI ID.....9
Size.....34715 MB
State.....Online
Array letter.....B
PFA error.....No

Type.....Hard disk drive
Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736

Serial number.....3HX0QNM9
Firmware level.....B85B
Channel.....1
SCSI ID.....10
Size.....34715 MB
State.....Online
Array letter.....B
PFA error.....No

Logical drives in spanned array 1

Logical drive.....1
Spanned array number1
State.....Okay
RAID level.....1E0
Data space.....347150 MB
Parity space.....347150 MB
Date created.....05/02/2005
Write-cache mode.....Write through
Merge-group number.....207
Merge-group state.....Non-shared

SCSI channel 1

Number of drives.....10
SCSI transfer speed.....Optimal
SCSI initiator ID.....7

Type.....Hard disk drive
Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0R1KG
Firmware level.....B85B
Channel.....1
SCSI ID.....0
Size.....34715 MB
State.....Online
Array letter.....A
PFA error.....No

Type.....Hard disk drive
Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0MHQ8
Firmware level.....B85B
Channel.....1
SCSI ID.....1
Size.....34715 MB
State.....Online
Array letter.....A
PFA error.....No

Type.....Hard disk drive
Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0MHK7
Firmware level.....B85B
Channel.....1
SCSI ID.....2
Size.....34715 MB
State.....Online
Array letter.....A
PFA error.....No

Type.....Hard disk drive
Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0R1NK
Firmware level.....B85B
Channel.....1
SCSI ID.....3
Size.....34715 MB
State.....Online
Array letter.....A
PFA error.....No

Type.....Hard disk drive
Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0R1HT
Firmware level.....B85B
Channel.....1
SCSI ID.....4
Size.....34715 MB
State.....Online
Array letter.....A
PFA error.....No

Type.....Hard disk drive
Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0QG2J
Firmware level.....B85B
Channel.....1
SCSI ID.....5
Size.....34715 MB
State.....Online
Array letter.....B
PFA error.....No

Type.....Hard disk drive
Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0PFE6
Firmware level.....B85B
Channel.....1
SCSI ID.....6
Size.....34715 MB
State.....Online
Array letter.....B
PFA error.....No

Type.....Hard disk drive
Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0R0XP
Firmware level.....B85B
Channel.....1
SCSI ID.....8
Size.....34715 MB
State.....Online
Array letter.....B
PFA error.....No

Type.....Hard disk drive
Vendor.....IBM-ESXS

Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0MK3Z
Firmware level.....B85B
Channel.....1
SCSI ID.....9
Size.....34715 MB
State.....Online
Array letter.....B
PFA error.....No

Type.....Hard disk drive
Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0QNM9
Firmware level.....B85B
Channel.....1
SCSI ID.....10
Size.....34715 MB
State.....Online
Array letter.....B
PFA error.....No

Type.....Enclosure
Vendor.....IBM
Product or model number.....EXP400
Serial number.....23M0095
Firmware level.....D110
FRU type.....MIDPLANE
FRU vendor.....IBM
FRU date of manufacture.....07/2003
FRU part number.....59P4865
FRU serial number.....137B342
FRU type.....CARD
FRU vendor.....IBM
FRU date of manufacture.....07/2003
FRU part number.....59P4866
FRU serial number.....1R04337B344
FRU type.....Power
FRU vendor.....IBM
FRU date of manufacture.....07/2003
FRU part number.....71P8146
FRU serial number.....1R04537B346
FRU type.....Power
FRU vendor.....IBM
FRU date of manufacture.....07/2003
FRU part number.....71P8146
FRU serial number.....1R04537B347
FRU type.....CARD
FRU vendor.....IBM
FRU date of manufacture.....07/2003
FRU part number.....59P4869
FRU serial number.....1R04437B345
Channel.....1
SCSI ID.....15
Enclosure ID.....0
Enclosure status.....Okay
Fan 1 status.....Okay
Fan 2 status.....Okay
Power supply 1 status.....Okay
Power supply 2 status.....Okay
Temperature status.....Normal

SCSI channel 2

Number of drives.....10
SCSI transfer speed.....Optimal

SCSI initiator ID.....7
Type.....Hard disk drive
Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0Q4ZT
Firmware level.....B85B
Channel.....2
SCSI ID.....0
Size.....34715 MB
State.....Online
Array letter.....A
PFA error.....No

Type.....Hard disk drive
Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0R1JG
Firmware level.....B85B
Channel.....2
SCSI ID.....1
Size.....34715 MB
State.....Online
Array letter.....A
PFA error.....No

Type.....Hard disk drive
Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0R1R6
Firmware level.....B85B
Channel.....2
SCSI ID.....2
Size.....34715 MB
State.....Online
Array letter.....A
PFA error.....No

Type.....Hard disk drive
Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0MLFE
Firmware level.....B85B
Channel.....2
SCSI ID.....3
Size.....34715 MB
State.....Online
Array letter.....A
PFA error.....No

Type.....Hard disk drive
Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0PPYX
Firmware level.....B85B
Channel.....2
SCSI ID.....4
Size.....34715 MB
State.....Online
Array letter.....A
PFA error.....No

Type.....Hard disk drive

Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0PF7M
Firmware level.....B85B
Channel.....2
SCSI ID.....5
Size.....34715 MB
State.....Online
Array letter.....B
PFA error.....No

Type.....Hard disk drive
Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0R16T
Firmware level.....B85B
Channel.....2
SCSI ID.....6
Size.....34715 MB
State.....Online
Array letter.....B
PFA error.....No

Type.....Hard disk drive
Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0QPHW
Firmware level.....B85B
Channel.....2
SCSI ID.....8
Size.....34715 MB
State.....Online
Array letter.....B
PFA error.....No

Type.....Hard disk drive
Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0R1GN
Firmware level.....B85B
Channel.....2
SCSI ID.....9
Size.....34715 MB
State.....Online
Array letter.....B
PFA error.....No

Type.....Hard disk drive
Vendor.....IBM-ESXS
Product or model number.....ST336753
FRU part number.....32P0736
Serial number.....3HX0R139
Firmware level.....B85B
Channel.....2
SCSI ID.....10
Size.....34715 MB
State.....Online
Array letter.....B
PFA error.....No

Type.....Enclosure
Vendor.....IBM
Product or model number.....EXP400
Serial number.....23M0063

Firmware level.....D110
 FRU type.....MIDPLANE
 FRU vendor.....IBM
 FRU date of manufacture.....07/2003
 FRU part number.....59P4865
 FRU serial number.....1379007
 FRU type.....CARD
 FRU vendor.....IBM
 FRU date of manufacture.....07/2003
 FRU part number.....59P4866
 FRU serial number.....1R043379009
 FRU type.....Power
 FRU vendor.....IBM
 FRU date of manufacture.....07/2003
 FRU part number.....71P8146
 FRU serial number.....1R045379011
 FRU type.....Power
 FRU vendor.....IBM
 FRU date of manufacture.....07/2003
 FRU part number.....71P8146
 FRU serial number.....1R045379012
 FRU type.....CARD
 FRU vendor.....IBM
 FRU date of manufacture.....07/2003
 FRU part number.....59P4869
 FRU serial number.....1R044379010
 Channel.....2
 SCSI ID.....15
 Enclosure ID.....0
 Enclosure status.....Okay
 Fan 1 status.....Okay
 Fan 2 status.....Okay
 Power supply 1 status.....Okay
 Power supply 2 status.....Okay
 Temperature status.....Normal

End of the configuration information for controller 2

DS4500 Disk Subsystem Configuration

Rack 1

PROFILE FOR STORAGE SUBSYSTEM: Rack_1 (5/20/05 8:49:11 AM)

SUMMARY-----

Number of controllers: 2
 Number of arrays: 14
 Total number of logical drives (includes an access logical drive): 15 of 2048 used
 Number of standard logical drives: 14
 Number of access logical drives: 1
 Number of drives: 140
 Supported drive types: Fibre (140)
 Total hot spare drives: 0
 Standby: 0
 In use: 0
 Access logical drive: None mapped
 Default host type: Windows 2000/Server 2003 Non-Clustered (Host type index 2)
 Current configuration
 Firmware version: 06.12.03.00
 NVSRAM version: N1742F900R912V06
 Pending configuration
 Staged firmware download supported?: Yes
 Firmware version: None

NVSRAM version: None
 Transferred on: None
 NVSRAM configured for batteries?: Yes
 Start cache flushing at (in percentage): 8
 Stop cache flushing at (in percentage): 8
 Cache block size (in KB): 16
 Media scan frequency (in days): Disabled
 Failover alert delay (in minutes): 5
 Feature enable identifier: 38333930340035353735390040082413
 Storage Subsystem worldwide name (ID):
 600A0B80001363C30000000041659740

CONTROLLERS-----

Number of controllers: 2

Controller in Slot A

Status: Online
 Current configuration
 Firmware version: 06.12.03.00
 Appware version: 06.12.03.00
 Bootware version: 06.10.04.00
 NVSRAM version: N1742F900R912V06
 Pending configuration
 Firmware version: None
 Appware version: None
 Bootware version: None
 NVSRAM version: None
 Transferred on: None
 Board ID: 5884
 Product ID: 1742-900
 Product revision: 0520
 Serial number: 1T40668523
 Date of manufacture: February 20, 2004
 Cache/processor size (MB): 1024/128
 Date/Time: Fri May 20 08:44:53 EDT 2005
 Associated Logical Drives (* = Preferred Owner):
 RACK1LUN0*, RACK1LUN10*, RACK1LUN12*, RACK1LUN2*,
 RACK1LUN4*, RACK1LUN6*,
 RACK1LUN8*
 Ethernet port: 1
 MAC address: 00:a0:b8:13:63:c3
 Host name: FLUTE_00A
 Network configuration: Static
 IP address: 192.168.122.240
 Subnet mask: 255.255.255.0
 Gateway: 0.0.0.0
 Remote login: Disabled
 Drive interface: Fibre
 Channel: 1
 Current ID: 125/0x1
 Maximum data rate: 2 Gbps
 Current data rate: 2 Gbps
 Data rate control: Switch
 Link status: Up
 Drive interface: Fibre
 Channel: 2
 Current ID: 125/0x1
 Maximum data rate: 2 Gbps
 Current data rate: 2 Gbps
 Data rate control: Switch
 Link status: Up
 Drive interface: Fibre
 Channel: 3
 Current ID: 125/0x1
 Maximum data rate: 2 Gbps
 Current data rate: 2 Gbps
 Data rate control: Switch

Link status: Up
 Drive interface: Fibre
 Channel: 4
 Current ID: 125/0x1
 Maximum data rate: 2 Gbps
 Current data rate: 2 Gbps
 Data rate control: Switch
 Link status: Up
 Host interface: Fibre
 Port: 1
 Current ID: 0/0xEF
 Preferred ID: 0/0xEF
 NL-Port ID: 0x0000EF
 Maximum data rate: 2 Gbps
 Current data rate: 2 Gbps
 Data rate control: Switch
 Link status: Up
 Topology: Arbitrated Loop - Private
 World-wide port name: 20:02:00:a0:b8:13:63:c4
 World-wide node name: 20:02:00:a0:b8:13:63:c3
 Part type: HPFC-5400 revision 6
 Host interface: Fibre
 Port: 2
 Current ID: Not applicable/0xFFFFFFFF
 Preferred ID: 1/0xE8
 NL-Port ID: 0x000000
 Maximum data rate: 2 Gbps
 Current data rate: 1 Gbps
 Data rate control: Switch
 Link status: Down
 Topology: Not available
 World-wide port name: 20:02:00:a0:b8:13:63:c5
 World-wide node name: 20:02:00:a0:b8:13:63:c3
 Part type: HPFC-5400 revision 6
 Controller in Slot B
 Status: Online
 Current configuration
 Firmware version: 06.12.03.00
 Appware version: 06.12.03.00
 Bootware version: 06.10.04.00
 NVSRAM version: N1742F900R912V06
 Pending configuration
 Firmware version: None
 Appware version: None
 Bootware version: None
 NVSRAM version: None
 Transferred on: None
 Board ID: 5884
 Product ID: 1742-900
 Product revision: 0520
 Serial number: 1T40667443
 Date of manufacture: February 12, 2004
 Cache/processor size (MB): 1024/128
 Date/Time: Fri May 20 08:44:42 EDT 2005
 Associated Logical Drives (* = Preferred Owner):
 RACK1LUN1*, RACK1LUN11*, RACK1LUN13*, RACK1LUN3*,
 RACK1LUN5*, RACK1LUN7*,
 RACK1LUN9*
 Ethernet port: 1
 MAC address: 00:a0:b8:12:a4:b8
 Host name: FLUTE_00B
 Network configuration: Static
 IP address: 192.168.122.241
 Subnet mask: 255.255.255.0
 Gateway: 0.0.0.0
 Remote login: Disabled

Drive interface: Fibre
 Channel: 1
 Current ID: 124/0x2
 Maximum data rate: 2 Gbps
 Current data rate: 2 Gbps
 Data rate control: Switch
 Link status: Up
 Drive interface: Fibre
 Channel: 2
 Current ID: 124/0x2
 Maximum data rate: 2 Gbps
 Current data rate: 2 Gbps
 Data rate control: Switch
 Link status: Up
 Drive interface: Fibre
 Channel: 3
 Current ID: 124/0x2
 Maximum data rate: 2 Gbps
 Current data rate: 2 Gbps
 Data rate control: Switch
 Link status: Up
 Drive interface: Fibre
 Channel: 4
 Current ID: 124/0x2
 Maximum data rate: 2 Gbps
 Current data rate: 2 Gbps
 Data rate control: Switch
 Link status: Up
 Host interface: Fibre
 Port: 1
 Current ID: 2/0xE4
 Preferred ID: 2/0xE4
 NL-Port ID: 0x0000E4
 Maximum data rate: 2 Gbps
 Current data rate: 2 Gbps
 Data rate control: Switch
 Link status: Up
 Topology: Arbitrated Loop - Private
 World-wide port name: 20:03:00:a0:b8:13:63:c4
 World-wide node name: 20:03:00:a0:b8:13:63:c3
 Part type: HPFC-5400 revision 6
 Host interface: Fibre
 Port: 2
 Current ID: Not applicable/0xFFFFFFFF
 Preferred ID: 3/0xE2
 NL-Port ID: 0x000000
 Maximum data rate: 2 Gbps
 Current data rate: 1 Gbps
 Data rate control: Switch
 Link status: Down
 Topology: Not available
 World-wide port name: 20:03:00:a0:b8:13:63:c5
 World-wide node name: 20:03:00:a0:b8:13:63:c3
 Part type: HPFC-5400 revision 6

ARRAYS-----

Number of arrays: 14

Array 1 (RAID 0)
 Status: Online
 Drive type: Fibre Channel
 Enclosure loss protection: No
 Current owner: Controller in slot A
 Associated logical drives and free capacities:
 RACK1LUN0 (334.014 GB)
 Associated drives (in piece order):
 Drive at Enclosure 0, Slot 1
 Drive at Enclosure 1, Slot 1

Drive at Enclosure 2, Slot 1
Drive at Enclosure 3, Slot 1
Drive at Enclosure 4, Slot 1
Drive at Enclosure 10, Slot 1
Drive at Enclosure 11, Slot 1
Drive at Enclosure 12, Slot 1
Drive at Enclosure 13, Slot 1
Drive at Enclosure 14, Slot 1

Array 2 (RAID 0)

Status: Online

Drive type: Fibre Channel

Enclosure loss protection: No

Current owner: Controller in slot B

Associated logical drives and free capacities:

RACK1LUN1 (334.014 GB)

Associated drives (in piece order):

Drive at Enclosure 0, Slot 2
Drive at Enclosure 1, Slot 2
Drive at Enclosure 2, Slot 2
Drive at Enclosure 3, Slot 2
Drive at Enclosure 4, Slot 2
Drive at Enclosure 10, Slot 2
Drive at Enclosure 11, Slot 2
Drive at Enclosure 12, Slot 2
Drive at Enclosure 13, Slot 2
Drive at Enclosure 14, Slot 2

Array 3 (RAID 0)

Status: Online

Drive type: Fibre Channel

Enclosure loss protection: No

Current owner: Controller in slot A

Associated logical drives and free capacities:

RACK1LUN2 (334.014 GB)

Associated drives (in piece order):

Drive at Enclosure 0, Slot 3
Drive at Enclosure 1, Slot 3
Drive at Enclosure 2, Slot 3
Drive at Enclosure 3, Slot 3
Drive at Enclosure 4, Slot 3
Drive at Enclosure 10, Slot 3
Drive at Enclosure 11, Slot 3
Drive at Enclosure 12, Slot 3
Drive at Enclosure 13, Slot 3
Drive at Enclosure 14, Slot 3

Array 4 (RAID 0)

Status: Online

Drive type: Fibre Channel

Enclosure loss protection: No

Current owner: Controller in slot B

Associated logical drives and free capacities:

RACK1LUN3 (334.014 GB)

Associated drives (in piece order):

Drive at Enclosure 0, Slot 4
Drive at Enclosure 1, Slot 4
Drive at Enclosure 2, Slot 4
Drive at Enclosure 3, Slot 4
Drive at Enclosure 4, Slot 4
Drive at Enclosure 10, Slot 4
Drive at Enclosure 11, Slot 4
Drive at Enclosure 12, Slot 4
Drive at Enclosure 13, Slot 4
Drive at Enclosure 14, Slot 4

Array 5 (RAID 0)

Status: Online

Drive type: Fibre Channel

Enclosure loss protection: No

Current owner: Controller in slot A

Associated logical drives and free capacities:

RACK1LUN4 (334.014 GB)

Associated drives (in piece order):

Drive at Enclosure 0, Slot 5
Drive at Enclosure 1, Slot 5
Drive at Enclosure 2, Slot 5
Drive at Enclosure 3, Slot 5
Drive at Enclosure 4, Slot 5
Drive at Enclosure 10, Slot 5
Drive at Enclosure 11, Slot 5
Drive at Enclosure 12, Slot 5
Drive at Enclosure 13, Slot 5
Drive at Enclosure 14, Slot 5

Array 6 (RAID 0)

Status: Online

Drive type: Fibre Channel

Enclosure loss protection: No

Current owner: Controller in slot B

Associated logical drives and free capacities:

RACK1LUN5 (334.014 GB)

Associated drives (in piece order):

Drive at Enclosure 0, Slot 6
Drive at Enclosure 1, Slot 6
Drive at Enclosure 2, Slot 6
Drive at Enclosure 3, Slot 6
Drive at Enclosure 4, Slot 6
Drive at Enclosure 10, Slot 6
Drive at Enclosure 11, Slot 6
Drive at Enclosure 12, Slot 6
Drive at Enclosure 13, Slot 6
Drive at Enclosure 14, Slot 6

Array 7 (RAID 0)

Status: Online

Drive type: Fibre Channel

Enclosure loss protection: No

Current owner: Controller in slot A

Associated logical drives and free capacities:

RACK1LUN6 (334.014 GB)

Associated drives (in piece order):

Drive at Enclosure 0, Slot 7
Drive at Enclosure 1, Slot 7
Drive at Enclosure 2, Slot 7
Drive at Enclosure 3, Slot 7
Drive at Enclosure 4, Slot 7
Drive at Enclosure 10, Slot 7
Drive at Enclosure 12, Slot 7
Drive at Enclosure 13, Slot 7
Drive at Enclosure 14, Slot 7
Drive at Enclosure 11, Slot 7

Array 8 (RAID 0)

Status: Online

Drive type: Fibre Channel

Enclosure loss protection: No

Current owner: Controller in slot B

Associated logical drives and free capacities:

RACK1LUN7 (334.014 GB)

Associated drives (in piece order):

Drive at Enclosure 0, Slot 8
Drive at Enclosure 1, Slot 8
Drive at Enclosure 2, Slot 8
Drive at Enclosure 3, Slot 8
Drive at Enclosure 4, Slot 8
Drive at Enclosure 10, Slot 8
Drive at Enclosure 11, Slot 8
Drive at Enclosure 12, Slot 8
Drive at Enclosure 13, Slot 8
Drive at Enclosure 14, Slot 8

Array 9 (RAID 0)

Status: Online

Drive type: Fibre Channel
 Enclosure loss protection: No
 Current owner: Controller in slot A
 Associated logical drives and free capacities:
 RACK1LUN8 (334.014 GB)
 Associated drives (in piece order):
 Drive at Enclosure 0, Slot 9
 Drive at Enclosure 1, Slot 9
 Drive at Enclosure 2, Slot 9
 Drive at Enclosure 3, Slot 9
 Drive at Enclosure 4, Slot 9
 Drive at Enclosure 10, Slot 9
 Drive at Enclosure 11, Slot 9
 Drive at Enclosure 12, Slot 9
 Drive at Enclosure 13, Slot 9
 Drive at Enclosure 14, Slot 9

Array 10 (RAID 0)
 Status: Online
 Drive type: Fibre Channel
 Enclosure loss protection: No
 Current owner: Controller in slot B
 Associated logical drives and free capacities:
 RACK1LUN9 (334.014 GB)
 Associated drives (in piece order):
 Drive at Enclosure 0, Slot 10
 Drive at Enclosure 1, Slot 10
 Drive at Enclosure 2, Slot 10
 Drive at Enclosure 3, Slot 10
 Drive at Enclosure 4, Slot 10
 Drive at Enclosure 10, Slot 10
 Drive at Enclosure 11, Slot 10
 Drive at Enclosure 12, Slot 10
 Drive at Enclosure 13, Slot 10
 Drive at Enclosure 14, Slot 10

Array 11 (RAID 0)
 Status: Online
 Drive type: Fibre Channel
 Enclosure loss protection: No
 Current owner: Controller in slot A
 Associated logical drives and free capacities:
 RACK1LUN10 (334.014 GB)
 Associated drives (in piece order):
 Drive at Enclosure 0, Slot 11
 Drive at Enclosure 1, Slot 11
 Drive at Enclosure 2, Slot 11
 Drive at Enclosure 3, Slot 11
 Drive at Enclosure 4, Slot 11
 Drive at Enclosure 10, Slot 11
 Drive at Enclosure 11, Slot 11
 Drive at Enclosure 12, Slot 11
 Drive at Enclosure 13, Slot 11
 Drive at Enclosure 14, Slot 11

Array 12 (RAID 0)
 Status: Online
 Drive type: Fibre Channel
 Enclosure loss protection: No
 Current owner: Controller in slot B
 Associated logical drives and free capacities:
 RACK1LUN11 (334.014 GB)
 Associated drives (in piece order):
 Drive at Enclosure 0, Slot 12
 Drive at Enclosure 1, Slot 12
 Drive at Enclosure 2, Slot 12
 Drive at Enclosure 3, Slot 12
 Drive at Enclosure 4, Slot 12
 Drive at Enclosure 10, Slot 12
 Drive at Enclosure 11, Slot 12
 Drive at Enclosure 12, Slot 12

Drive at Enclosure 13, Slot 12
 Drive at Enclosure 14, Slot 12
 Array 13 (RAID 5)
 Status: Online
 Drive type: Fibre Channel
 Enclosure loss protection: Yes
 Current owner: Controller in slot A
 Associated logical drives and free capacities:
 RACK1LUN12 (300.612 GB)
 Associated drives (in piece order):
 Drive at Enclosure 0, Slot 13
 Drive at Enclosure 1, Slot 13
 Drive at Enclosure 2, Slot 13
 Drive at Enclosure 3, Slot 13
 Drive at Enclosure 4, Slot 13
 Drive at Enclosure 10, Slot 13
 Drive at Enclosure 11, Slot 13
 Drive at Enclosure 12, Slot 13
 Drive at Enclosure 13, Slot 13
 Drive at Enclosure 14, Slot 13

Array 14 (RAID 5)
 Status: Online
 Drive type: Fibre Channel
 Enclosure loss protection: Yes
 Current owner: Controller in slot B
 Associated logical drives and free capacities:
 RACK1LUN13 (300.612 GB)
 Associated drives (in piece order):
 Drive at Enclosure 0, Slot 14
 Drive at Enclosure 1, Slot 14
 Drive at Enclosure 2, Slot 14
 Drive at Enclosure 3, Slot 14
 Drive at Enclosure 4, Slot 14
 Drive at Enclosure 10, Slot 14
 Drive at Enclosure 11, Slot 14
 Drive at Enclosure 12, Slot 14
 Drive at Enclosure 13, Slot 14
 Drive at Enclosure 14, Slot 14

STANDARD LOGICAL DRIVES-----

SUMMARY

Number of standard logical drives: 14
 See other Logical Drives sub-tabs for premium feature information.

NAME	STATUS	CAPACITY	RAID	LEVEL	ARRAY
RACK1LUN0	Optimal	334.014 GB	0	1	
RACK1LUN1	Optimal	334.014 GB	0	2	
RACK1LUN10	Optimal	334.014 GB	0	11	
RACK1LUN11	Optimal	334.014 GB	0	12	
RACK1LUN12	Optimal	300.612 GB	5	13	
RACK1LUN13	Optimal	300.612 GB	5	14	
RACK1LUN2	Optimal	334.014 GB	0	3	
RACK1LUN3	Optimal	334.014 GB	0	4	
RACK1LUN4	Optimal	334.014 GB	0	5	
RACK1LUN5	Optimal	334.014 GB	0	6	
RACK1LUN6	Optimal	334.014 GB	0	7	
RACK1LUN7	Optimal	334.014 GB	0	8	
RACK1LUN8	Optimal	334.014 GB	0	9	
RACK1LUN9	Optimal	334.014 GB	0	10	

DETAILS

Logical Drive name: RACK1LUN0
 Logical Drive ID: 60:0a:0b:80:00:13:63:c3:00:00:00:00:42:52:b0:5a
 Subsystem ID (SSID): 0
 Status: Optimal
 Drive type: Fibre Channel
 Enclosure loss protection: No

Preferred owner: Controller in slot A
Current owner: Controller in slot A
Capacity: 334.014 GB
RAID level: 0
Segment size: 64 KB
Modification priority: High
Associated array: 1
Read cache: Enabled
Write cache: Enabled
Write cache without batteries: Disabled
Write cache with mirroring: Disabled
Flush write cache after (in seconds): 10.00
Cache read ahead multiplier: 0
Enable background media scan: Disabled
Media scan with redundancy check: Disabled

Logical Drive name: RACK1LUN1
Logical Drive ID: 60:0a:0b:80:00:12:a4:b8:00:00:00:01:42:52:b0:31
Subsystem ID (SSID): 1
Status: Optimal
Drive type: Fibre Channel
Enclosure loss protection: No
Preferred owner: Controller in slot B
Current owner: Controller in slot B
Capacity: 334.014 GB
RAID level: 0
Segment size: 64 KB
Modification priority: High
Associated array: 2
Read cache: Enabled
Write cache: Enabled
Write cache without batteries: Disabled
Write cache with mirroring: Disabled
Flush write cache after (in seconds): 10.00
Cache read ahead multiplier: 0
Enable background media scan: Disabled
Media scan with redundancy check: Disabled

Logical Drive name: RACK1LUN10
Logical Drive ID: 60:0a:0b:80:00:13:63:c3:00:00:00:0a:42:52:b3:6c
Subsystem ID (SSID): 10
Status: Optimal
Drive type: Fibre Channel
Enclosure loss protection: No
Preferred owner: Controller in slot A
Current owner: Controller in slot A
Capacity: 334.014 GB
RAID level: 0
Segment size: 64 KB
Modification priority: High
Associated array: 11
Read cache: Enabled
Write cache: Enabled
Write cache without batteries: Disabled
Write cache with mirroring: Disabled
Flush write cache after (in seconds): 10.00
Cache read ahead multiplier: 0
Enable background media scan: Disabled
Media scan with redundancy check: Disabled

Logical Drive name: RACK1LUN11
Logical Drive ID: 60:0a:0b:80:00:12:a4:b8:00:00:00:10:42:52:b3:3b
Subsystem ID (SSID): 11
Status: Optimal
Drive type: Fibre Channel
Enclosure loss protection: No
Preferred owner: Controller in slot B
Current owner: Controller in slot B

Capacity: 334.014 GB
RAID level: 0
Segment size: 64 KB
Modification priority: High
Associated array: 12
Read cache: Enabled
Write cache: Enabled
Write cache without batteries: Disabled
Write cache with mirroring: Disabled
Flush write cache after (in seconds): 10.00
Cache read ahead multiplier: 0
Enable background media scan: Disabled
Media scan with redundancy check: Disabled

Logical Drive name: RACK1LUN12
Logical Drive ID: 60:0a:0b:80:00:13:63:c3:00:00:00:0e:42:53:81:76
Subsystem ID (SSID): 12
Status: Optimal
Drive type: Fibre Channel
Enclosure loss protection: Yes
Preferred owner: Controller in slot A
Current owner: Controller in slot A
Capacity: 300.612 GB
RAID level: 5
Segment size: 512 KB
Modification priority: High
Associated array: 13
Read cache: Enabled
Write cache: Enabled
Write cache without batteries: Disabled
Write cache with mirroring: Disabled
Flush write cache after (in seconds): 10.00
Cache read ahead multiplier: 0
Enable background media scan: Disabled
Media scan with redundancy check: Disabled

Logical Drive name: RACK1LUN13
Logical Drive ID: 60:0a:0b:80:00:12:a4:b8:00:00:00:16:42:53:81:6b
Subsystem ID (SSID): 13
Status: Optimal
Drive type: Fibre Channel
Enclosure loss protection: Yes
Preferred owner: Controller in slot B
Current owner: Controller in slot B
Capacity: 300.612 GB
RAID level: 5
Segment size: 512 KB
Modification priority: High
Associated array: 14
Read cache: Enabled
Write cache: Enabled
Write cache without batteries: Disabled
Write cache with mirroring: Disabled
Flush write cache after (in seconds): 10.00
Cache read ahead multiplier: 0
Enable background media scan: Disabled
Media scan with redundancy check: Disabled

Logical Drive name: RACK1LUN2
Logical Drive ID: 60:0a:0b:80:00:13:63:c3:00:00:00:02:42:52:b0:ea
Subsystem ID (SSID): 2
Status: Optimal
Drive type: Fibre Channel
Enclosure loss protection: No
Preferred owner: Controller in slot A
Current owner: Controller in slot A
Capacity: 334.014 GB
RAID level: 0

Segment size: 64 KB
Modification priority: High
Associated array: 3
Read cache: Enabled
Write cache: Enabled
Write cache without batteries: Disabled
Write cache with mirroring: Disabled
Flush write cache after (in seconds): 10.00
Cache read ahead multiplier: 0
Enable background media scan: Disabled
Media scan with redundancy check: Disabled

Logical Drive name: RACK1LUN3
Logical Drive ID: 60:0a:0b:80:00:12:a4:b8:00:00:00:04:42:52:b0:c1
Subsystem ID (SSID): 3
Status: Optimal
Drive type: Fibre Channel
Enclosure loss protection: No
Preferred owner: Controller in slot B
Current owner: Controller in slot B
Capacity: 334.014 GB
RAID level: 0
Segment size: 64 KB
Modification priority: High
Associated array: 4
Read cache: Enabled
Write cache: Enabled
Write cache without batteries: Disabled
Write cache with mirroring: Disabled
Flush write cache after (in seconds): 10.00
Cache read ahead multiplier: 0
Enable background media scan: Disabled
Media scan with redundancy check: Disabled

Logical Drive name: RACK1LUN4
Logical Drive ID: 60:0a:0b:80:00:13:63:c3:00:00:00:04:42:52:b1:92
Subsystem ID (SSID): 4
Status: Optimal
Drive type: Fibre Channel
Enclosure loss protection: No
Preferred owner: Controller in slot A
Current owner: Controller in slot A
Capacity: 334.014 GB
RAID level: 0
Segment size: 64 KB
Modification priority: High
Associated array: 5
Read cache: Enabled
Write cache: Enabled
Write cache without batteries: Disabled
Write cache with mirroring: Disabled
Flush write cache after (in seconds): 10.00
Cache read ahead multiplier: 0
Enable background media scan: Disabled
Media scan with redundancy check: Disabled

Logical Drive name: RACK1LUN5
Logical Drive ID: 60:0a:0b:80:00:12:a4:b8:00:00:00:07:42:52:b1:6d
Subsystem ID (SSID): 5
Status: Optimal
Drive type: Fibre Channel
Enclosure loss protection: No
Preferred owner: Controller in slot B
Current owner: Controller in slot B
Capacity: 334.014 GB
RAID level: 0
Segment size: 64 KB
Modification priority: High

Associated array: 6
Read cache: Enabled
Write cache: Enabled
Write cache without batteries: Disabled
Write cache with mirroring: Disabled
Flush write cache after (in seconds): 10.00
Cache read ahead multiplier: 0
Enable background media scan: Disabled
Media scan with redundancy check: Disabled

Logical Drive name: RACK1LUN6
Logical Drive ID: 60:0a:0b:80:00:13:63:c3:00:00:00:06:42:52:b2:0e
Subsystem ID (SSID): 6
Status: Optimal
Drive type: Fibre Channel
Enclosure loss protection: No
Preferred owner: Controller in slot A
Current owner: Controller in slot A
Capacity: 334.014 GB
RAID level: 0
Segment size: 64 KB
Modification priority: High
Associated array: 7
Read cache: Enabled
Write cache: Enabled
Write cache without batteries: Disabled
Write cache with mirroring: Disabled
Flush write cache after (in seconds): 10.00
Cache read ahead multiplier: 0
Enable background media scan: Disabled
Media scan with redundancy check: Disabled

Logical Drive name: RACK1LUN7
Logical Drive ID: 60:0a:0b:80:00:12:a4:b8:00:00:00:0a:42:52:b1:d5
Subsystem ID (SSID): 7
Status: Optimal
Drive type: Fibre Channel
Enclosure loss protection: No
Preferred owner: Controller in slot B
Current owner: Controller in slot B
Capacity: 334.014 GB
RAID level: 0
Segment size: 64 KB
Modification priority: High
Associated array: 8
Read cache: Enabled
Write cache: Enabled
Write cache without batteries: Disabled
Write cache with mirroring: Disabled
Flush write cache after (in seconds): 10.00
Cache read ahead multiplier: 0
Enable background media scan: Disabled
Media scan with redundancy check: Disabled

Logical Drive name: RACK1LUN8
Logical Drive ID: 60:0a:0b:80:00:13:63:c3:00:00:00:08:42:52:b2:80
Subsystem ID (SSID): 8
Status: Optimal
Drive type: Fibre Channel
Enclosure loss protection: No
Preferred owner: Controller in slot A
Current owner: Controller in slot A
Capacity: 334.014 GB
RAID level: 0
Segment size: 64 KB
Modification priority: High
Associated array: 9
Read cache: Enabled

Write cache: Enabled
 Write cache without batteries: Disabled
 Write cache with mirroring: Disabled
 Flush write cache after (in seconds): 10.00
 Cache read ahead multiplier: 0
 Enable background media scan: Disabled
 Media scan with redundancy check: Disabled

Logical Drive name: RACK1LUN9
 Logical Drive ID: 60:0a:0b:80:00:12:a4:b8:00:00:00:0d:42:52:b2:51
 Subsystem ID (SSID): 9
 Status: Optimal
 Drive type: Fibre Channel
 Enclosure loss protection: No
 Preferred owner: Controller in slot B
 Current owner: Controller in slot B
 Capacity: 334.014 GB
 RAID level: 0
 Segment size: 64 KB
 Modification priority: High
 Associated array: 10
 Read cache: Enabled
 Write cache: Enabled
 Write cache without batteries: Disabled
 Write cache with mirroring: Disabled
 Flush write cache after (in seconds): 10.00
 Cache read ahead multiplier: 0
 Enable background media scan: Disabled
 Media scan with redundancy check: Disabled

MISSING LOGICAL DRIVES-----
 Number of missing logical drives: 0

DRIVES-----

SUMMARY
 Number of drives: 140
 Supported drive types: Fibre (140)

BASIC:

TRAY, SLOT	STATUS	CAPACITY	CURRENT DATA RATE
PRODUCT ID	FIRMWARE VERSION		
0, 1	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
0, 2	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
0, 3	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
0, 4	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
0, 5	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
0, 6	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
0, 7	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
0, 8	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
0, 9	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
0, 10	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
0, 11	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
0, 12	Optimal	33.902 GB 2 Gbps	ST336753FC F B954

0, 13	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
0, 14	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
1, 1	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
1, 2	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
1, 3	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
1, 4	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
1, 5	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
1, 6	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
1, 7	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
1, 8	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
1, 9	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
1, 10	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
1, 11	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
1, 12	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
1, 13	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
1, 14	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
2, 1	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
2, 2	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
2, 3	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
2, 4	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
2, 5	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
2, 6	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
2, 7	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
2, 8	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
2, 9	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
2, 10	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
2, 11	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
2, 12	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
2, 13	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
2, 14	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
3, 1	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
3, 2	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
3, 3	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
3, 4	Optimal	33.902 GB 2 Gbps	ST336753FC F B954

13, 3	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954	0, 13	2	1
13, 4	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954	0, 14	1	2
13, 5	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954	1, 1	2	1
13, 6	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954	1, 2	1	2
13, 7	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954	1, 3	2	1
13, 8	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954	1, 4	1	2
13, 9	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954	1, 5	2	1
13, 10	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954	1, 6	1	2
13, 11	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954	1, 7	2	1
13, 12	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954	1, 8	1	2
13, 13	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954	1, 9	2	1
13, 14	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954	1, 10	1	2
14, 1	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954	1, 11	2	1
14, 2	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954	1, 12	1	2
14, 3	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954	1, 13	2	1
14, 4	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954	1, 14	1	2
14, 5	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954	2, 1	2	1
14, 6	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954	2, 2	1	2
14, 7	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954	2, 3	2	1
14, 8	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954	2, 4	1	2
14, 9	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954	2, 5	2	1
14, 10	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954	2, 6	1	2
14, 11	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954	2, 7	2	1
14, 12	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954	2, 8	1	2
14, 13	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954	2, 9	2	1
14, 14	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954	2, 10	1	2
							2, 11	2	1
							2, 12	1	2
							2, 13	2	1
							2, 14	1	2
							3, 1	2	1
							3, 2	1	2
							3, 3	2	1
							3, 4	1	2
							3, 5	2	1
							3, 6	1	2
							3, 7	2	1
							3, 8	1	2
							3, 9	2	1
							3, 10	1	2
							3, 11	2	1
							3, 12	1	2
							3, 13	2	1
							3, 14	1	2
							4, 1	2	1
							4, 2	1	2
							4, 3	2	1
							4, 4	1	2
							4, 5	2	1
							4, 6	1	2
							4, 7	2	1
							4, 8	1	2
							4, 9	2	1
							4, 10	1	2
							4, 11	2	1
							4, 12	1	2
							4, 13	2	1
							4, 14	1	2
							10, 1	4	3
							10, 2	3	4
							10, 3	4	3
							10, 4	3	4
							10, 5	4	3
							10, 6	3	4
							10, 7	4	3
							10, 8	3	4
							10, 9	4	3
							10, 10	3	4

DRIVE CHANNELS:

TRAY, SLOT PREFERRED CHANNEL REDUNDANT CHANNEL

0, 1	2	1
0, 2	1	2
0, 3	2	1
0, 4	1	2
0, 5	2	1
0, 6	1	2
0, 7	2	1
0, 8	1	2
0, 9	2	1
0, 10	1	2
0, 11	2	1
0, 12	1	2

10, 11	4	3
10, 12	3	4
10, 13	4	3
10, 14	3	4
11, 1	4	3
11, 2	3	4
11, 3	4	3
11, 4	3	4
11, 5	4	3
11, 6	3	4
11, 7	4	3
11, 8	3	4
11, 9	4	3
11, 10	3	4
11, 11	4	3
11, 12	3	4
11, 13	4	3
11, 14	3	4
12, 1	4	3
12, 2	3	4
12, 3	4	3
12, 4	3	4
12, 5	4	3
12, 6	3	4
12, 7	4	3
12, 8	3	4
12, 9	4	3
12, 10	3	4
12, 11	4	3
12, 12	3	4
12, 13	4	3
12, 14	3	4
13, 1	4	3
13, 2	3	4
13, 3	4	3
13, 4	3	4
13, 5	4	3
13, 6	3	4
13, 7	4	3
13, 8	3	4
13, 9	4	3
13, 10	3	4
13, 11	4	3
13, 12	3	4
13, 13	4	3
13, 14	3	4
14, 1	4	3
14, 2	3	4
14, 3	4	3
14, 4	3	4
14, 5	4	3
14, 6	3	4
14, 7	4	3
14, 8	3	4
14, 9	4	3
14, 10	3	4
14, 11	4	3
14, 12	3	4
14, 13	4	3
14, 14	3	4

HOT SPARE COVERAGE:

The following arrays are not protected: 13, 7, 2, 8, 12, 3, 14, 5, 4, 10, 11, 6, 1, 9

Total hot spare drives: 0

Standby: 0

In use: 0

DETAILS

Drive at Enclosure 0, Slot 1

Drive port: 1, Channel: 2, ID: 0/0xEF
 Drive port: 2, Channel: 1, ID: 0/0xEF
 Drive path redundancy: OK
 Status: Optimal
 Raw capacity: 33.902 GB
 Usable capacity: 33.402 GB
 Current data rate: 2 Gbps
 Product ID: ST336753FC F
 Firmware version: B954
 Serial number: 3HX0HRQQ00007340X5HR
 Vendor: IBM-ESXS
 Date of manufacture: April 5, 2003
 World-wide name: 20:00:00:04:cf:f9:e8:f8
 Drive type: Fibre Channel
 Speed: 15015 RPM
 Mode: Assigned
 Associated array: 1

Drive at Enclosure 0, Slot 2

Drive port: 1, Channel: 1, ID: 1/0xE8
 Drive port: 2, Channel: 2, ID: 1/0xE8
 Drive path redundancy: OK
 Status: Optimal
 Raw capacity: 33.902 GB
 Usable capacity: 33.402 GB
 Current data rate: 2 Gbps
 Product ID: ST336753FC F
 Firmware version: B954
 Serial number: 3HX0HMT500007339TYJJ
 Vendor: IBM-ESXS
 Date of manufacture: April 6, 2003
 World-wide name: 20:00:00:04:cf:ff:03:36
 Drive type: Fibre Channel
 Speed: 15015 RPM
 Mode: Assigned
 Associated array: 2

Drive at Enclosure 0, Slot 3

Drive port: 1, Channel: 2, ID: 2/0xE4
 Drive port: 2, Channel: 1, ID: 2/0xE4
 Drive path redundancy: OK
 Status: Optimal
 Raw capacity: 33.902 GB
 Usable capacity: 33.402 GB
 Current data rate: 2 Gbps
 Product ID: ST336753FC F
 Firmware version: B954
 Serial number: 3HX0K785000073405FTK
 Vendor: IBM-ESXS
 Date of manufacture: April 5, 2003
 World-wide name: 20:00:00:04:cf:f9:e4:c0
 Drive type: Fibre Channel
 Speed: 15015 RPM
 Mode: Assigned
 Associated array: 3

Drive at Enclosure 0, Slot 4

Drive port: 1, Channel: 1, ID: 3/0xE2
 Drive port: 2, Channel: 2, ID: 3/0xE2
 Drive path redundancy: OK
 Status: Optimal
 Raw capacity: 33.902 GB
 Usable capacity: 33.402 GB
 Current data rate: 2 Gbps
 Product ID: ST336753FC F

Firmware version: B954
Serial number: 3HX0K2T100007339SNY4
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:02:b5
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 4

Drive at Enclosure 0, Slot 5

Drive port: 1, Channel: 2, ID: 4/0xE1
Drive port: 2, Channel: 1, ID: 4/0xE1
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0JPPJ000073392E3P
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:02:a8
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 5

Drive at Enclosure 0, Slot 6

Drive port: 1, Channel: 1, ID: 5/0xE0
Drive port: 2, Channel: 2, ID: 5/0xE0
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0KCTZ000073405G1W
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:0c:fb
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 6

Drive at Enclosure 0, Slot 7

Drive port: 1, Channel: 2, ID: 6/0xDC
Drive port: 2, Channel: 1, ID: 6/0xDC
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K0S100007339RGS4
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:08:1f
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 7

Drive at Enclosure 0, Slot 8

Drive port: 1, Channel: 1, ID: 7/0xDA
Drive port: 2, Channel: 2, ID: 7/0xDA
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0KCLQ00007340WVPE
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:0d:53
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 8

Drive at Enclosure 0, Slot 9

Drive port: 1, Channel: 2, ID: 80/0x55
Drive port: 2, Channel: 1, ID: 80/0x55
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0JY300000734058C3
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:0d:87
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 9

Drive at Enclosure 0, Slot 10

Drive port: 1, Channel: 1, ID: 96/0x3A
Drive port: 2, Channel: 2, ID: 96/0x3A
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K5QP00007340X5FR
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:0d:30
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 10

Drive at Enclosure 0, Slot 11

Drive port: 1, Channel: 2, ID: 64/0x72
Drive port: 2, Channel: 1, ID: 64/0x72
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0JXRJ000073409KRS
Vendor: IBM-ESXS

Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:0d:11
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 11

Drive at Enclosure 0, Slot 12

Drive port: 1, Channel: 1, ID: 72/0x67
Drive port: 2, Channel: 2, ID: 72/0x67
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0JYAP0000734058K6
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:0d:57
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 12

Drive at Enclosure 0, Slot 13

Drive port: 1, Channel: 2, ID: 88/0x4B
Drive port: 2, Channel: 1, ID: 88/0x4B
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0JZSB0000734058CF
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:0d:7e
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 13

Drive at Enclosure 0, Slot 14

Drive port: 1, Channel: 1, ID: 104/0x2E
Drive port: 2, Channel: 2, ID: 104/0x2E
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0KBL000073410HMK
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:12:93
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 14

Drive at Enclosure 1, Slot 1

Drive port: 1, Channel: 2, ID: 8/0xD9
Drive port: 2, Channel: 1, ID: 8/0xD9
Drive path redundancy: OK

Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HPEX000073392DT8
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:f9:fb:1f
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 1

Drive at Enclosure 1, Slot 2

Drive port: 1, Channel: 1, ID: 9/0xD6
Drive port: 2, Channel: 2, ID: 9/0xD6
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HSZN000073405G6Z
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:f9:e5:c4
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 2

Drive at Enclosure 1, Slot 3

Drive port: 1, Channel: 2, ID: 10/0xD5
Drive port: 2, Channel: 1, ID: 10/0xD5
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0KCG6000073410J6T
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:12:7b
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 3

Drive at Enclosure 1, Slot 4

Drive port: 1, Channel: 1, ID: 11/0xD4
Drive port: 2, Channel: 2, ID: 11/0xD4
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K7Q9000073405G31
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:ed:3f
Drive type: Fibre Channel

Speed: 15015 RPM
Mode: Assigned
Associated array: 4

Drive at Enclosure 1, Slot 5

Drive port: 1, Channel: 2, ID: 12/0xD3
Drive port: 2, Channel: 1, ID: 12/0xD3
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HLY500007340WVKB
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:ed:74
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 5

Drive at Enclosure 1, Slot 6

Drive port: 1, Channel: 1, ID: 13/0xD2
Drive port: 2, Channel: 2, ID: 13/0xD2
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K2B2000073405A6W
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:0e:37
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 6

Drive at Enclosure 1, Slot 7

Drive port: 1, Channel: 2, ID: 14/0xD1
Drive port: 2, Channel: 1, ID: 14/0xD1
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0JZM1000073405A8Q
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:0a:25
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 7

Drive at Enclosure 1, Slot 8

Drive port: 1, Channel: 1, ID: 15/0xCE
Drive port: 2, Channel: 2, ID: 15/0xCE
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB

Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K88C000073392DZ6
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e3:4a
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 8

Drive at Enclosure 1, Slot 9

Drive port: 1, Channel: 2, ID: 81/0x54
Drive port: 2, Channel: 1, ID: 81/0x54
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HT6100007339SNTB
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:02:46
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 9

Drive at Enclosure 1, Slot 10

Drive port: 1, Channel: 1, ID: 97/0x39
Drive port: 2, Channel: 2, ID: 97/0x39
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K6VS00007340WVLE
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:ed:3a
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 10

Drive at Enclosure 1, Slot 11

Drive port: 1, Channel: 2, ID: 65/0x71
Drive port: 2, Channel: 1, ID: 65/0x71
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0KCPW0000734058CG
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:0d:25
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 11

Drive at Enclosure 1, Slot 12
Drive port: 1, Channel: 1, ID: 73/0x66
Drive port: 2, Channel: 2, ID: 73/0x66
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0JZDG000073405A9Z
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:0a:32
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 12

Drive at Enclosure 1, Slot 13
Drive port: 1, Channel: 2, ID: 89/0x4A
Drive port: 2, Channel: 1, ID: 89/0x4A
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0KCL5000073405FGJ
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:0d:5d
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 13

Drive at Enclosure 1, Slot 14
Drive port: 1, Channel: 1, ID: 105/0x2D
Drive port: 2, Channel: 2, ID: 105/0x2D
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HQ5P000073393FLA
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:04:e2
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 14

Drive at Enclosure 2, Slot 1
Drive port: 1, Channel: 2, ID: 16/0xCD
Drive port: 2, Channel: 1, ID: 16/0xCD
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954

Serial number: 3HX0JZD300007339PQPY
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:0a:13
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 1

Drive at Enclosure 2, Slot 2
Drive port: 1, Channel: 1, ID: 17/0xCC
Drive port: 2, Channel: 2, ID: 17/0xCC
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0JZK900007339TYQE
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:0a:15
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 2

Drive at Enclosure 2, Slot 3
Drive port: 1, Channel: 2, ID: 18/0xCB
Drive port: 2, Channel: 1, ID: 18/0xCB
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K7SK000073405G3V
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:ed:4a
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 3

Drive at Enclosure 2, Slot 4
Drive port: 1, Channel: 1, ID: 19/0xCA
Drive port: 2, Channel: 2, ID: 19/0xCA
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0JYAL00007340X57P
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:0d:4d
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 4

Drive at Enclosure 2, Slot 5
Drive port: 1, Channel: 2, ID: 20/0xC9

Drive port: 2, Channel: 1, ID: 20/0xC9
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0KCSR000073409K3X
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:0d:09
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 5

Drive at Enclosure 2, Slot 6

Drive port: 1, Channel: 1, ID: 21/0xC7
Drive port: 2, Channel: 2, ID: 21/0xC7
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0KD5T000073410HMB
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:12:90
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 6

Drive at Enclosure 2, Slot 7

Drive port: 1, Channel: 2, ID: 22/0xC6
Drive port: 2, Channel: 1, ID: 22/0xC6
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HT1E000073392E0T
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e2:cd
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 7

Drive at Enclosure 2, Slot 8

Drive port: 1, Channel: 1, ID: 23/0xC5
Drive port: 2, Channel: 2, ID: 23/0xC5
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K79G00007340X4RD
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003

World-wide name: 20:00:00:04:cf:f9:e5:c6
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 8

Drive at Enclosure 2, Slot 9

Drive port: 1, Channel: 2, ID: 82/0x53
Drive port: 2, Channel: 1, ID: 82/0x53
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K2JQ00007339TYUG
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:02:6b
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 9

Drive at Enclosure 2, Slot 10

Drive port: 1, Channel: 1, ID: 98/0x36
Drive port: 2, Channel: 2, ID: 98/0x36
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K20V00007339LXKL
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:02:ce
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 10

Drive at Enclosure 2, Slot 11

Drive port: 1, Channel: 2, ID: 66/0x6E
Drive port: 2, Channel: 1, ID: 66/0x6E
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0JRQ4000073409KFX
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:04:e3
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 11

Drive at Enclosure 2, Slot 12

Drive port: 1, Channel: 1, ID: 74/0x65
Drive port: 2, Channel: 2, ID: 74/0x65
Drive path redundancy: OK
Status: Optimal

Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0JQV000007339EGAZ
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:03:af
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 12

Drive at Enclosure 2, Slot 13
Drive port: 1, Channel: 2, ID: 90/0x49
Drive port: 2, Channel: 1, ID: 90/0x49
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0JPFQ00007340X4Z5
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:fb:a9
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 13

Drive at Enclosure 2, Slot 14
Drive port: 1, Channel: 1, ID: 106/0x2C
Drive port: 2, Channel: 2, ID: 106/0x2C
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K16800007339SP6X
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:03:a9
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 14

Drive at Enclosure 3, Slot 1
Drive port: 1, Channel: 2, ID: 24/0xC3
Drive port: 2, Channel: 1, ID: 24/0xC3
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K88H000073405G7B
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e5:2d
Drive type: Fibre Channel
Speed: 15015 RPM

Mode: Assigned
Associated array: 1

Drive at Enclosure 3, Slot 2
Drive port: 1, Channel: 1, ID: 25/0xBC
Drive port: 2, Channel: 2, ID: 25/0xBC
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0JT7N000073405G4K
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:07:e3
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 2

Drive at Enclosure 3, Slot 3
Drive port: 1, Channel: 2, ID: 26/0xBA
Drive port: 2, Channel: 1, ID: 26/0xBA
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K7V4000073405G32
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:ed:00
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 3

Drive at Enclosure 3, Slot 4
Drive port: 1, Channel: 1, ID: 27/0xB9
Drive port: 2, Channel: 2, ID: 27/0xB9
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HQGM000073409K32
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:fc:03
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 4

Drive at Enclosure 3, Slot 5
Drive port: 1, Channel: 2, ID: 28/0xB6
Drive port: 2, Channel: 1, ID: 28/0xB6
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps

Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0JLGC0000734059ND
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e6:ea
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 5

Drive at Enclosure 3, Slot 6

Drive port: 1, Channel: 1, ID: 29/0xB5
Drive port: 2, Channel: 2, ID: 29/0xB5
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K6NK000073405A8D
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:07:d2
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 6

Drive at Enclosure 3, Slot 7

Drive port: 1, Channel: 2, ID: 30/0xB4
Drive port: 2, Channel: 1, ID: 30/0xB4
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HRNN00007340X5JM
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e8:ac
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 7

Drive at Enclosure 3, Slot 8

Drive port: 1, Channel: 1, ID: 31/0xB3
Drive port: 2, Channel: 2, ID: 31/0xB3
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HRG0000734059NM
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e7:0e
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 8

Drive at Enclosure 3, Slot 9

Drive port: 1, Channel: 2, ID: 83/0x52
Drive port: 2, Channel: 1, ID: 83/0x52
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HT8B0000734059NU
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e7:43
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 9

Drive at Enclosure 3, Slot 10

Drive port: 1, Channel: 1, ID: 99/0x35
Drive port: 2, Channel: 2, ID: 99/0x35
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HT0A000073392DUW
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e3:b4
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 10

Drive at Enclosure 3, Slot 11

Drive port: 1, Channel: 2, ID: 67/0x6D
Drive port: 2, Channel: 1, ID: 67/0x6D
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K09M000073405A81
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:07:e1
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 11

Drive at Enclosure 3, Slot 12

Drive port: 1, Channel: 1, ID: 75/0x63
Drive port: 2, Channel: 2, ID: 75/0x63
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0KD1R000073405FHZ

Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:11:c9
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 12

Drive at Enclosure 3, Slot 13

Drive port: 1, Channel: 2, ID: 91/0x47
Drive port: 2, Channel: 1, ID: 91/0x47
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HLLP00007340GUVB
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:13:49
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 13

Drive at Enclosure 3, Slot 14

Drive port: 1, Channel: 1, ID: 107/0x2B
Drive port: 2, Channel: 2, ID: 107/0x2B
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K7F100007339TYU
Vendor: IBM-ESXS
Date of manufacture: April 4, 2003
World-wide name: 20:00:00:04:cf:f9:e6:33
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 14

Drive at Enclosure 4, Slot 1

Drive port: 1, Channel: 2, ID: 32/0xB2
Drive port: 2, Channel: 1, ID: 32/0xB2
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HRJP00007339TYNP
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:02:a7
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 1

Drive at Enclosure 4, Slot 2

Drive port: 1, Channel: 1, ID: 33/0xB1
Drive port: 2, Channel: 2, ID: 33/0xB1

Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K0TW000073393FXQ
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:0e:7b
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 2

Drive at Enclosure 4, Slot 3

Drive port: 1, Channel: 2, ID: 34/0xAE
Drive port: 2, Channel: 1, ID: 34/0xAE
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HT05000073405FS1
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e6:2d
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 3

Drive at Enclosure 4, Slot 4

Drive port: 1, Channel: 1, ID: 35/0xAD
Drive port: 2, Channel: 2, ID: 35/0xAD
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HM3A000073409KM6
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:dc:87
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 4

Drive at Enclosure 4, Slot 5

Drive port: 1, Channel: 2, ID: 36/0xAC
Drive port: 2, Channel: 1, ID: 36/0xAC
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HKB9000073393FPW
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:f9:fb:18

Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 5

Drive at Enclosure 4, Slot 6

Drive port: 1, Channel: 1, ID: 37/0xAB
Drive port: 2, Channel: 2, ID: 37/0xAB
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0J900000073392DWT
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e3:b0
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 6

Drive at Enclosure 4, Slot 7

Drive port: 1, Channel: 2, ID: 38/0xAA
Drive port: 2, Channel: 1, ID: 38/0xAA
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HT5P00007340X567
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e8:9c
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 7

Drive at Enclosure 4, Slot 8

Drive port: 1, Channel: 1, ID: 39/0xA9
Drive port: 2, Channel: 2, ID: 39/0xA9
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HT4J000073405FRV
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e6:09
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 8

Drive at Enclosure 4, Slot 9

Drive port: 1, Channel: 2, ID: 84/0x51
Drive port: 2, Channel: 1, ID: 84/0x51
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB

Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K78C00007339RGJL
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e5:1e
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 9

Drive at Enclosure 4, Slot 10

Drive port: 1, Channel: 1, ID: 100/0x34
Drive port: 2, Channel: 2, ID: 100/0x34
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336732FC F
Firmware version: B947
Serial number: 3ET0YGRW000072443UZB
Vendor: IBM-ESXS
Date of manufacture: May 15, 2002
World-wide name: 20:00:00:04:cf:5f:53:14
Drive type: Fibre Channel
Speed: 14996 RPM
Mode: Assigned
Associated array: 10

Drive at Enclosure 4, Slot 11

Drive port: 1, Channel: 2, ID: 68/0x6C
Drive port: 2, Channel: 1, ID: 68/0x6C
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0H7EB00007339N0TD
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:02:82
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 11

Drive at Enclosure 4, Slot 12

Drive port: 1, Channel: 1, ID: 76/0x5C
Drive port: 2, Channel: 2, ID: 76/0x5C
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HZPE000073411NJR
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:13:b5
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned

Associated array: 12

Drive at Enclosure 4, Slot 13

Drive port: 1, Channel: 2, ID: 92/0x46
Drive port: 2, Channel: 1, ID: 92/0x46
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HT630000734058FQ
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e5:ff
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 13

Drive at Enclosure 4, Slot 14

Drive port: 1, Channel: 1, ID: 108/0x2A
Drive port: 2, Channel: 2, ID: 108/0x2A
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0KD0C00007340WW1S
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:11:cd
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 14

Drive at Enclosure 10, Slot 1

Drive port: 1, Channel: 4, ID: 0/0xEF
Drive port: 2, Channel: 3, ID: 0/0xEF
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HT4F0000734058KQ
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e6:fc
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 1

Drive at Enclosure 10, Slot 2

Drive port: 1, Channel: 3, ID: 1/0xE8
Drive port: 2, Channel: 4, ID: 1/0xE8
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F

Firmware version: B954
Serial number: 3HX0KBTQ000073405FUP
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:12:ce
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 2

Drive at Enclosure 10, Slot 3

Drive port: 1, Channel: 4, ID: 2/0xE4
Drive port: 2, Channel: 3, ID: 2/0xE4
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0H3RG00007339TYTY
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:02:74
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 3

Drive at Enclosure 10, Slot 4

Drive port: 1, Channel: 3, ID: 3/0xE2
Drive port: 2, Channel: 4, ID: 3/0xE2
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0KBLV00007340GUF4
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:12:db
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 4

Drive at Enclosure 10, Slot 5

Drive port: 1, Channel: 4, ID: 4/0xE1
Drive port: 2, Channel: 3, ID: 4/0xE1
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K45900007339TYRV
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:02:55
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 5

Drive at Enclosure 10, Slot 6

Drive port: 1, Channel: 3, ID: 5/0xE0
Drive port: 2, Channel: 4, ID: 5/0xE0
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HRTV000073405FRL
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:13:09
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 6

Drive at Enclosure 10, Slot 7

Drive port: 1, Channel: 4, ID: 6/0xDC
Drive port: 2, Channel: 3, ID: 6/0xDC
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX35T7Q00007518DDR3
Vendor: IBM-ESXS
Date of manufacture: November 7, 2004
World-wide name: 20:00:00:11:c6:28:7b:19
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 7

Drive at Enclosure 10, Slot 8

Drive port: 1, Channel: 3, ID: 7/0xDA
Drive port: 2, Channel: 4, ID: 7/0xDA
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K8F5000073405FS5
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:12:a5
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 8

Drive at Enclosure 10, Slot 9

Drive port: 1, Channel: 4, ID: 80/0x55
Drive port: 2, Channel: 3, ID: 80/0x55
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0KBJ5000073405FVE
Vendor: IBM-ESXS

Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:12:66
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 9

Drive at Enclosure 10, Slot 10

Drive port: 1, Channel: 3, ID: 96/0x3A
Drive port: 2, Channel: 4, ID: 96/0x3A
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HMOZ00007339TY8W
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:eb:b1
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 10

Drive at Enclosure 10, Slot 11

Drive port: 1, Channel: 4, ID: 64/0x72
Drive port: 2, Channel: 3, ID: 64/0x72
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0KBSW000073405FS4
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:12:7d
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 11

Drive at Enclosure 10, Slot 12

Drive port: 1, Channel: 3, ID: 72/0x67
Drive port: 2, Channel: 4, ID: 72/0x67
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K07Q00007339TYZU
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:02:3f
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 12

Drive at Enclosure 10, Slot 13

Drive port: 1, Channel: 4, ID: 88/0x4B
Drive port: 2, Channel: 3, ID: 88/0x4B
Drive path redundancy: OK

Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HRLK00007340X5AX
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e8:77
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 13

Drive at Enclosure 10, Slot 14
Drive port: 1, Channel: 3, ID: 104/0x2E
Drive port: 2, Channel: 4, ID: 104/0x2E
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K1BN00007339TYUN
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:02:5b
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 14

Drive at Enclosure 11, Slot 1
Drive port: 1, Channel: 4, ID: 8/0xD9
Drive port: 2, Channel: 3, ID: 8/0xD9
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HSYC00007339RGCC
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e1:f6
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 1

Drive at Enclosure 11, Slot 2
Drive port: 1, Channel: 3, ID: 9/0xD6
Drive port: 2, Channel: 4, ID: 9/0xD6
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HSR700007339RGPH
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e1:96
Drive type: Fibre Channel

Speed: 15015 RPM
Mode: Assigned
Associated array: 2

Drive at Enclosure 11, Slot 3
Drive port: 1, Channel: 4, ID: 10/0xD5
Drive port: 2, Channel: 3, ID: 10/0xD5
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HM6B000073409KGT
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:db:d3
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 3

Drive at Enclosure 11, Slot 4
Drive port: 1, Channel: 3, ID: 11/0xD4
Drive port: 2, Channel: 4, ID: 11/0xD4
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K85E00007339RGMP
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e1:d8
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 4

Drive at Enclosure 11, Slot 5
Drive port: 1, Channel: 4, ID: 12/0xD3
Drive port: 2, Channel: 3, ID: 12/0xD3
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K78K00007340X4ZH
Vendor: IBM-ESXS
Date of manufacture: April 4, 2003
World-wide name: 20:00:00:04:cf:f9:e5:8c
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 5

Drive at Enclosure 11, Slot 6
Drive port: 1, Channel: 3, ID: 13/0xD2
Drive port: 2, Channel: 4, ID: 13/0xD2
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB

Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K86E00007339RGQJ
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e1:b4
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 6

Drive at Enclosure 11, Slot 7
Drive port: 1, Channel: 4, ID: 14/0xD1
Drive port: 2, Channel: 3, ID: 14/0xD1
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HSYT00007339RGJU
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e1:cb
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 7

Drive at Enclosure 11, Slot 8
Drive port: 1, Channel: 3, ID: 15/0xCE
Drive port: 2, Channel: 4, ID: 15/0xCE
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HSXF000073392DTN
Vendor: IBM-ESXS
Date of manufacture: April 4, 2003
World-wide name: 20:00:00:04:cf:f9:e8:61
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 8

Drive at Enclosure 11, Slot 9
Drive port: 1, Channel: 4, ID: 81/0x54
Drive port: 2, Channel: 3, ID: 81/0x54
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HT8600007340X5AS
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e7:5a
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 9

Drive at Enclosure 11, Slot 10
Drive port: 1, Channel: 3, ID: 97/0x39
Drive port: 2, Channel: 4, ID: 97/0x39
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HT1D00007339TYN5
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e8:80
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 10

Drive at Enclosure 11, Slot 11
Drive port: 1, Channel: 4, ID: 65/0x71
Drive port: 2, Channel: 3, ID: 65/0x71
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HRPJ0000734058G1
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e7:bf
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 11

Drive at Enclosure 11, Slot 12
Drive port: 1, Channel: 3, ID: 73/0x66
Drive port: 2, Channel: 4, ID: 73/0x66
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HT490000734058CT
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e7:ab
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 12

Drive at Enclosure 11, Slot 13
Drive port: 1, Channel: 4, ID: 89/0x4A
Drive port: 2, Channel: 3, ID: 89/0x4A
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954

Serial number: 3HX0HNRH00007339TYMJ
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e8:27
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 13

Drive at Enclosure 11, Slot 14

Drive port: 1, Channel: 3, ID: 105/0x2D
Drive port: 2, Channel: 4, ID: 105/0x2D
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HRGX00007340WW07
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:03:ff
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 14

Drive at Enclosure 12, Slot 1

Drive port: 1, Channel: 4, ID: 16/0xCD
Drive port: 2, Channel: 3, ID: 16/0xCD
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0JE4K000073392D9J
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:f9:fc:19
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 1

Drive at Enclosure 12, Slot 2

Drive port: 1, Channel: 3, ID: 17/0xCC
Drive port: 2, Channel: 4, ID: 17/0xCC
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0JZJV00007339GMRQ
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:08:40
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 2

Drive at Enclosure 12, Slot 3

Drive port: 1, Channel: 4, ID: 18/0xCB

Drive port: 2, Channel: 3, ID: 18/0xCB
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K0SH00007339TYS3
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:08:42
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 3

Drive at Enclosure 12, Slot 4

Drive port: 1, Channel: 3, ID: 19/0xCA
Drive port: 2, Channel: 4, ID: 19/0xCA
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0JKTE000073405G6W
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e5:73
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 4

Drive at Enclosure 12, Slot 5

Drive port: 1, Channel: 4, ID: 20/0xC9
Drive port: 2, Channel: 3, ID: 20/0xC9
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K1BT00007339TYS5
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:02:6e
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 5

Drive at Enclosure 12, Slot 6

Drive port: 1, Channel: 3, ID: 21/0xC7
Drive port: 2, Channel: 4, ID: 21/0xC7
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HRN3000073392E0B
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003

World-wide name: 20:00:00:04:cf:f9:ea:6f
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 6

Drive at Enclosure 12, Slot 7

Drive port: 1, Channel: 4, ID: 22/0xC6
Drive port: 2, Channel: 3, ID: 22/0xC6
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K5TY00007339TYUV
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:02:72
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 7

Drive at Enclosure 12, Slot 8

Drive port: 1, Channel: 3, ID: 23/0xC5
Drive port: 2, Channel: 4, ID: 23/0xC5
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0TQBD00007346FB5T
Vendor: IBM-ESXS
Date of manufacture: May 17, 2003
World-wide name: 20:00:00:0c:50:20:a1:b4
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 8

Drive at Enclosure 12, Slot 9

Drive port: 1, Channel: 4, ID: 82/0x53
Drive port: 2, Channel: 3, ID: 82/0x53
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HRG80000734058JG
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e7:6a
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 9

Drive at Enclosure 12, Slot 10

Drive port: 1, Channel: 3, ID: 98/0x36
Drive port: 2, Channel: 4, ID: 98/0x36
Drive path redundancy: OK
Status: Optimal

Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HRFP00007340X5BG
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e8:ab
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 10

Drive at Enclosure 12, Slot 11

Drive port: 1, Channel: 4, ID: 66/0x6E
Drive port: 2, Channel: 3, ID: 66/0x6E
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0J1G300007339GMRV
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:f9:fb:29
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 11

Drive at Enclosure 12, Slot 12

Drive port: 1, Channel: 3, ID: 74/0x65
Drive port: 2, Channel: 4, ID: 74/0x65
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HT2M0000734058JX
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e6:c3
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 12

Drive at Enclosure 12, Slot 13

Drive port: 1, Channel: 4, ID: 90/0x49
Drive port: 2, Channel: 3, ID: 90/0x49
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K2HR00007340X4ZP
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:03:62
Drive type: Fibre Channel
Speed: 15015 RPM

Mode: Assigned
Associated array: 13

Drive at Enclosure 12, Slot 14

Drive port: 1, Channel: 3, ID: 106/0x2C
Drive port: 2, Channel: 4, ID: 106/0x2C
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0JRTS00007339TYCA
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e5:d9
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 14

Drive at Enclosure 13, Slot 1

Drive port: 1, Channel: 4, ID: 24/0xC3
Drive port: 2, Channel: 3, ID: 24/0xC3
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HMZR00007339SP49
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:02:b0
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 1

Drive at Enclosure 13, Slot 2

Drive port: 1, Channel: 3, ID: 25/0xBC
Drive port: 2, Channel: 4, ID: 25/0xBC
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0JSAQ000073392DCV
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:02:f6
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 2

Drive at Enclosure 13, Slot 3

Drive port: 1, Channel: 4, ID: 26/0xBA
Drive port: 2, Channel: 3, ID: 26/0xBA
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps

Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K1MB00007338JX9B
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:03:a7
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 3

Drive at Enclosure 13, Slot 4

Drive port: 1, Channel: 3, ID: 27/0xB9
Drive port: 2, Channel: 4, ID: 27/0xB9
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K2EA00007339RGEB
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:03:47
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 4

Drive at Enclosure 13, Slot 5

Drive port: 1, Channel: 4, ID: 28/0xB6
Drive port: 2, Channel: 3, ID: 28/0xB6
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HH9B00007340WW09
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:03:f8
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 5

Drive at Enclosure 13, Slot 6

Drive port: 1, Channel: 3, ID: 29/0xB5
Drive port: 2, Channel: 4, ID: 29/0xB5
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0JTRB00007339SPDE
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:02:85
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 6

Drive at Enclosure 13, Slot 7
Drive port: 1, Channel: 4, ID: 30/0xB4
Drive port: 2, Channel: 3, ID: 30/0xB4
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K1F400007338K1B3
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:03:12
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 7

Drive at Enclosure 13, Slot 8
Drive port: 1, Channel: 3, ID: 31/0xB3
Drive port: 2, Channel: 4, ID: 31/0xB3
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0JLL200007339RG6H
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:02:7e
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 8

Drive at Enclosure 13, Slot 9
Drive port: 1, Channel: 4, ID: 83/0x52
Drive port: 2, Channel: 3, ID: 83/0x52
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HSRV00007339TYRR
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:05:4c
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 9

Drive at Enclosure 13, Slot 10
Drive port: 1, Channel: 3, ID: 99/0x35
Drive port: 2, Channel: 4, ID: 99/0x35
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HN6D00007339GMHP

Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:02:92
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 10

Drive at Enclosure 13, Slot 11
Drive port: 1, Channel: 4, ID: 67/0x6D
Drive port: 2, Channel: 3, ID: 67/0x6D
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K7SH00007339SPCH
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:02:d4
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 11

Drive at Enclosure 13, Slot 12
Drive port: 1, Channel: 3, ID: 75/0x63
Drive port: 2, Channel: 4, ID: 75/0x63
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K82K00007339FG72
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:02:dc
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 12

Drive at Enclosure 13, Slot 13
Drive port: 1, Channel: 4, ID: 91/0x47
Drive port: 2, Channel: 3, ID: 91/0x47
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K6SP00007340WW2N
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:03:f5
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 13

Drive at Enclosure 13, Slot 14
Drive port: 1, Channel: 3, ID: 107/0x2B
Drive port: 2, Channel: 4, ID: 107/0x2B

Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0H44J00007340WW3U
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:03:bd
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 14

Drive at Enclosure 14, Slot 1
Drive port: 1, Channel: 4, ID: 32/0xB2
Drive port: 2, Channel: 3, ID: 32/0xB2
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0H41N00007339SNQW
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e3:72
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 1

Drive at Enclosure 14, Slot 2
Drive port: 1, Channel: 3, ID: 33/0xB1
Drive port: 2, Channel: 4, ID: 33/0xB1
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K7S3000073405G3R
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:ed:2d
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 2

Drive at Enclosure 14, Slot 3
Drive port: 1, Channel: 4, ID: 34/0xAE
Drive port: 2, Channel: 3, ID: 34/0xAE
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K70P000073405G36
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:ed:2b

Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 3

Drive at Enclosure 14, Slot 4
Drive port: 1, Channel: 3, ID: 35/0xAD
Drive port: 2, Channel: 4, ID: 35/0xAD
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0JLV6000073405A9D
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:07:e2
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 4

Drive at Enclosure 14, Slot 5
Drive port: 1, Channel: 4, ID: 36/0xAC
Drive port: 2, Channel: 3, ID: 36/0xAC
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0JSWG00007338W96D
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:ca
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 5

Drive at Enclosure 14, Slot 6
Drive port: 1, Channel: 3, ID: 37/0xAB
Drive port: 2, Channel: 4, ID: 37/0xAB
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HRHX000073405G54
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e7:f5
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 6

Drive at Enclosure 14, Slot 7
Drive port: 1, Channel: 4, ID: 38/0xAA
Drive port: 2, Channel: 3, ID: 38/0xAA
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB

Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0JYX3000073405A44
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:0d:70
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 7

Drive at Enclosure 14, Slot 8
Drive port: 1, Channel: 3, ID: 39/0xA9
Drive port: 2, Channel: 4, ID: 39/0xA9
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0JPG00007339GMK8
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:02:cb
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 8

Drive at Enclosure 14, Slot 9
Drive port: 1, Channel: 4, ID: 84/0x51
Drive port: 2, Channel: 3, ID: 84/0x51
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HLY600007340X50U
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:db:d1
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 9

Drive at Enclosure 14, Slot 10
Drive port: 1, Channel: 3, ID: 100/0x34
Drive port: 2, Channel: 4, ID: 100/0x34
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0JAA200007340WEWF
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:0e:28
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned

Associated array: 10

Drive at Enclosure 14, Slot 11
Drive port: 1, Channel: 4, ID: 68/0x6C
Drive port: 2, Channel: 3, ID: 68/0x6C
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0KBFX000073410J7B
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:14:69
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 11

Drive at Enclosure 14, Slot 12
Drive port: 1, Channel: 3, ID: 76/0x5C
Drive port: 2, Channel: 4, ID: 76/0x5C
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K2RZ00007339LJ8H
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:02:90
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 12

Drive at Enclosure 14, Slot 13
Drive port: 1, Channel: 4, ID: 92/0x46
Drive port: 2, Channel: 3, ID: 92/0x46
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K6H500007340GTVD
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:f6:00
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 13

Drive at Enclosure 14, Slot 14
Drive port: 1, Channel: 3, ID: 108/0x2A
Drive port: 2, Channel: 4, ID: 108/0x2A
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F

Firmware version: B954
Serial number: 3HX0KBSV00007340GUZK
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:13:3b
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 14

DRIVE CHANNEL INFORMATION - Channel 1

SUMMARY-----

Channel 1 status: Optimal
Controller A link status: Up
Controller B link status: Up

CUMULATIVE ERROR COUNTS-----

Controller A Cumulative Error Counts
Baseline time set: 4/19/05 3:46:55 PM
Sample period (days, hh:mm:ss): 30 days, 16:59:51
Controller detected errors: 0
Drive detected errors: 0
Timeout errors: 0
Link down errors: 2
Total I/O count: 87316
Controller B Cumulative Error Counts
Baseline time set: 4/19/05 3:46:55 PM
Sample period (days, hh:mm:ss): 30 days, 16:59:51
Controller detected errors: 0
Drive detected errors: 0
Timeout errors: 0
Link down errors: 8
Total I/O count: 1204044712

DRIVE CHANNEL INFORMATION - Channel 2

SUMMARY-----

Channel 2 status: Optimal
Controller A link status: Up
Controller B link status: Up

CUMULATIVE ERROR COUNTS-----

Controller A Cumulative Error Counts
Baseline time set: 4/19/05 3:46:57 PM
Sample period (days, hh:mm:ss): 30 days, 16:59:49
Controller detected errors: 0
Drive detected errors: 0
Timeout errors: 0
Link down errors: 2
Total I/O count: 18085033
Controller B Cumulative Error Counts
Baseline time set: 4/19/05 3:46:57 PM
Sample period (days, hh:mm:ss): 30 days, 16:59:49
Controller detected errors: 0
Drive detected errors: 0
Timeout errors: 0
Link down errors: 8
Total I/O count: 3988936

DRIVE CHANNEL INFORMATION - Channel 3

SUMMARY-----

Channel 3 status: Optimal
Controller A link status: Up
Controller B link status: Up

CUMULATIVE ERROR COUNTS-----

Controller A Cumulative Error Counts
Baseline time set: 4/19/05 3:46:57 PM
Sample period (days, hh:mm:ss): 30 days, 16:59:49
Controller detected errors: 0
Drive detected errors: 0
Timeout errors: 0
Link down errors: 2
Total I/O count: 88850

Controller B Cumulative Error Counts
Baseline time set: 4/19/05 3:46:57 PM
Sample period (days, hh:mm:ss): 30 days, 16:59:49
Controller detected errors: 0
Drive detected errors: 0
Timeout errors: 1
Link down errors: 8
Total I/O count: 1195307654

DRIVE CHANNEL INFORMATION - Channel 4

SUMMARY-----

Channel 4 status: Optimal
Controller A link status: Up
Controller B link status: Up

CUMULATIVE ERROR COUNTS-----

Controller A Cumulative Error Counts
Baseline time set: 4/19/05 3:46:57 PM
Sample period (days, hh:mm:ss): 30 days, 16:59:49
Controller detected errors: 0
Drive detected errors: 0
Timeout errors: 0
Link down errors: 2
Total I/O count: 17978945
Controller B Cumulative Error Counts
Baseline time set: 4/19/05 3:46:57 PM
Sample period (days, hh:mm:ss): 30 days, 16:59:49
Controller detected errors: 0
Drive detected errors: 0
Timeout errors: 0
Link down errors: 8
Total I/O count: 4168575

Rack 2

Rack 2 is identical to Rack 1.

Rack 3

PROFILE FOR STORAGE SUBSYSTEM: Rack_3 (5/20/05 8:56:45 AM)

SUMMARY-----

Number of controllers: 2
Number of arrays: 14
Total number of logical drives (includes an access logical drive): 15 of 2048 used
Number of standard logical drives: 14
Number of access logical drives: 1
Number of drives: 140
Supported drive types: Fibre (140)
Total hot spare drives: 0
Standby: 0
In use: 0
Access logical drive: None mapped
Default host type: Windows 2000/Server 2003 Non-Clustered (Host type index 2)
Current configuration
Firmware version: 06.12.03.00

NVSRAM version: N1742F900R912V06
Pending configuration
Staged firmware download supported?: Yes
Firmware version: None
NVSRAM version: None
Transferred on: None
NVSRAM configured for batteries?: Yes
Start cache flushing at (in percentage): 8
Stop cache flushing at (in percentage): 8
Cache block size (in KB): 16
Media scan frequency (in days): Disabled
Failover alert delay (in minutes): 5
Feature enable identifier: 3037373535003038303935003F379E17
Storage Subsystem worldwide name (ID):
600A0B80001361CE000000004163CEF7

CONTROLLERS-----

Number of controllers: 2

Controller in Slot A

Status: Online

Current configuration

Firmware version: 06.12.03.00

Appware version: 06.12.03.00

Bootware version: 06.10.04.00

NVSRAM version: N1742F900R912V06

Pending configuration

Firmware version: None

Appware version: None

Bootware version: None

NVSRAM version: None

Transferred on: None

Board ID: 5884

Product ID: 1742-900

Product revision: 0520

Serial number: 1T40667740

Date of manufacture: February 19, 2004

Cache/processor size (MB): 1024/128

Date/Time: Fri May 20 08:52:31 EDT 2005

Associated Logical Drives (* = Preferred Owner):

RACK3LUN0*, RACK3LUN10*, RACK3LUN12*, RACK3LUN2*,

RACK3LUN4*, RACK3LUN6*,

RACK3LUN8*

Ethernet port: 1

MAC address: 00:a0:b8:13:61:ce

Host name: FLUTE_02A

Network configuration: Static

IP address: 192.168.122.244

Subnet mask: 255.255.255.0

Gateway: 0.0.0.0

Remote login: Disabled

Drive interface: Fibre

Channel: 1

Current ID: 125/0x1

Maximum data rate: 2 Gbps

Current data rate: 2 Gbps

Data rate control: Switch

Link status: Up

Drive interface: Fibre

Channel: 2

Current ID: 125/0x1

Maximum data rate: 2 Gbps

Current data rate: 2 Gbps

Data rate control: Switch

Link status: Up

Drive interface: Fibre

Channel: 3

Current ID: 125/0x1

Maximum data rate: 2 Gbps

Current data rate: 2 Gbps

Data rate control: Switch

Link status: Up

Drive interface: Fibre

Channel: 4

Current ID: 125/0x1

Maximum data rate: 2 Gbps

Current data rate: 2 Gbps

Data rate control: Switch

Link status: Up

Host interface: Fibre

Port: 1

Current ID: 0/0xEF

Preferred ID: 0/0xEF

NL-Port ID: 0x0000EF

Maximum data rate: 2 Gbps

Current data rate: 2 Gbps

Data rate control: Switch

Link status: Up

Topology: Arbitrated Loop - Private

World-wide port name: 20:02:00:a0:b8:13:61:cf

World-wide node name: 20:02:00:a0:b8:13:61:ce

Part type: HPFC-5400 revision 6

Host interface: Fibre

Port: 2

Current ID: Not applicable/0xFFFFFFFF

Preferred ID: 1/0xE8

NL-Port ID: 0x000000

Maximum data rate: 2 Gbps

Current data rate: 1 Gbps

Data rate control: Switch

Link status: Down

Topology: Not available

World-wide port name: 20:02:00:a0:b8:13:61:d0

World-wide node name: 20:02:00:a0:b8:13:61:ce

Part type: HPFC-5400 revision 6

Controller in Slot B

Status: Online

Current configuration

Firmware version: 06.12.03.00

Appware version: 06.12.03.00

Bootware version: 06.10.04.00

NVSRAM version: N1742F900R912V06

Pending configuration

Firmware version: None

Appware version: None

Bootware version: None

NVSRAM version: None

Transferred on: None

Board ID: 5884

Product ID: 1742-900

Product revision: 0520

Serial number: 1T40667811

Date of manufacture: February 19, 2004

Cache/processor size (MB): 1024/128

Date/Time: Fri May 20 08:52:56 EDT 2005

Associated Logical Drives (* = Preferred Owner):

RACK3LUN1*, RACK3LUN11*, RACK3LUN13*, RACK3LUN3*,

RACK3LUN5*, RACK3LUN7*,

RACK3LUN9*

Ethernet port: 1

MAC address: 00:a0:b8:13:60:24

Host name: FLUTE_02B

Network configuration: Static

IP address: 192.168.122.245

Subnet mask: 255.255.255.0
Gateway: 0.0.0.0
Remote login: Disabled

Drive interface: Fibre
Channel: 1
Current ID: 124/0x2
Maximum data rate: 2 Gbps
Current data rate: 2 Gbps
Data rate control: Switch
Link status: Up

Drive interface: Fibre
Channel: 2
Current ID: 124/0x2
Maximum data rate: 2 Gbps
Current data rate: 2 Gbps
Data rate control: Switch
Link status: Up

Drive interface: Fibre
Channel: 3
Current ID: 124/0x2
Maximum data rate: 2 Gbps
Current data rate: 2 Gbps
Data rate control: Switch
Link status: Up

Drive interface: Fibre
Channel: 4
Current ID: 124/0x2
Maximum data rate: 2 Gbps
Current data rate: 2 Gbps
Data rate control: Switch
Link status: Up

Host interface: Fibre
Port: 1
Current ID: 2/0xE4
Preferred ID: 2/0xE4
NL-Port ID: 0x0000E4
Maximum data rate: 2 Gbps
Current data rate: 2 Gbps
Data rate control: Switch
Link status: Up
Topology: Arbitrated Loop - Private
World-wide port name: 20:03:00:a0:b8:13:61:cf
World-wide node name: 20:03:00:a0:b8:13:61:ce
Part type: HPFC-5400 revision 6

Host interface: Fibre
Port: 2
Current ID: Not applicable/0xFFFFFFFF
Preferred ID: 3/0xE2
NL-Port ID: 0x000000
Maximum data rate: 2 Gbps
Current data rate: 1 Gbps
Data rate control: Switch
Link status: Down
Topology: Not available
World-wide port name: 20:03:00:a0:b8:13:61:d0
World-wide node name: 20:03:00:a0:b8:13:61:ce
Part type: HPFC-5400 revision 6

ARRAYS-----
Number of arrays: 14

Array 1 (RAID 0)
Status: Online
Drive type: Fibre Channel
Enclosure loss protection: No
Current owner: Controller in slot A
Associated logical drives and free capacities:

RACK3LUN0 (334.014 GB)
Associated drives (in piece order):
Drive at Enclosure 40, Slot 1
Drive at Enclosure 41, Slot 1
Drive at Enclosure 42, Slot 1
Drive at Enclosure 43, Slot 1
Drive at Enclosure 44, Slot 1
Drive at Enclosure 50, Slot 1
Drive at Enclosure 51, Slot 1
Drive at Enclosure 52, Slot 1
Drive at Enclosure 53, Slot 1
Drive at Enclosure 54, Slot 1

Array 2 (RAID 0)
Status: Online
Drive type: Fibre Channel
Enclosure loss protection: No
Current owner: Controller in slot B
Associated logical drives and free capacities:

RACK3LUN1 (334.014 GB)
Associated drives (in piece order):
Drive at Enclosure 40, Slot 2
Drive at Enclosure 41, Slot 2
Drive at Enclosure 42, Slot 2
Drive at Enclosure 43, Slot 2
Drive at Enclosure 44, Slot 2
Drive at Enclosure 50, Slot 2
Drive at Enclosure 51, Slot 2
Drive at Enclosure 52, Slot 2
Drive at Enclosure 53, Slot 2
Drive at Enclosure 54, Slot 2

Array 3 (RAID 0)
Status: Online
Drive type: Fibre Channel
Enclosure loss protection: No
Current owner: Controller in slot A
Associated logical drives and free capacities:

RACK3LUN2 (334.014 GB)
Associated drives (in piece order):
Drive at Enclosure 40, Slot 3
Drive at Enclosure 41, Slot 3
Drive at Enclosure 42, Slot 3
Drive at Enclosure 43, Slot 3
Drive at Enclosure 44, Slot 3
Drive at Enclosure 50, Slot 3
Drive at Enclosure 51, Slot 3
Drive at Enclosure 52, Slot 3
Drive at Enclosure 53, Slot 3
Drive at Enclosure 54, Slot 3

Array 4 (RAID 0)
Status: Online
Drive type: Fibre Channel
Enclosure loss protection: No
Current owner: Controller in slot B
Associated logical drives and free capacities:

RACK3LUN3 (334.014 GB)
Associated drives (in piece order):
Drive at Enclosure 40, Slot 4
Drive at Enclosure 41, Slot 4
Drive at Enclosure 42, Slot 4
Drive at Enclosure 43, Slot 4
Drive at Enclosure 44, Slot 4
Drive at Enclosure 50, Slot 4
Drive at Enclosure 51, Slot 4
Drive at Enclosure 52, Slot 4
Drive at Enclosure 53, Slot 4
Drive at Enclosure 54, Slot 4

Array 5 (RAID 0)
Status: Online

Drive type: Fibre Channel
Enclosure loss protection: No
Current owner: Controller in slot A
Associated logical drives and free capacities:
RACK3LUN4 (334.014 GB)
Associated drives (in piece order):
Drive at Enclosure 40, Slot 5
Drive at Enclosure 41, Slot 5
Drive at Enclosure 42, Slot 5
Drive at Enclosure 43, Slot 5
Drive at Enclosure 44, Slot 5
Drive at Enclosure 50, Slot 5
Drive at Enclosure 51, Slot 5
Drive at Enclosure 52, Slot 5
Drive at Enclosure 53, Slot 5
Drive at Enclosure 54, Slot 5

Array 6 (RAID 0)
Status: Online
Drive type: Fibre Channel
Enclosure loss protection: No
Current owner: Controller in slot B
Associated logical drives and free capacities:
RACK3LUN5 (334.014 GB)
Associated drives (in piece order):
Drive at Enclosure 40, Slot 6
Drive at Enclosure 41, Slot 6
Drive at Enclosure 42, Slot 6
Drive at Enclosure 43, Slot 6
Drive at Enclosure 44, Slot 6
Drive at Enclosure 50, Slot 6
Drive at Enclosure 51, Slot 6
Drive at Enclosure 52, Slot 6
Drive at Enclosure 53, Slot 6
Drive at Enclosure 54, Slot 6

Array 7 (RAID 0)
Status: Online
Drive type: Fibre Channel
Enclosure loss protection: No
Current owner: Controller in slot A
Associated logical drives and free capacities:
RACK3LUN6 (334.014 GB)
Associated drives (in piece order):
Drive at Enclosure 40, Slot 7
Drive at Enclosure 41, Slot 7
Drive at Enclosure 42, Slot 7
Drive at Enclosure 43, Slot 7
Drive at Enclosure 44, Slot 7
Drive at Enclosure 50, Slot 7
Drive at Enclosure 51, Slot 7
Drive at Enclosure 52, Slot 7
Drive at Enclosure 53, Slot 7
Drive at Enclosure 54, Slot 7

Array 8 (RAID 0)
Status: Online
Drive type: Fibre Channel
Enclosure loss protection: No
Current owner: Controller in slot B
Associated logical drives and free capacities:
RACK3LUN7 (334.014 GB)
Associated drives (in piece order):
Drive at Enclosure 40, Slot 8
Drive at Enclosure 41, Slot 8
Drive at Enclosure 42, Slot 8
Drive at Enclosure 43, Slot 8
Drive at Enclosure 44, Slot 8
Drive at Enclosure 50, Slot 8
Drive at Enclosure 51, Slot 8
Drive at Enclosure 52, Slot 8

Drive at Enclosure 53, Slot 8
Drive at Enclosure 54, Slot 8
Array 9 (RAID 0)
Status: Online
Drive type: Fibre Channel
Enclosure loss protection: No
Current owner: Controller in slot A
Associated logical drives and free capacities:
RACK3LUN8 (334.014 GB)
Associated drives (in piece order):
Drive at Enclosure 40, Slot 9
Drive at Enclosure 41, Slot 9
Drive at Enclosure 42, Slot 9
Drive at Enclosure 43, Slot 9
Drive at Enclosure 44, Slot 9
Drive at Enclosure 50, Slot 9
Drive at Enclosure 51, Slot 9
Drive at Enclosure 52, Slot 9
Drive at Enclosure 53, Slot 9
Drive at Enclosure 54, Slot 9

Array 10 (RAID 0)
Status: Online
Drive type: Fibre Channel
Enclosure loss protection: No
Current owner: Controller in slot B
Associated logical drives and free capacities:
RACK3LUN9 (334.014 GB)
Associated drives (in piece order):
Drive at Enclosure 40, Slot 10
Drive at Enclosure 41, Slot 10
Drive at Enclosure 42, Slot 10
Drive at Enclosure 43, Slot 10
Drive at Enclosure 44, Slot 10
Drive at Enclosure 50, Slot 10
Drive at Enclosure 51, Slot 10
Drive at Enclosure 52, Slot 10
Drive at Enclosure 53, Slot 10
Drive at Enclosure 54, Slot 10

Array 11 (RAID 0)
Status: Online
Drive type: Fibre Channel
Enclosure loss protection: No
Current owner: Controller in slot A
Associated logical drives and free capacities:
RACK3LUN10 (334.014 GB)
Associated drives (in piece order):
Drive at Enclosure 40, Slot 11
Drive at Enclosure 41, Slot 11
Drive at Enclosure 42, Slot 11
Drive at Enclosure 43, Slot 11
Drive at Enclosure 44, Slot 11
Drive at Enclosure 50, Slot 11
Drive at Enclosure 51, Slot 11
Drive at Enclosure 52, Slot 11
Drive at Enclosure 53, Slot 11
Drive at Enclosure 54, Slot 11

Array 12 (RAID 0)
Status: Online
Drive type: Fibre Channel
Enclosure loss protection: No
Current owner: Controller in slot B
Associated logical drives and free capacities:
RACK3LUN11 (334.014 GB)
Associated drives (in piece order):
Drive at Enclosure 40, Slot 12
Drive at Enclosure 41, Slot 12
Drive at Enclosure 42, Slot 12
Drive at Enclosure 43, Slot 12

Drive at Enclosure 44, Slot 12
 Drive at Enclosure 50, Slot 12
 Drive at Enclosure 51, Slot 12
 Drive at Enclosure 52, Slot 12
 Drive at Enclosure 53, Slot 12
 Drive at Enclosure 54, Slot 12

Array 13 (RAID 0)

Status: Online
 Drive type: Fibre Channel
 Enclosure loss protection: No
 Current owner: Controller in slot A
 Associated logical drives and free capacities:
 RACK3LUN12 (334.014 GB)

Associated drives (in piece order):

Drive at Enclosure 40, Slot 13
 Drive at Enclosure 41, Slot 13
 Drive at Enclosure 42, Slot 13
 Drive at Enclosure 43, Slot 13
 Drive at Enclosure 44, Slot 13
 Drive at Enclosure 50, Slot 13
 Drive at Enclosure 51, Slot 13
 Drive at Enclosure 52, Slot 13
 Drive at Enclosure 53, Slot 13
 Drive at Enclosure 54, Slot 13

Array 14 (RAID 5)

Status: Online
 Drive type: Fibre Channel
 Enclosure loss protection: Yes
 Current owner: Controller in slot B
 Associated logical drives and free capacities:
 RACK3LUN13 (300.612 GB)

Associated drives (in piece order):

Drive at Enclosure 40, Slot 14
 Drive at Enclosure 41, Slot 14
 Drive at Enclosure 42, Slot 14
 Drive at Enclosure 43, Slot 14
 Drive at Enclosure 44, Slot 14
 Drive at Enclosure 50, Slot 14
 Drive at Enclosure 51, Slot 14
 Drive at Enclosure 52, Slot 14
 Drive at Enclosure 53, Slot 14
 Drive at Enclosure 54, Slot 14

STANDARD LOGICAL DRIVES-----

SUMMARY

Number of standard logical drives: 14
 See other Logical Drives sub-tabs for premium feature information.

NAME	STATUS	CAPACITY	RAID LEVEL	ARRAY
RACK3LUN0	Optimal	334.014 GB	0	1
RACK3LUN1	Optimal	334.014 GB	0	2
RACK3LUN10	Optimal	334.014 GB	0	11
RACK3LUN11	Optimal	334.014 GB	0	12
RACK3LUN12	Optimal	334.014 GB	0	13
RACK3LUN13	Optimal	300.612 GB	5	14
RACK3LUN2	Optimal	334.014 GB	0	3
RACK3LUN3	Optimal	334.014 GB	0	4
RACK3LUN4	Optimal	334.014 GB	0	5
RACK3LUN5	Optimal	334.014 GB	0	6
RACK3LUN6	Optimal	334.014 GB	0	7
RACK3LUN7	Optimal	334.014 GB	0	8
RACK3LUN8	Optimal	334.014 GB	0	9
RACK3LUN9	Optimal	334.014 GB	0	10

DETAILS

Logical Drive name: RACK3LUN0
 Logical Drive ID: 60:0a:0b:80:00:13:61:ce:00:00:00:00:42:52:a6:c8

Subsystem ID (SSID): 0
 Status: Optimal
 Drive type: Fibre Channel
 Enclosure loss protection: No
 Preferred owner: Controller in slot A
 Current owner: Controller in slot A
 Capacity: 334.014 GB
 RAID level: 0
 Segment size: 64 KB
 Modification priority: High
 Associated array: 1
 Read cache: Enabled
 Write cache: Enabled
 Write cache without batteries: Disabled
 Write cache with mirroring: Disabled
 Flush write cache after (in seconds): 10.00
 Cache read ahead multiplier: 0
 Enable background media scan: Disabled
 Media scan with redundancy check: Disabled

Logical Drive name: RACK3LUN1

Logical Drive ID: 60:0a:0b:80:00:13:60:24:00:00:00:00:42:52:a8:47

Subsystem ID (SSID): 1
 Status: Optimal
 Drive type: Fibre Channel
 Enclosure loss protection: No
 Preferred owner: Controller in slot B
 Current owner: Controller in slot B
 Capacity: 334.014 GB
 RAID level: 0
 Segment size: 64 KB
 Modification priority: High
 Associated array: 2
 Read cache: Enabled
 Write cache: Enabled
 Write cache without batteries: Disabled
 Write cache with mirroring: Disabled
 Flush write cache after (in seconds): 10.00
 Cache read ahead multiplier: 0
 Enable background media scan: Disabled
 Media scan with redundancy check: Disabled

Logical Drive name: RACK3LUN10

Logical Drive ID: 60:0a:0b:80:00:13:61:ce:00:00:00:0f:42:52:a9:2a

Subsystem ID (SSID): 10
 Status: Optimal
 Drive type: Fibre Channel
 Enclosure loss protection: No
 Preferred owner: Controller in slot A
 Current owner: Controller in slot A
 Capacity: 334.014 GB
 RAID level: 0
 Segment size: 64 KB
 Modification priority: High
 Associated array: 11
 Read cache: Enabled
 Write cache: Enabled
 Write cache without batteries: Disabled
 Write cache with mirroring: Disabled
 Flush write cache after (in seconds): 10.00
 Cache read ahead multiplier: 0
 Enable background media scan: Disabled
 Media scan with redundancy check: Disabled

Logical Drive name: RACK3LUN11

Logical Drive ID: 60:0a:0b:80:00:13:60:24:00:00:00:0a:42:52:aa:61

Subsystem ID (SSID): 11
 Status: Optimal

Drive type: Fibre Channel
Enclosure loss protection: No
Preferred owner: Controller in slot B
Current owner: Controller in slot B
Capacity: 334.014 GB
RAID level: 0
Segment size: 64 KB
Modification priority: High
Associated array: 12
Read cache: Enabled
Write cache: Enabled
Write cache without batteries: Disabled
Write cache with mirroring: Disabled
Flush write cache after (in seconds): 10.00
Cache read ahead multiplier: 0
Enable background media scan: Disabled
Media scan with redundancy check: Disabled

Logical Drive name: RACK3LUN12
Logical Drive ID: 60:0a:0b:80:00:13:61:ce:00:00:00:12:42:52:a9:92
Subsystem ID (SSID): 12
Status: Optimal
Drive type: Fibre Channel
Enclosure loss protection: No
Preferred owner: Controller in slot A
Current owner: Controller in slot A
Capacity: 334.014 GB
RAID level: 0
Segment size: 64 KB
Modification priority: High
Associated array: 13
Read cache: Enabled
Write cache: Enabled
Write cache without batteries: Disabled
Write cache with mirroring: Disabled
Flush write cache after (in seconds): 10.00
Cache read ahead multiplier: 0
Enable background media scan: Disabled
Media scan with redundancy check: Disabled

Logical Drive name: RACK3LUN13
Logical Drive ID: 60:0a:0b:80:00:13:60:24:00:00:00:0e:42:53:84:df
Subsystem ID (SSID): 13
Status: Optimal
Drive type: Fibre Channel
Enclosure loss protection: Yes
Preferred owner: Controller in slot B
Current owner: Controller in slot B
Capacity: 300.612 GB
RAID level: 5
Segment size: 512 KB
Modification priority: High
Associated array: 14
Read cache: Enabled
Write cache: Enabled
Write cache without batteries: Disabled
Write cache with mirroring: Disabled
Flush write cache after (in seconds): 10.00
Cache read ahead multiplier: 0
Enable background media scan: Disabled
Media scan with redundancy check: Disabled

Logical Drive name: RACK3LUN2
Logical Drive ID: 60:0a:0b:80:00:13:61:ce:00:00:00:03:42:52:a7:7a
Subsystem ID (SSID): 2
Status: Optimal
Drive type: Fibre Channel
Enclosure loss protection: No

Preferred owner: Controller in slot A
Current owner: Controller in slot A
Capacity: 334.014 GB
RAID level: 0
Segment size: 64 KB
Modification priority: High
Associated array: 3
Read cache: Enabled
Write cache: Enabled
Write cache without batteries: Disabled
Write cache with mirroring: Disabled
Flush write cache after (in seconds): 10.00
Cache read ahead multiplier: 0
Enable background media scan: Disabled
Media scan with redundancy check: Disabled

Logical Drive name: RACK3LUN3
Logical Drive ID: 60:0a:0b:80:00:13:60:24:00:00:00:02:42:52:a8:c1
Subsystem ID (SSID): 3
Status: Optimal
Drive type: Fibre Channel
Enclosure loss protection: No
Preferred owner: Controller in slot B
Current owner: Controller in slot B
Capacity: 334.014 GB
RAID level: 0
Segment size: 64 KB
Modification priority: High
Associated array: 4
Read cache: Enabled
Write cache: Enabled
Write cache without batteries: Disabled
Write cache with mirroring: Disabled
Flush write cache after (in seconds): 10.00
Cache read ahead multiplier: 0
Enable background media scan: Disabled
Media scan with redundancy check: Disabled

Logical Drive name: RACK3LUN4
Logical Drive ID: 60:0a:0b:80:00:13:61:ce:00:00:00:06:42:52:a7:f0
Subsystem ID (SSID): 4
Status: Optimal
Drive type: Fibre Channel
Enclosure loss protection: No
Preferred owner: Controller in slot A
Current owner: Controller in slot A
Capacity: 334.014 GB
RAID level: 0
Segment size: 64 KB
Modification priority: High
Associated array: 5
Read cache: Enabled
Write cache: Enabled
Write cache without batteries: Disabled
Write cache with mirroring: Disabled
Flush write cache after (in seconds): 10.00
Cache read ahead multiplier: 0
Enable background media scan: Disabled
Media scan with redundancy check: Disabled

Logical Drive name: RACK3LUN5
Logical Drive ID: 60:0a:0b:80:00:13:60:24:00:00:00:04:42:52:a9:23
Subsystem ID (SSID): 5
Status: Optimal
Drive type: Fibre Channel
Enclosure loss protection: No
Preferred owner: Controller in slot B
Current owner: Controller in slot B

Capacity: 334.014 GB
 RAID level: 0
 Segment size: 64 KB
 Modification priority: High
 Associated array: 6
 Read cache: Enabled
 Write cache: Enabled
 Write cache without batteries: Disabled
 Write cache with mirroring: Disabled
 Flush write cache after (in seconds): 10.00
 Cache read ahead multiplier: 0
 Enable background media scan: Disabled
 Media scan with redundancy check: Disabled

Logical Drive name: RACK3LUN6
 Logical Drive ID: 60:0a:0b:80:00:13:61:ce:00:00:00:09:42:52:a8:4c
 Subsystem ID (SSID): 6
 Status: Optimal
 Drive type: Fibre Channel
 Enclosure loss protection: No
 Preferred owner: Controller in slot A
 Current owner: Controller in slot A
 Capacity: 334.014 GB
 RAID level: 0
 Segment size: 64 KB
 Modification priority: High
 Associated array: 7
 Read cache: Enabled
 Write cache: Enabled
 Write cache without batteries: Disabled
 Write cache with mirroring: Disabled
 Flush write cache after (in seconds): 10.00
 Cache read ahead multiplier: 0
 Enable background media scan: Disabled
 Media scan with redundancy check: Disabled

Logical Drive name: RACK3LUN7
 Logical Drive ID: 60:0a:0b:80:00:13:60:24:00:00:00:06:42:52:a9:a3
 Subsystem ID (SSID): 7
 Status: Optimal
 Drive type: Fibre Channel
 Enclosure loss protection: No
 Preferred owner: Controller in slot B
 Current owner: Controller in slot B
 Capacity: 334.014 GB
 RAID level: 0
 Segment size: 64 KB
 Modification priority: High
 Associated array: 8
 Read cache: Enabled
 Write cache: Enabled
 Write cache without batteries: Disabled
 Write cache with mirroring: Disabled
 Flush write cache after (in seconds): 10.00
 Cache read ahead multiplier: 0
 Enable background media scan: Disabled
 Media scan with redundancy check: Disabled

Logical Drive name: RACK3LUN8
 Logical Drive ID: 60:0a:0b:80:00:13:61:ce:00:00:00:0c:42:52:a8:ca
 Subsystem ID (SSID): 8
 Status: Optimal
 Drive type: Fibre Channel
 Enclosure loss protection: No
 Preferred owner: Controller in slot A
 Current owner: Controller in slot A
 Capacity: 334.014 GB
 RAID level: 0

Segment size: 64 KB
 Modification priority: High
 Associated array: 9
 Read cache: Enabled
 Write cache: Enabled
 Write cache without batteries: Disabled
 Write cache with mirroring: Disabled
 Flush write cache after (in seconds): 10.00
 Cache read ahead multiplier: 0
 Enable background media scan: Disabled
 Media scan with redundancy check: Disabled

Logical Drive name: RACK3LUN9
 Logical Drive ID: 60:0a:0b:80:00:13:60:24:00:00:00:08:42:52:a9:fb
 Subsystem ID (SSID): 9
 Status: Optimal
 Drive type: Fibre Channel
 Enclosure loss protection: No
 Preferred owner: Controller in slot B
 Current owner: Controller in slot B
 Capacity: 334.014 GB
 RAID level: 0
 Segment size: 64 KB
 Modification priority: High
 Associated array: 10
 Read cache: Enabled
 Write cache: Enabled
 Write cache without batteries: Disabled
 Write cache with mirroring: Disabled
 Flush write cache after (in seconds): 10.00
 Cache read ahead multiplier: 0
 Enable background media scan: Disabled
 Media scan with redundancy check: Disabled

MISSING LOGICAL DRIVES-----
 Number of missing logical drives: 0

DRIVES-----

SUMMARY
 Number of drives: 140
 Supported drive types: Fibre (140)

BASIC:

TRAY, SLOT	STATUS	CAPACITY	CURRENT DATA RATE
PRODUCT ID	FIRMWARE	VERSION	
40, 1	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
40, 2	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
40, 3	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
40, 4	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
40, 5	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
40, 6	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
40, 7	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
40, 8	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
40, 9	Optimal	33.902 GB 2 Gbps	ST336753FC F B954
40, 10	Optimal	33.902 GB 2 Gbps	ST336753FC F B954

50, 9	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
50, 10	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
50, 11	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
50, 12	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
50, 13	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
50, 14	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
51, 1	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
51, 2	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
51, 3	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
51, 4	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
51, 5	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
51, 6	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
51, 7	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
51, 8	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
51, 9	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
51, 10	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
51, 11	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
51, 12	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
51, 13	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
51, 14	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
52, 1	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
52, 2	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
52, 3	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
52, 4	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
52, 5	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
52, 6	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
52, 7	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
52, 8	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
52, 9	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
52, 10	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
52, 11	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
52, 12	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
52, 13	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
52, 14	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954

53, 1	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
53, 2	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
53, 3	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
53, 4	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
53, 5	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
53, 6	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
53, 7	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
53, 8	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
53, 9	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
53, 10	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
53, 11	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
53, 12	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
53, 13	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
53, 14	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
54, 1	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
54, 2	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
54, 3	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
54, 4	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
54, 5	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
54, 6	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
54, 7	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
54, 8	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
54, 9	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
54, 10	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
54, 11	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
54, 12	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
54, 13	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954
54, 14	Optimal	33.902 GB	2 Gbps	ST336753FC	F	B954

DRIVE CHANNELS:

TRAY, SLOT	PREFERRED CHANNEL	REDUNDANT CHANNEL
40, 1	2	1
40, 2	1	2
40, 3	2	1
40, 4	1	2
40, 5	2	1
40, 6	1	2
40, 7	2	1
40, 8	1	2

40, 9	2	1
40, 10	1	2
40, 11	2	1
40, 12	1	2
40, 13	2	1
40, 14	1	2
41, 1	2	1
41, 2	1	2
41, 3	2	1
41, 4	1	2
41, 5	2	1
41, 6	1	2
41, 7	2	1
41, 8	1	2
41, 9	2	1
41, 10	1	2
41, 11	2	1
41, 12	1	2
41, 13	2	1
41, 14	1	2
42, 1	2	1
42, 2	1	2
42, 3	2	1
42, 4	1	2
42, 5	2	1
42, 6	1	2
42, 7	2	1
42, 8	1	2
42, 9	2	1
42, 10	1	2
42, 11	2	1
42, 12	1	2
42, 13	2	1
42, 14	1	2
43, 1	2	1
43, 2	1	2
43, 3	2	1
43, 4	1	2
43, 5	2	1
43, 6	1	2
43, 7	2	1
43, 8	1	2
43, 9	2	1
43, 10	1	2
43, 11	2	1
43, 12	1	2
43, 13	2	1
43, 14	1	2
44, 1	2	1
44, 2	1	2
44, 3	2	1
44, 4	1	2
44, 5	2	1
44, 6	1	2
44, 7	2	1
44, 8	1	2
44, 9	2	1
44, 10	1	2
44, 11	2	1
44, 12	1	2
44, 13	2	1
44, 14	1	2
50, 1	4	3
50, 2	3	4
50, 3	4	3
50, 4	3	4
50, 5	4	3
50, 6	3	4

50, 7	4	3
50, 8	3	4
50, 9	4	3
50, 10	3	4
50, 11	4	3
50, 12	3	4
50, 13	4	3
50, 14	3	4
51, 1	4	3
51, 2	3	4
51, 3	4	3
51, 4	3	4
51, 5	4	3
51, 6	3	4
51, 7	4	3
51, 8	3	4
51, 9	4	3
51, 10	3	4
51, 11	4	3
51, 12	3	4
51, 13	4	3
51, 14	3	4
52, 1	4	3
52, 2	3	4
52, 3	4	3
52, 4	3	4
52, 5	4	3
52, 6	3	4
52, 7	4	3
52, 8	3	4
52, 9	4	3
52, 10	3	4
52, 11	4	3
52, 12	3	4
52, 13	4	3
52, 14	3	4
53, 1	4	3
53, 2	3	4
53, 3	4	3
53, 4	3	4
53, 5	4	3
53, 6	3	4
53, 7	4	3
53, 8	3	4
53, 9	4	3
53, 10	3	4
53, 11	4	3
53, 12	3	4
53, 13	4	3
53, 14	3	4
54, 1	4	3
54, 2	3	4
54, 3	4	3
54, 4	3	4
54, 5	4	3
54, 6	3	4
54, 7	4	3
54, 8	3	4
54, 9	4	3
54, 10	3	4
54, 11	4	3
54, 12	3	4
54, 13	4	3
54, 14	3	4

HOT SPARE COVERAGE:

The following arrays are not protected: 7, 3, 6, 14, 12, 1, 10, 5, 13, 4, 11, 8, 2, 9

Total hot spare drives: 0
Standby: 0
In use: 0

DETAILS

Drive at Enclosure 40, Slot 1

Drive port: 1, Channel: 2, ID: 0/0xEF
Drive port: 2, Channel: 1, ID: 0/0xEF
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HSJX000073405FSJ
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:12:88
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 1

Drive at Enclosure 40, Slot 2

Drive port: 1, Channel: 1, ID: 1/0xE8
Drive port: 2, Channel: 2, ID: 1/0xE8
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K40R00007339SP25
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:f5:7b
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 2

Drive at Enclosure 40, Slot 3

Drive port: 1, Channel: 2, ID: 2/0xE4
Drive port: 2, Channel: 1, ID: 2/0xE4
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0KD7F00007340WVXJ
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:12:e4
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 3

Drive at Enclosure 40, Slot 4

Drive port: 1, Channel: 1, ID: 3/0xE2
Drive port: 2, Channel: 2, ID: 3/0xE2
Drive path redundancy: OK
Status: Optimal

Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0KD3E000073409KX4
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:11:89
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 4

Drive at Enclosure 40, Slot 5

Drive port: 1, Channel: 2, ID: 4/0xE1
Drive port: 2, Channel: 1, ID: 4/0xE1
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HSZF000073392DU9
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e3:78
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 5

Drive at Enclosure 40, Slot 6

Drive port: 1, Channel: 1, ID: 5/0xE0
Drive port: 2, Channel: 2, ID: 5/0xE0
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HM8W00007340X4YM
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:fb:99
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 6

Drive at Enclosure 40, Slot 7

Drive port: 1, Channel: 2, ID: 6/0xDC
Drive port: 2, Channel: 1, ID: 6/0xDC
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K03N00007340X59K
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:0d:2d
Drive type: Fibre Channel
Speed: 15015 RPM

Mode: Assigned
Associated array: 7

Drive at Enclosure 40, Slot 8

Drive port: 1, Channel: 1, ID: 7/0xDA
Drive port: 2, Channel: 2, ID: 7/0xDA
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0KD5X000073405FVJ
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:12:5d
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 8

Drive at Enclosure 40, Slot 9

Drive port: 1, Channel: 2, ID: 80/0x55
Drive port: 2, Channel: 1, ID: 80/0x55
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HMX4000073409KF1
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:db:2f
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 9

Drive at Enclosure 40, Slot 10

Drive port: 1, Channel: 1, ID: 96/0x3A
Drive port: 2, Channel: 2, ID: 96/0x3A
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HLYW00007340YESX
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:dd:8a
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 10

Drive at Enclosure 40, Slot 11

Drive port: 1, Channel: 2, ID: 64/0x72
Drive port: 2, Channel: 1, ID: 64/0x72
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps

Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K0V100007339SP68
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:08:41
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 11

Drive at Enclosure 40, Slot 12

Drive port: 1, Channel: 1, ID: 72/0x67
Drive port: 2, Channel: 2, ID: 72/0x67
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HNCH00007339SNP5
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:02:45
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 12

Drive at Enclosure 40, Slot 13

Drive port: 1, Channel: 2, ID: 88/0x4B
Drive port: 2, Channel: 1, ID: 88/0x4B
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0JT8L00007339GMT3
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:f9:fc:22
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 13

Drive at Enclosure 40, Slot 14

Drive port: 1, Channel: 1, ID: 104/0x2E
Drive port: 2, Channel: 2, ID: 104/0x2E
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K5YK00007339RGCS
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:fa:e1
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 14

Drive at Enclosure 41, Slot 1
Drive port: 1, Channel: 2, ID: 8/0xD9
Drive port: 2, Channel: 1, ID: 8/0xD9
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX23XZQ0000743449PP
Vendor: IBM-ESXS
Date of manufacture: February 26, 2004
World-wide name: 20:00:00:0c:50:b6:5d:69
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 1

Drive at Enclosure 41, Slot 2
Drive port: 1, Channel: 1, ID: 9/0xD6
Drive port: 2, Channel: 2, ID: 9/0xD6
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0JQYY000073405G81
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e5:db
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 2

Drive at Enclosure 41, Slot 3
Drive port: 1, Channel: 2, ID: 10/0xD5
Drive port: 2, Channel: 1, ID: 10/0xD5
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HRQF00007339TYAC
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:ea:5f
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 3

Drive at Enclosure 41, Slot 4
Drive port: 1, Channel: 1, ID: 11/0xD4
Drive port: 2, Channel: 2, ID: 11/0xD4
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX117K9000073514CQT

Vendor: IBM-ESXS
Date of manufacture: June 19, 2003
World-wide name: 20:00:00:0c:50:3d:9d:a5
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 4

Drive at Enclosure 41, Slot 5
Drive port: 1, Channel: 2, ID: 12/0xD3
Drive port: 2, Channel: 1, ID: 12/0xD3
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HRV300007340WVQ3
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:ec:a4
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 5

Drive at Enclosure 41, Slot 6
Drive port: 1, Channel: 1, ID: 13/0xD2
Drive port: 2, Channel: 2, ID: 13/0xD2
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HPLN00007339GN40
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:02:b8
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 6

Drive at Enclosure 41, Slot 7
Drive port: 1, Channel: 2, ID: 14/0xD1
Drive port: 2, Channel: 1, ID: 14/0xD1
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HT0H000073392DUC
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e3:46
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 7

Drive at Enclosure 41, Slot 8
Drive port: 1, Channel: 1, ID: 15/0xCE
Drive port: 2, Channel: 2, ID: 15/0xCE

Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0KBPL000073405FUU
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:12:8b
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 8

Drive at Enclosure 41, Slot 9
Drive port: 1, Channel: 2, ID: 81/0x54
Drive port: 2, Channel: 1, ID: 81/0x54
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K0V300007339SNTZ
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:02:3b
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 9

Drive at Enclosure 41, Slot 10
Drive port: 1, Channel: 1, ID: 97/0x39
Drive port: 2, Channel: 2, ID: 97/0x39
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K5BR00007340WVK7
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:ed:52
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 10

Drive at Enclosure 41, Slot 11
Drive port: 1, Channel: 2, ID: 65/0x71
Drive port: 2, Channel: 1, ID: 65/0x71
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K6NF000073393FD6
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:02:8c

Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 11

Drive at Enclosure 41, Slot 12
Drive port: 1, Channel: 1, ID: 73/0x66
Drive port: 2, Channel: 2, ID: 73/0x66
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HSZ9000073392E2L
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e2:ca
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 12

Drive at Enclosure 41, Slot 13
Drive port: 1, Channel: 2, ID: 89/0x4A
Drive port: 2, Channel: 1, ID: 89/0x4A
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX24C9000007433GUUW
Vendor: IBM-ESXS
Date of manufacture: February 27, 2004
World-wide name: 20:00:00:0c:50:b6:64:4f
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 13

Drive at Enclosure 41, Slot 14
Drive port: 1, Channel: 1, ID: 105/0x2D
Drive port: 2, Channel: 2, ID: 105/0x2D
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0JPE700007340GTYD
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:f4:ec
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 14

Drive at Enclosure 42, Slot 1
Drive port: 1, Channel: 2, ID: 16/0xCD
Drive port: 2, Channel: 1, ID: 16/0xCD
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB

Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K67H00007339SP3G
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:f5:40
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 1

Drive at Enclosure 42, Slot 2
Drive port: 1, Channel: 1, ID: 17/0xCC
Drive port: 2, Channel: 2, ID: 17/0xCC
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HT48000073392DQT
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:ea:28
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 2

Drive at Enclosure 42, Slot 3
Drive port: 1, Channel: 2, ID: 18/0xCB
Drive port: 2, Channel: 1, ID: 18/0xCB
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HK8Q00007339SNNC
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:f5:1b
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 3

Drive at Enclosure 42, Slot 4
Drive port: 1, Channel: 1, ID: 19/0xCA
Drive port: 2, Channel: 2, ID: 19/0xCA
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K4EQ00007339SNXX
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:f5:28
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned

Associated array: 4

Drive at Enclosure 42, Slot 5
Drive port: 1, Channel: 2, ID: 20/0xC9
Drive port: 2, Channel: 1, ID: 20/0xC9
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K6CP00007340GU24
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:f4:e8
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 5

Drive at Enclosure 42, Slot 6
Drive port: 1, Channel: 1, ID: 21/0xC7
Drive port: 2, Channel: 2, ID: 21/0xC7
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K40A00007340GU4Z
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:f4:ea
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 6

Drive at Enclosure 42, Slot 7
Drive port: 1, Channel: 2, ID: 22/0xC6
Drive port: 2, Channel: 1, ID: 22/0xC6
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HT66000073405G6G
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e6:17
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 7

Drive at Enclosure 42, Slot 8
Drive port: 1, Channel: 1, ID: 23/0xC5
Drive port: 2, Channel: 2, ID: 23/0xC5
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F

Firmware version: B954
Serial number: 3HX0HM4E000073409KKZ
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:dc:2f
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 8

Drive at Enclosure 42, Slot 9

Drive port: 1, Channel: 2, ID: 82/0x53
Drive port: 2, Channel: 1, ID: 82/0x53
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0V56000007346PCAS
Vendor: IBM-ESXS
Date of manufacture: May 17, 2003
World-wide name: 20:00:00:0c:50:20:a2:1f
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 9

Drive at Enclosure 42, Slot 10

Drive port: 1, Channel: 1, ID: 98/0x36
Drive port: 2, Channel: 2, ID: 98/0x36
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K4SF00007339SNNK
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:f5:25
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 10

Drive at Enclosure 42, Slot 11

Drive port: 1, Channel: 2, ID: 66/0x6E
Drive port: 2, Channel: 1, ID: 66/0x6E
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HM52000073409KEZ
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:dd:45
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 11

Drive at Enclosure 42, Slot 12

Drive port: 1, Channel: 1, ID: 74/0x65
Drive port: 2, Channel: 2, ID: 74/0x65
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HSQF00007339RGRW
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e1:97
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 12

Drive at Enclosure 42, Slot 13

Drive port: 1, Channel: 2, ID: 90/0x49
Drive port: 2, Channel: 1, ID: 90/0x49
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HLYD000073409K2D
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:dd:bc
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 13

Drive at Enclosure 42, Slot 14

Drive port: 1, Channel: 1, ID: 106/0x2C
Drive port: 2, Channel: 2, ID: 106/0x2C
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HSWK000073392E28
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e2:45
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 14

Drive at Enclosure 43, Slot 1

Drive port: 1, Channel: 2, ID: 24/0xC3
Drive port: 2, Channel: 1, ID: 24/0xC3
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HN8F00007339GMGU
Vendor: IBM-ESXS

Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:da:46
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 1

Drive at Enclosure 43, Slot 2
Drive port: 1, Channel: 1, ID: 25/0xBC
Drive port: 2, Channel: 2, ID: 25/0xBC
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HT1W00007339SNMS
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e2:b8
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 2

Drive at Enclosure 43, Slot 3
Drive port: 1, Channel: 2, ID: 26/0xBA
Drive port: 2, Channel: 1, ID: 26/0xBA
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX34181000075172RVA
Vendor: IBM-ESXS
Date of manufacture: October 29, 2004
World-wide name: 20:00:00:11:c6:23:dd:6c
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 3

Drive at Enclosure 43, Slot 4
Drive port: 1, Channel: 1, ID: 27/0xB9
Drive port: 2, Channel: 2, ID: 27/0xB9
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HLYY000073409KGF
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:dd:bf
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 4

Drive at Enclosure 43, Slot 5
Drive port: 1, Channel: 2, ID: 28/0xB6
Drive port: 2, Channel: 1, ID: 28/0xB6
Drive path redundancy: OK

Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0GTWN000073392DYK
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e2:e4
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 5

Drive at Enclosure 43, Slot 6
Drive port: 1, Channel: 1, ID: 29/0xB5
Drive port: 2, Channel: 2, ID: 29/0xB5
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K86A000073392DZ9
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e2:58
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 6

Drive at Enclosure 43, Slot 7
Drive port: 1, Channel: 2, ID: 30/0xB4
Drive port: 2, Channel: 1, ID: 30/0xB4
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K57Q000073410HSY
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:12:9a
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 7

Drive at Enclosure 43, Slot 8
Drive port: 1, Channel: 1, ID: 31/0xB3
Drive port: 2, Channel: 2, ID: 31/0xB3
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K5W9000073405FWF
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:12:c9
Drive type: Fibre Channel

Speed: 15015 RPM
Mode: Assigned
Associated array: 8

Drive at Enclosure 43, Slot 9

Drive port: 1, Channel: 2, ID: 83/0x52
Drive port: 2, Channel: 1, ID: 83/0x52
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K4EJ00007339SNWV
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:f5:5f
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 9

Drive at Enclosure 43, Slot 10

Drive port: 1, Channel: 1, ID: 99/0x35
Drive port: 2, Channel: 2, ID: 99/0x35
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HQ4D00007339SP4Z
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:f9:f5:c5
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 10

Drive at Enclosure 43, Slot 11

Drive port: 1, Channel: 2, ID: 67/0x6D
Drive port: 2, Channel: 1, ID: 67/0x6D
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0KBS6000073405FXS
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:12:9c
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 11

Drive at Enclosure 43, Slot 12

Drive port: 1, Channel: 1, ID: 75/0x63
Drive port: 2, Channel: 2, ID: 75/0x63
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB

Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0JPQ7000073392DXG
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:18:c7
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 12

Drive at Enclosure 43, Slot 13

Drive port: 1, Channel: 2, ID: 91/0x47
Drive port: 2, Channel: 1, ID: 91/0x47
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0JAE000007339GMPM
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:fc:61
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 13

Drive at Enclosure 43, Slot 14

Drive port: 1, Channel: 1, ID: 107/0x2B
Drive port: 2, Channel: 2, ID: 107/0x2B
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HWYB000073393FVR
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:f9:fc:1c
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 14

Drive at Enclosure 44, Slot 1

Drive port: 1, Channel: 2, ID: 32/0xB2
Drive port: 2, Channel: 1, ID: 32/0xB2
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HT3000007339FFYH
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e4:3d
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 1

Drive at Enclosure 44, Slot 2
Drive port: 1, Channel: 1, ID: 33/0xB1
Drive port: 2, Channel: 2, ID: 33/0xB1
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HRGN00007339TYA2
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:eb:cf
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 2

Drive at Enclosure 44, Slot 3
Drive port: 1, Channel: 2, ID: 34/0xAE
Drive port: 2, Channel: 1, ID: 34/0xAE
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HRPH00007340WVK1
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:ed:05
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 3

Drive at Enclosure 44, Slot 4
Drive port: 1, Channel: 1, ID: 35/0xAD
Drive port: 2, Channel: 2, ID: 35/0xAD
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HLWZ00007340WVK9
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:ed:5d
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 4

Drive at Enclosure 44, Slot 5
Drive port: 1, Channel: 2, ID: 36/0xAC
Drive port: 2, Channel: 1, ID: 36/0xAC
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954

Serial number: 3HX0K1BC00007339TYYW
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:02:37
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 5

Drive at Enclosure 44, Slot 6
Drive port: 1, Channel: 1, ID: 37/0xAB
Drive port: 2, Channel: 2, ID: 37/0xAB
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX33QJP00007517B1R9
Vendor: IBM-ESXS
Date of manufacture: October 25, 2004
World-wide name: 20:00:00:11:c6:23:90:c5
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 6

Drive at Enclosure 44, Slot 7
Drive port: 1, Channel: 2, ID: 38/0xAA
Drive port: 2, Channel: 1, ID: 38/0xAA
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HH1K00007339TYUA
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:fc:1f
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 7

Drive at Enclosure 44, Slot 8
Drive port: 1, Channel: 1, ID: 39/0xA9
Drive port: 2, Channel: 2, ID: 39/0xA9
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K56P000023125CXC
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:02:f2
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 8

Drive at Enclosure 44, Slot 9
Drive port: 1, Channel: 2, ID: 84/0x51

Drive port: 2, Channel: 1, ID: 84/0x51
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K7K700007340WVN8
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:ec:0e
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 9

Drive at Enclosure 44, Slot 10

Drive port: 1, Channel: 1, ID: 100/0x34
Drive port: 2, Channel: 2, ID: 100/0x34
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K1QD00007340WVYH
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:03:d4
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 10

Drive at Enclosure 44, Slot 11

Drive port: 1, Channel: 2, ID: 68/0x6C
Drive port: 2, Channel: 1, ID: 68/0x6C
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K89J00007340X534
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e9:16
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 11

Drive at Enclosure 44, Slot 12

Drive port: 1, Channel: 1, ID: 76/0x5C
Drive port: 2, Channel: 2, ID: 76/0x5C
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HM8E00007339TYTF
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003

World-wide name: 20:00:00:04:cf:ff:02:5d
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 12

Drive at Enclosure 44, Slot 13

Drive port: 1, Channel: 2, ID: 92/0x46
Drive port: 2, Channel: 1, ID: 92/0x46
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K88M000073392DV0
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e3:8a
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 13

Drive at Enclosure 44, Slot 14

Drive port: 1, Channel: 1, ID: 108/0x2A
Drive port: 2, Channel: 2, ID: 108/0x2A
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0BC3Y00007328RYAB
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:f9:dd
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 14

Drive at Enclosure 50, Slot 1

Drive port: 1, Channel: 4, ID: 0/0xEF
Drive port: 2, Channel: 3, ID: 0/0xEF
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HLYQ000073409K43
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:dd:b7
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 1

Drive at Enclosure 50, Slot 2

Drive port: 1, Channel: 3, ID: 1/0xE8
Drive port: 2, Channel: 4, ID: 1/0xE8
Drive path redundancy: OK
Status: Optimal

Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0J3SB000073409KJ7
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:da:e4
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 2

Drive at Enclosure 50, Slot 3
Drive port: 1, Channel: 4, ID: 2/0xE4
Drive port: 2, Channel: 3, ID: 2/0xE4
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HNM9000073405FEM
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:db:bc
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 3

Drive at Enclosure 50, Slot 4
Drive port: 1, Channel: 3, ID: 3/0xE2
Drive port: 2, Channel: 4, ID: 3/0xE2
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0JLFY00007339RG84
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e1:ef
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 4

Drive at Enclosure 50, Slot 5
Drive port: 1, Channel: 4, ID: 4/0xE1
Drive port: 2, Channel: 3, ID: 4/0xE1
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HM9900007340YETV
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:db:9e
Drive type: Fibre Channel
Speed: 15015 RPM

Mode: Assigned
Associated array: 5

Drive at Enclosure 50, Slot 6
Drive port: 1, Channel: 3, ID: 5/0xE0
Drive port: 2, Channel: 4, ID: 5/0xE0
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HSY400007339RGPF
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e1:ed
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 6

Drive at Enclosure 50, Slot 7
Drive port: 1, Channel: 4, ID: 6/0xDC
Drive port: 2, Channel: 3, ID: 6/0xDC
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K85X000073392DUF
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e2:71
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 7

Drive at Enclosure 50, Slot 8
Drive port: 1, Channel: 3, ID: 7/0xDA
Drive port: 2, Channel: 4, ID: 7/0xDA
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HM3Y000073409KQF
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:dc:80
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 8

Drive at Enclosure 50, Slot 9
Drive port: 1, Channel: 4, ID: 80/0x55
Drive port: 2, Channel: 3, ID: 80/0x55
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps

Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HNNH000073409K7L
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:da:ef
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 9

Drive at Enclosure 50, Slot 10
Drive port: 1, Channel: 3, ID: 96/0x3A
Drive port: 2, Channel: 4, ID: 96/0x3A
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HNSV200007339TY7Q
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e2:4e
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 10

Drive at Enclosure 50, Slot 11
Drive port: 1, Channel: 4, ID: 64/0x72
Drive port: 2, Channel: 3, ID: 64/0x72
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HWN700007339SPCB
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:02:d5
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 11

Drive at Enclosure 50, Slot 12
Drive port: 1, Channel: 3, ID: 72/0x67
Drive port: 2, Channel: 4, ID: 72/0x67
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K21400007339TYJG
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:02:98
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 12

Drive at Enclosure 50, Slot 13
Drive port: 1, Channel: 4, ID: 88/0x4B
Drive port: 2, Channel: 3, ID: 88/0x4B
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HNVB00007339RG18
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:03:a4
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 13

Drive at Enclosure 50, Slot 14
Drive port: 1, Channel: 3, ID: 104/0x2E
Drive port: 2, Channel: 4, ID: 104/0x2E
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K4EL00007335MBAA
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:f9:e0
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 14

Drive at Enclosure 51, Slot 1
Drive port: 1, Channel: 4, ID: 8/0xD9
Drive port: 2, Channel: 3, ID: 8/0xD9
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HLEA00007340WEZ5
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:04:e6
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 1

Drive at Enclosure 51, Slot 2
Drive port: 1, Channel: 3, ID: 9/0xD6
Drive port: 2, Channel: 4, ID: 9/0xD6
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HT2D000073392DSJ

Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e3:ad
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 2

Drive at Enclosure 51, Slot 3

Drive port: 1, Channel: 4, ID: 10/0xD5
Drive port: 2, Channel: 3, ID: 10/0xD5
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K18B000073392DYC
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:02:b3
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 3

Drive at Enclosure 51, Slot 4

Drive port: 1, Channel: 3, ID: 11/0xD4
Drive port: 2, Channel: 4, ID: 11/0xD4
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K2KY00007338WBA0
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:03:06
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 4

Drive at Enclosure 51, Slot 5

Drive port: 1, Channel: 4, ID: 12/0xD3
Drive port: 2, Channel: 3, ID: 12/0xD3
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0J4BJ00007339GN8Z
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:02:b9
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 5

Drive at Enclosure 51, Slot 6

Drive port: 1, Channel: 3, ID: 13/0xD2
Drive port: 2, Channel: 4, ID: 13/0xD2

Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0GVEA00007338K0TD
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:fc:54
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 6

Drive at Enclosure 51, Slot 7

Drive port: 1, Channel: 4, ID: 14/0xD1
Drive port: 2, Channel: 3, ID: 14/0xD1
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0JMBL00007339SP6R
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:02:89
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 7

Drive at Enclosure 51, Slot 8

Drive port: 1, Channel: 3, ID: 15/0xCE
Drive port: 2, Channel: 4, ID: 15/0xCE
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K1FA00007339TYMT
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:03:3f
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 8

Drive at Enclosure 51, Slot 9

Drive port: 1, Channel: 4, ID: 81/0x54
Drive port: 2, Channel: 3, ID: 81/0x54
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0J2B200007340X572
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:fa:9f

Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 9

Drive at Enclosure 51, Slot 10

Drive port: 1, Channel: 3, ID: 97/0x39
Drive port: 2, Channel: 4, ID: 97/0x39
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K7EN00007340WW8P
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:03:c5
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 10

Drive at Enclosure 51, Slot 11

Drive port: 1, Channel: 4, ID: 65/0x71
Drive port: 2, Channel: 3, ID: 65/0x71
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K87B00007339SNPS
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:03:21
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 11

Drive at Enclosure 51, Slot 12

Drive port: 1, Channel: 3, ID: 73/0x66
Drive port: 2, Channel: 4, ID: 73/0x66
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0GXXKH000073393FMH
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:02:a3
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 12

Drive at Enclosure 51, Slot 13

Drive port: 1, Channel: 4, ID: 89/0x4A
Drive port: 2, Channel: 3, ID: 89/0x4A
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB

Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HP3C00007339SPAE
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:02:dd
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 13

Drive at Enclosure 51, Slot 14

Drive port: 1, Channel: 3, ID: 105/0x2D
Drive port: 2, Channel: 4, ID: 105/0x2D
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HPKC00007338K0N4
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:02:d2
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 14

Drive at Enclosure 52, Slot 1

Drive port: 1, Channel: 4, ID: 16/0xCD
Drive port: 2, Channel: 3, ID: 16/0xCD
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0H16R00007339TYQG
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:02:d0
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 1

Drive at Enclosure 52, Slot 2

Drive port: 1, Channel: 3, ID: 17/0xCC
Drive port: 2, Channel: 4, ID: 17/0xCC
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HNT6000073393FK2
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:03:22
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned

Associated array: 2

Drive at Enclosure 52, Slot 3

Drive port: 1, Channel: 4, ID: 18/0xCB
Drive port: 2, Channel: 3, ID: 18/0xCB
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HRTY00007339TYBC
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:ea:18
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 3

Drive at Enclosure 52, Slot 4

Drive port: 1, Channel: 3, ID: 19/0xCA
Drive port: 2, Channel: 4, ID: 19/0xCA
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HRFK0000734059RJ
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e7:28
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 4

Drive at Enclosure 52, Slot 5

Drive port: 1, Channel: 4, ID: 20/0xC9
Drive port: 2, Channel: 3, ID: 20/0xC9
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0TMYQ00008305U0QC
Vendor: IBM-ESXS
Date of manufacture: May 17, 2003
World-wide name: 20:00:00:0c:50:20:a1:c8
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 5

Drive at Enclosure 52, Slot 6

Drive port: 1, Channel: 3, ID: 21/0xC7
Drive port: 2, Channel: 4, ID: 21/0xC7
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F

Firmware version: B954
Serial number: 3HX0HT8D0000734059V2
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e7:a6
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 6

Drive at Enclosure 52, Slot 7

Drive port: 1, Channel: 4, ID: 22/0xC6
Drive port: 2, Channel: 3, ID: 22/0xC6
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HT6M0000734058JL
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e7:2a
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 7

Drive at Enclosure 52, Slot 8

Drive port: 1, Channel: 3, ID: 23/0xC5
Drive port: 2, Channel: 4, ID: 23/0xC5
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HRT3000073392DYS
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:ea:bc
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 8

Drive at Enclosure 52, Slot 9

Drive port: 1, Channel: 4, ID: 82/0x53
Drive port: 2, Channel: 3, ID: 82/0x53
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K7K9000073392DZP
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:ea:8c
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 9

Drive at Enclosure 52, Slot 10

Drive port: 1, Channel: 3, ID: 98/0x36
Drive port: 2, Channel: 4, ID: 98/0x36
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HSR000007339RGR2
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e1:98
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 10

Drive at Enclosure 52, Slot 11
Drive port: 1, Channel: 4, ID: 66/0x6E
Drive port: 2, Channel: 3, ID: 66/0x6E
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HLZ5000073392DQW
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:eb:a4
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 11

Drive at Enclosure 52, Slot 12
Drive port: 1, Channel: 3, ID: 74/0x65
Drive port: 2, Channel: 4, ID: 74/0x65
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HRNB000073392DQR
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:ea:41
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 12

Drive at Enclosure 52, Slot 13
Drive port: 1, Channel: 4, ID: 90/0x49
Drive port: 2, Channel: 3, ID: 90/0x49
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K81E00007340X5J3
Vendor: IBM-ESXS

Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e8:72
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 13

Drive at Enclosure 52, Slot 14
Drive port: 1, Channel: 3, ID: 106/0x2C
Drive port: 2, Channel: 4, ID: 106/0x2C
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K7LS00007340X5JL
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e8:ad
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 14

Drive at Enclosure 53, Slot 1
Drive port: 1, Channel: 4, ID: 24/0xC3
Drive port: 2, Channel: 3, ID: 24/0xC3
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HPGH000073393FM6
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:fb:58
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 1

Drive at Enclosure 53, Slot 2
Drive port: 1, Channel: 3, ID: 25/0xBC
Drive port: 2, Channel: 4, ID: 25/0xBC
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K0TS00007339TYSD
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:ff:02:50
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 2

Drive at Enclosure 53, Slot 3
Drive port: 1, Channel: 4, ID: 26/0xBA
Drive port: 2, Channel: 3, ID: 26/0xBA
Drive path redundancy: OK

Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0JR4900007340WW3K
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:03:b7
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 3

Drive at Enclosure 53, Slot 4
Drive port: 1, Channel: 3, ID: 27/0xB9
Drive port: 2, Channel: 4, ID: 27/0xB9
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K14Z00007339SNV6
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:03:ab
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 4

Drive at Enclosure 53, Slot 5
Drive port: 1, Channel: 4, ID: 28/0xB6
Drive port: 2, Channel: 3, ID: 28/0xB6
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HM1J000073409KQC
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:dc:c4
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 5

Drive at Enclosure 53, Slot 6
Drive port: 1, Channel: 3, ID: 29/0xB5
Drive port: 2, Channel: 4, ID: 29/0xB5
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K79K0000734058FJ
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e8:15
Drive type: Fibre Channel

Speed: 15015 RPM
Mode: Assigned
Associated array: 6

Drive at Enclosure 53, Slot 7
Drive port: 1, Channel: 4, ID: 30/0xB4
Drive port: 2, Channel: 3, ID: 30/0xB4
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K89M0000734059QL
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e7:94
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 7

Drive at Enclosure 53, Slot 8
Drive port: 1, Channel: 3, ID: 31/0xB3
Drive port: 2, Channel: 4, ID: 31/0xB3
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HT9P00007340X5AA
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e8:6c
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 8

Drive at Enclosure 53, Slot 9
Drive port: 1, Channel: 4, ID: 83/0x52
Drive port: 2, Channel: 3, ID: 83/0x52
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HT7W00007340X5AF
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e8:f2
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 9

Drive at Enclosure 53, Slot 10
Drive port: 1, Channel: 3, ID: 99/0x35
Drive port: 2, Channel: 4, ID: 99/0x35
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB

Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0JN550000734059MP
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e7:a8
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 10

Drive at Enclosure 53, Slot 11
Drive port: 1, Channel: 4, ID: 67/0x6D
Drive port: 2, Channel: 3, ID: 67/0x6D
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HRM900007340X5AU
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e8:62
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 11

Drive at Enclosure 53, Slot 12
Drive port: 1, Channel: 3, ID: 75/0x63
Drive port: 2, Channel: 4, ID: 75/0x63
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0V6HR000073462EUF
Vendor: IBM-ESXS
Date of manufacture: May 17, 2003
World-wide name: 20:00:00:0c:50:20:a1:76
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 12

Drive at Enclosure 53, Slot 13
Drive port: 1, Channel: 4, ID: 91/0x47
Drive port: 2, Channel: 3, ID: 91/0x47
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0JD9D00007339TYB2
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:ea:9a
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 13

Drive at Enclosure 53, Slot 14
Drive port: 1, Channel: 3, ID: 107/0x2B
Drive port: 2, Channel: 4, ID: 107/0x2B
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HM3T000073409KRH
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:dc:0d
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 14

Drive at Enclosure 54, Slot 1
Drive port: 1, Channel: 4, ID: 32/0xB2
Drive port: 2, Channel: 3, ID: 32/0xB2
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K6YS00007340WVKN
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:ed:32
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 1

Drive at Enclosure 54, Slot 2
Drive port: 1, Channel: 3, ID: 33/0xB1
Drive port: 2, Channel: 4, ID: 33/0xB1
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HPGC00007339EG7M
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:fe:5a
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 2

Drive at Enclosure 54, Slot 3
Drive port: 1, Channel: 4, ID: 34/0xAE
Drive port: 2, Channel: 3, ID: 34/0xAE
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954

Serial number: 3HX0HGVA00007340X4QC
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e5:17
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 3

Drive at Enclosure 54, Slot 4

Drive port: 1, Channel: 3, ID: 35/0xAD
Drive port: 2, Channel: 4, ID: 35/0xAD
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HBVE00007339TY96
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:02:84
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 4

Drive at Enclosure 54, Slot 5

Drive port: 1, Channel: 4, ID: 36/0xAC
Drive port: 2, Channel: 3, ID: 36/0xAC
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HLXZ000073409K10
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:dc:d9
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 5

Drive at Enclosure 54, Slot 6

Drive port: 1, Channel: 3, ID: 37/0xAB
Drive port: 2, Channel: 4, ID: 37/0xAB
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K89B00007340X503
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e5:b5
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 6

Drive at Enclosure 54, Slot 7

Drive port: 1, Channel: 4, ID: 38/0xAA

Drive port: 2, Channel: 3, ID: 38/0xAA
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HND4000073409KRW
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:dc:27
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 7

Drive at Enclosure 54, Slot 8

Drive port: 1, Channel: 3, ID: 39/0xA9
Drive port: 2, Channel: 4, ID: 39/0xA9
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HNB00007339RGFJ
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e4:f0
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 8

Drive at Enclosure 54, Slot 9

Drive port: 1, Channel: 4, ID: 84/0x51
Drive port: 2, Channel: 3, ID: 84/0x51
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HT6L00007339RFZD
Vendor: IBM-ESXS
Date of manufacture: April 4, 2003
World-wide name: 20:00:00:04:cf:f9:e4:95
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 9

Drive at Enclosure 54, Slot 10

Drive port: 1, Channel: 3, ID: 100/0x34
Drive port: 2, Channel: 4, ID: 100/0x34
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0J60V00007339EG3V
Vendor: IBM-ESXS
Date of manufacture: April 4, 2003

World-wide name: 20:00:00:04:cf:f9:e4:42
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 10

Drive at Enclosure 54, Slot 11
Drive port: 1, Channel: 4, ID: 68/0x6C
Drive port: 2, Channel: 3, ID: 68/0x6C
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HRK20000734058GJ
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e8:09
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 11

Drive at Enclosure 54, Slot 12
Drive port: 1, Channel: 3, ID: 76/0x5C
Drive port: 2, Channel: 4, ID: 76/0x5C
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HRP8000073392DTP
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:f9:e8:64
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 12

Drive at Enclosure 54, Slot 13
Drive port: 1, Channel: 4, ID: 92/0x46
Drive port: 2, Channel: 3, ID: 92/0x46
Drive path redundancy: OK
Status: Optimal
Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0HEYQ00007339SNVJ
Vendor: IBM-ESXS
Date of manufacture: April 5, 2003
World-wide name: 20:00:00:04:cf:ff:02:6a
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 13

Drive at Enclosure 54, Slot 14
Drive port: 1, Channel: 3, ID: 108/0x2A
Drive port: 2, Channel: 4, ID: 108/0x2A
Drive path redundancy: OK
Status: Optimal

Raw capacity: 33.902 GB
Usable capacity: 33.402 GB
Current data rate: 2 Gbps
Product ID: ST336753FC F
Firmware version: B954
Serial number: 3HX0K2AM00007339GMLP
Vendor: IBM-ESXS
Date of manufacture: April 6, 2003
World-wide name: 20:00:00:04:cf:f9:fb:12
Drive type: Fibre Channel
Speed: 15015 RPM
Mode: Assigned
Associated array: 14

DRIVE CHANNEL INFORMATION - Channel 1

SUMMARY-----

Channel 1 status: Optimal
Controller A link status: Up
Controller B link status: Up

CUMULATIVE ERROR COUNTS-----

Controller A Cumulative Error Counts
Baseline time set: 4/19/05 3:47:13 PM
Sample period (days, hh:mm:ss): 30 days, 17:06:51
Controller detected errors: 1
Drive detected errors: 0
Timeout errors: 0
Link down errors: 6
Total I/O count: 4095773
Controller B Cumulative Error Counts
Baseline time set: 4/19/05 3:47:13 PM
Sample period (days, hh:mm:ss): 30 days, 17:06:51
Controller detected errors: 0
Drive detected errors: 0
Timeout errors: 0
Link down errors: 6
Total I/O count: 1002686153

DRIVE CHANNEL INFORMATION - Channel 2

SUMMARY-----

Channel 2 status: Optimal
Controller A link status: Up
Controller B link status: Up

CUMULATIVE ERROR COUNTS-----

Controller A Cumulative Error Counts
Baseline time set: 4/19/05 3:47:15 PM
Sample period (days, hh:mm:ss): 30 days, 17:06:49
Controller detected errors: 0
Drive detected errors: 0
Timeout errors: 1
Link down errors: 6
Total I/O count: 1345982891
Controller B Cumulative Error Counts
Baseline time set: 4/19/05 3:47:15 PM
Sample period (days, hh:mm:ss): 30 days, 17:06:49
Controller detected errors: 0
Drive detected errors: 0
Timeout errors: 0
Link down errors: 6
Total I/O count: 3816758

DRIVE CHANNEL INFORMATION - Channel 3

SUMMARY-----

Channel 3 status: Optimal

Controller A link status: Up
Controller B link status: Up

CUMULATIVE ERROR COUNTS-----

Controller A Cumulative Error Counts
Baseline time set: 4/19/05 3:47:15 PM
Sample period (days, hh:mm:ss): 30 days, 17:06:49
Controller detected errors: 0
Drive detected errors: 0
Timeout errors: 0
Link down errors: 2
Total I/O count: 4030849
Controller B Cumulative Error Counts
Baseline time set: 4/19/05 3:47:15 PM
Sample period (days, hh:mm:ss): 30 days, 17:06:49
Controller detected errors: 0
Drive detected errors: 0
Timeout errors: 0
Link down errors: 2
Total I/O count: 994104729

DRIVE CHANNEL INFORMATION - Channel 4

SUMMARY-----

Channel 4 status: Optimal
Controller A link status: Up
Controller B link status: Up

CUMULATIVE ERROR COUNTS-----

Controller A Cumulative Error Counts
Baseline time set: 4/19/05 3:47:15 PM
Sample period (days, hh:mm:ss): 30 days, 17:06:49
Controller detected errors: 0
Drive detected errors: 0
Timeout errors: 0
Link down errors: 2
Total I/O count: 1336127113
Controller B Cumulative Error Counts
Baseline time set: 4/19/05 3:47:15 PM
Sample period (days, hh:mm:ss): 30 days, 17:06:49
Controller detected errors: 0
Drive detected errors: 0
Timeout errors: 0
Link down errors: 2
Total I/O count: 3931690

Rack 4

Rack 4 is identical to Rack 3.

Client Configuration

Microsoft Windows 2000 Client System Information Report

Following is the system information report for Client 1. The system information report for clients 2-6 are identical to this one.

System Information report written at: 05/24/2005 04:45:57 PM

[System Information]

[Following are sub-categories of this main category]

[System Summary]

Item	Value
OS Name	Microsoft Windows 2000 Server
Version	5.0.2195 Service Pack 4 Build 2195
OS Manufacturer	Microsoft Corporation
System Name	VCLIENT30
System Manufacturer	IBM
System Model	IBM eServer x226-[86482BU]-
System Type	X86-based PC
Processor	x86 Family 15 Model 4 Stepping 3 GenuineIntel ~3400 Mhz
Processor	x86 Family 15 Model 4 Stepping 3 GenuineIntel ~3400 Mhz
Processor	x86 Family 15 Model 4 Stepping 3 GenuineIntel ~3400 Mhz
Processor	x86 Family 15 Model 4 Stepping 3 GenuineIntel ~3400 Mhz
BIOS Version	PhoenixBIOS 4.0 Release 6.1.U
Windows Directory	C:\WINNT
System Directory	C:\WINNT\system32
Boot Device	\Device\Harddisk0\Partition1
Locale	United States
User Name	VCLIENT30\Administrator
Time Zone	Eastern Daylight Time
Total Physical Memory	2,620,316 KB
Available Physical Memory	2,206,464 KB
Total Virtual Memory	7,176,076 KB
Available Virtual Memory	6,531,352 KB
Page File Space	4,555,760 KB
Page File	C:\pagefile.sys

[Hardware Resources]

[Following are sub-categories of this main category]

[Conflicts/Sharing]

Resource	Device	0x4000-0x4FFF	Intel(R) 6700PXH PCI Express-to-PCI Bridge B - 032A
		OK	
IRQ 16	Intel(R) E7525/E7520/E7320 PCI Express Root Port A0 - 3595	0x4000-0x4FFF	Intel(R) PRO/1000 MT Dual Port Server Adapter #5
		OK	
IRQ 16	Broadcom NetXtreme Gigabit Ethernet	0x4040-0x407F	Intel(R) PRO/1000 MT Dual Port Server Adapter #6
		OK	
IRQ 16	Intel(R) E7525/E7520/E7320 PCI Express Root Port A1 - 3596	0x1400-0x141F	Intel(R) 82801EB USB Universal Host Controller - 24D2
		OK	
IRQ 16	Intel(R) E7525/E7520 PCI Express Root Port B0 - 3597	0x1420-0x143F	Intel(R) 82801EB USB Universal Host Controller - 24D4
		OK	
IRQ 16	Intel(R) 82801EB USB Universal Host Controller - 24D2	0x1440-0x145F	Intel(R) 82801EB USB Universal Host Controller - 24D7
		OK	
IRQ 16	Intel(R) 82801EB USB Universal Host Controller - 24DE	0x1460-0x147F	Intel(R) 82801EB USB Universal Host Controller - 24DE
		OK	
[DMA]			
Channel	Device	Status	
4	Direct memory access controller	OK	0x5000-0x50FF RADEON 7000M (on board) OK
2	Standard floppy disk controller	OK	0x03B0-0x03BB RADEON 7000M (on board) OK
3	ECP Printer Port (LPT1)	OK	0x03C0-0x03DF RADEON 7000M (on board) OK
			0x0A79-0x0A79 ISAPNP Read Data Port OK
			0x0279-0x0279 ISAPNP Read Data Port OK
			0x0274-0x0277 ISAPNP Read Data Port OK
[Forced Hardware]			
Device	PNP Device ID		0x0010-0x001F Motherboard resources OK
No Forced Hardware			0x0024-0x0025 Motherboard resources OK
			0x0028-0x0029 Motherboard resources OK
			0x002C-0x002D Motherboard resources OK
			0x002E-0x002F Motherboard resources OK
			0x0030-0x0031 Motherboard resources OK
			0x0034-0x0035 Motherboard resources OK
			0x0038-0x0039 Motherboard resources OK
			0x003C-0x003D Motherboard resources OK
			0x0050-0x0053 Motherboard resources OK
			0x0072-0x0077 Motherboard resources OK
			0x0080-0x0080 Motherboard resources OK
			0x0090-0x009F Motherboard resources OK
			0x00A4-0x00A5 Motherboard resources OK
			0x00A8-0x00A9 Motherboard resources OK
			0x00AC-0x00AD Motherboard resources OK
			0x00B0-0x00B5 Motherboard resources OK
			0x00B8-0x00B9 Motherboard resources OK
Address Range	Device	Status	
0x0000-0x0CF7	PCI bus	OK	
0x0000-0x0CF7	Direct memory access controller	OK	
0x0D00-0xFFFF	PCI bus	OK	
0x2000-0x4FFF - 3596	Intel(R) E7525/E7520/E7320 PCI Express Root Port A1	OK	
0x2000-0x4FFF	Intel(R) 6700PXH PCI Express-to-PCI Bridge A - 0329	OK	
0x2000-0x4FFF	Adaptec AIC-7902B - Ultra320 SCSI	OK	
0x2400-0x24FF	Adaptec AIC-7902B - Ultra320 SCSI	OK	
0x2C00-0x2CFF	Adaptec AIC-7902B - Ultra320 SCSI	OK	
0x2800-0x28FF	Adaptec AIC-7902B - Ultra320 SCSI	OK	

0x00BC-0x00BD	Motherboard resources	OK
0x04D0-0x04D1	Motherboard resources	OK
0x1000-0x107F	Motherboard resources	OK
0x1180-0x118C	Motherboard resources	OK
0x118D-0x118D	Motherboard resources	OK
0x118F-0x118F	Motherboard resources	OK
0x1190-0x11BF	Motherboard resources	OK
0x0800-0x082F	Motherboard resources	OK
0xFE00-0xFE00	Motherboard resources	OK
0x0081-0x008F	Direct memory access controller	OK
0x00C0-0x00DF	Direct memory access controller	OK
0x00F0-0x00FE	Numeric data processor	OK
0x0020-0x0021	Programmable interrupt controller	OK
0x00A0-0x00A1	Programmable interrupt controller	OK
0x0070-0x0071	System CMOS/real time clock	OK
0x0061-0x0061	System speaker	OK
0x0040-0x0043	System timer	OK
0x118E-0x118E	Not Available	OK
0x03F0-0x03F5	Standard floppy disk controller	OK
0x03F7-0x03F7	Standard floppy disk controller	OK
0x0378-0x037F	ECP Printer Port (LPT1)	OK
0x0778-0x077F	ECP Printer Port (LPT1)	OK
0x03F8-0x03FF	Communications Port (COM1)	OK
0x02F8-0x02FF	Communications Port (COM2)	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
0x14A0-0x14AF	Intel(R) 82801EB Ultra ATA Storage Controllers	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
0x1100-0x111F	Intel(R) 82801EB SMBus Controller - 24D3	OK

[IRQs]

IRQ Number	Device
9	Microsoft ACPI-Compliant System
16	Intel(R) E7525/E7520/E7320 PCI Express Root Port A0 - 3595
16	Broadcom NetXtreme Gigabit Ethernet
16	Intel(R) E7525/E7520/E7320 PCI Express Root Port A1 - 3596
16	Intel(R) E7525/E7520 PCI Express Root Port B0 - 3597
16	Intel(R) 82801EB USB Universal Host Controller - 24D2
16	Intel(R) 82801EB USB Universal Host Controller - 24DE
30	Adaptec AIC-7902B - Ultra320 SCSI
31	Adaptec AIC-7902B - Ultra320 SCSI
48	Intel(R) PRO/1000 MT Dual Port Server Adapter #5
49	Intel(R) PRO/1000 MT Dual Port Server Adapter #6
19	Intel(R) 82801EB USB Universal Host Controller - 24D4
18	Intel(R) 82801EB USB Universal Host Controller - 24D7
23	Intel(R) 82801EB USB2 Enhanced Host Controller - 24DD
22	RADEON 7000M (on board)
13	Numeric data processor
8	System CMOS/real time clock
6	Standard floppy disk controller
4	Communications Port (COM1)
3	Communications Port (COM2)
12	PS/2 Compatible Mouse
1	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard
14	Primary IDE Channel
15	Secondary IDE Channel
10	Intel(R) 82801EB SMBus Controller - 24D3

[Memory]

Range	Device	Status
0xA0000-0xBFFFF	PCI bus	OK

0xA0000-0xBFFFF RADEON 7000M (on board) OK

0xD8000-0xDBFFF PCI bus OK

0xDC000-0xDFFFF PCI bus OK

0xA0000000-0xFEBFFFFF PCI bus OK

0xD0100000-0xD01FFFFF Intel(R) E7525/E7520/E7320 PCI Express Root Port A0 - 3595 OK

0xD0100000-0xD01FFFFF Broadcom NetXtreme Gigabit Ethernet OK

0xD0200000-0xD05FFFFF Intel(R) E7525/E7520/E7320 PCI Express Root Port A1 - 3596 OK

0xD0200000-0xD05FFFFF Intel(R) 6700PXH I/OxAPIC Interrupt Controller A - 0326 OK

0xD0300000-0xD04FFFFF Intel(R) 6700PXH PCI Express-to-PCI Bridge A - 0329 OK

0xD0300000-0xD04FFFFF Adaptec AIC-7902B - Ultra320 SCSI OK

0xD0302000-0xD0303FFF Adaptec AIC-7902B - Ultra320 SCSI OK

0xD0500000-0xD05FFFFF Intel(R) 6700PXH PCI Express-to-PCI Bridge B - 032A OK

0xD0500000-0xD05FFFFF Intel(R) PRO/1000 MT Dual Port Server Adapter #5OK

0xD0540000-0xD055FFFF Intel(R) PRO/1000 MT Dual Port Server Adapter #5OK

0xD0560000-0xD057FFFF Intel(R) PRO/1000 MT Dual Port Server Adapter #6OK

0xD0201000-0xD0201FFF Intel(R) 6700PXH I/OxAPIC Interrupt Controller B - 0327 OK

0xD0000000-0xD00003FF Intel(R) 82801EB USB2 Enhanced Host Controller - 24DD OK

0xD8000000-0xDFFFFFFF RADEON 7000M (on board) OK

0xD0600000-0xD060FFFF RADEON 7000M (on board) OK

0xE0000000-0xEFFFFFFF Motherboard resources OK

0xFEE00000-0xFEE0FFFF Motherboard resources OK

0xFEC81000-0xFEC81FFF Motherboard resources OK

0xFEC81400-0xFEC823FF Motherboard resources OK

0xFEBFFC00-0xFEBFFFFF Intel(R) 82801EB Ultra ATA Storage Controllers OK

[Following are sub-categories of this main category]

[Multimedia]

[Following are sub-categories of this main category]

[Audio Codecs]

Codec Version	Manufacturer Size	Description Creation Date	Status	File
c:\winnt\system32\iac25_32.ax	Intel Corporation	Indeo® audio software	OK	C:\WINNT\system32\IAC25_32.AX 2.05.53 195.00 KB (199,680 bytes) 12/7/1999 7:00:00 AM
c:\winnt\system32\msg723.acm	Microsoft Corporation	4.4.3385	OK	C:\WINNT\system32\MSG723.ACM 106.77 KB (109,328 bytes) 11/3/2004 1:58:46 PM
c:\winnt\system32\tssoft32.acm	DSP GROUP, INC.	OK	C:\WINNT\system32\TSOFT32.ACM 1.01	9.27 KB (9,488 bytes) 12/7/1999 7:00:00 AM
c:\winnt\system32\msadp32.acm	Microsoft Corporation	5.00.2134.1	OK	C:\WINNT\system32\MSADP32.ACM 14.77 KB (15,120 bytes) 12/7/1999 7:00:00 AM
c:\winnt\system32\imaadp32.acm	Microsoft Corporation	5.00.2195.6612	OK	C:\WINNT\system32\IMAADP32.ACM 16.27 KB (16,656 bytes) 11/3/2004 3:02:37 PM
c:\winnt\system32\lhacm.acm	Microsoft Corporation	4.4.3385	OK	C:\WINNT\system32\LHACM.ACM 33.27 KB (34,064 bytes) 11/3/2004 1:58:47 PM
c:\winnt\system32\msg711.acm	Microsoft Corporation	5.00.2134.1	OK	C:\WINNT\system32\MSG711.ACM 10.27 KB (10,512 bytes) 12/7/1999 7:00:00 AM
c:\winnt\system32\msgsm32.acm	Microsoft Corporation	5.00.2134.1	OK	C:\WINNT\system32\MSGSM32.ACM 22.27 KB (22,800 bytes) 12/7/1999 7:00:00 AM

[Video Codecs]

Codec Version	Manufacturer Size	Description Creation Date	Status	File
c:\winnt\system32\ir50_32.dll	Intel Corporation	Indeo® video 5.10	OK	C:\WINNT\system32\IR50_32.DLL R.5.10.15.2.55 737.50 KB (755,200 bytes) 12/7/1999 7:00:00 AM
c:\winnt\system32\msh261.drv	Microsoft Corporation	4.4.3385	OK	C:\WINNT\system32\MSH261.DRV 163.77 KB (167,696 bytes) 11/3/2004 1:58:46 PM

[Components]

c:\winnt\system32\msh263.drv Microsoft Corporation
 OK C:\WINNT\system32\MSH263.DRV 4.4.3385 252.27 KB
 (258,320 bytes) 11/3/2004 1:58:27 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 7:00:00 AM

c:\winnt\system32\iccvid.dll Radius Inc. OK
 C:\WINNT\system32\ICCVID.DLL 1.10.0.6 108.00 KB (110,592
 bytes) 12/7/1999 7:00:00 AM

c:\winnt\system32\msrle32.dll Microsoft Corporation
 OK C:\WINNT\system32\MSRLE32.DLL 5.00.2195.6612
 10.77 KB (11,024 bytes) 11/3/2004 3:02:45 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

[CD-ROM]

Item	Value
Drive	D:
Description	CD-ROM Drive
Media Loaded	False
Media Type	CD-ROM
Name	HL-DT-ST CD-ROM GCR-8482B
Manufacturer	(Standard CD-ROM drives)
Status	OK
Transfer Rate	Not Available
SCSI Target ID	0
PNP Device ID	IDE\CDROMHL-DT-ST_CD-ROM_GCR-8482B_1.04_5&25B98AF5&0&0.0.0

[Sound Device]

Item	Value
No sound devices	

[Display]

Item	Value
Name	RADEON 7000M (on board)
PNP Device ID	PCI\VEN_1002&DEV_5159&SUBSYS_02C81014&REV_004&3A321F38&0&20F0
Adapter Type	RADEON 7000 (0x5159), ATI Technologies Inc. compatible
Adapter Description	RADEON 7000M (on board)
Adapter RAM	16.00 MB (16,777,216 bytes)
Installed Drivers	ati2dvag.dll
Driver Version	5.2.3790.2
INF File	oem6.inf (ati2mtag_RV100 section)
Color Planes	1
Color Table Entries	65536
Resolution	1024 x 768 x 85 hertz
Bits/Pixel	16
[Infrared]	
Item	Value
No infrared devices	
[Input]	
[Following are sub-categories of this main category]	
[Keyboard]	
Item	Value
Description	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard
Name	Enhanced (101- or 102-key)
Layout	00000409
PNP Device ID	ACPI\PNP0303\5&9583612&0
NumberOfFunctionKeys	12

[Pointing Device]

Item	Value
Hardware Type	PS/2 Compatible Mouse
Number of Buttons	3
Status	OK
PNP Device ID	ACPI\PNP0F13\5&9583612&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	5/24/2005 8:25:43 AM
Index	0
Service Name	AsynMac

IP AddressNot 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 5/24/2005 8:25:43 AM

Index 1

Service Name Rasl2tp

IP AddressNot Available

IP Subnet Not Available

Default IP Gateway Not Available

DHCP Enabled False

DHCP Server Not Available

DHCP Lease Expires Not Available

DHCP Lease Obtained Not Available

MAC Address Not Available

Service Name Rasl2tp

Driver c:\winnt\system32\drivers\rasl2tp.sys (52112, 5.00.2195.6655)

Name [00000002] WAN Miniport (PPTP)

Adapter Type Wide Area Network (WAN)

Product Name WAN Miniport (PPTP)

Installed True

PNP Device ID ROOT\MS_PPTPMINIPORT\0000

Last Reset 5/24/2005 8:25:43 AM

Index 2

Service Name PptpMiniport

IP Address Not Available

IP Subnet Not Available

Default IP Gateway Not Available

DHCP Enabled False

DHCP Server Not Available

DHCP Lease Expires Not Available

DHCP Lease Obtained Not Available

MAC Address 50:50:54:50:30:30

Service Name PptpMiniport

Driver c:\winnt\system32\drivers\rasptp.sys (48464, 5.00.2195.6711)

Name [00000003] Direct Parallel

Adapter Type Not Available

Product Name Direct Parallel

Installed True

PNP Device ID ROOT\MS_PTMINIPORT\0000

Last Reset 5/24/2005 8:25:43 AM

Index 3

Service Name Raspti

IP Address Not Available

IP Subnet Not Available

Default IP Gateway Not Available

DHCP Enabled False

DHCP Server Not Available

DHCP Lease Expires Not Available

DHCP Lease Obtained Not Available

MAC Address Not Available

Service Name Raspti

Driver c:\winnt\system32\drivers\raspti.sys (16880, 5.00.2146.1)

Name [00000004] WAN Miniport (IP)

Adapter Type Not Available

Product Name WAN Miniport (IP)

Installed True

PNP Device ID ROOT\MS_NDISWANIP\0000

Last Reset 5/24/2005 8:25:43 AM

Index 4

Service Name NdisWan

IP Address Not Available

IP Subnet Not Available

Default IP Gateway Not Available

DHCP Enabled False

DHCP Server Not Available

DHCP Lease Expires Not Available

DHCP Lease Obtained Not Available

MAC Address Not Available

Service Name NdisWan

Driver c:\winnt\system32\drivers\ndiswan.sys (93360, 5.00.2195.6699)

Name [00000009] Intel(R) PRO/1000 MT Dual Port Server Adapter

Adapter Type Ethernet 802.3

Product Name Intel(R) PRO/1000 MT Dual Port Server Adapter

Installed True

PNP Device ID PCI\VEN_8086&DEV_1079&SUBSYS_117A8086&REV_03\5&A4D5A19&0&080218

Last Reset 5/24/2005 8:25:43 AM

Index 9

Service Name E1000

IP Address 192.168.30.99

IP Subnet 255.255.255.0

Default IP Gateway Not Available

DHCP Enabled False

DHCP Server Not Available

DHCP Lease Expires Not Available

DHCP Lease Obtained Not Available

MAC Address 00:0E:0C:36:79:D6
Service Name E1000
IRQ Number 48
I/O Port 0x4000-0x4FFF
Driver c:\winnt\system32\drivers\e1000nt5.sys (170496, 8.4.21.0 built by: WinDDK)

Name [00000010] Intel(R) PRO/1000 MT Dual Port Server Adapter
Adapter Type Ethernet 802.3
Product Name Intel(R) PRO/1000 MT Dual Port Server Adapter
Installed True

PNP Device ID
PCI\VEN_8086&DEV_1079&SUBSYS_117A8086&REV_03\5&A4D5A19&0&090218

Last Reset 5/24/2005 8:25:43 AM

Index 10

Service Name E1000

IP Address 192.168.31.99

IP Subnet 255.255.255.0

Default IP Gateway Not Available

DHCP Enabled False

DHCP Server Not Available

DHCP Lease Expires Not Available

DHCP Lease Obtained Not Available

MAC Address 00:0E:0C:36:79:D7

Service Name E1000

IRQ Number 49

I/O Port 0x4040-0x407F

Driver c:\winnt\system32\drivers\e1000nt5.sys (170496, 8.4.21.0 built by: WinDDK)

Name [00000011] Broadcom NetXtreme Gigabit Ethernet

Adapter Type Ethernet 802.3

Product Name Broadcom NetXtreme Gigabit Ethernet

Installed True

PNP Device ID
PCI\VEN_14E4&DEV_1659&SUBSYS_02C61014&REV_01\4&1855300&0&0010

Last Reset 5/24/2005 8:25:43 AM

Index 11

Service Name b57w2k

IP Address 10.0.0.30

IP Subnet 255.0.0.0

Default IP Gateway Not Available

DHCP Enabled False

DHCP Server Not Available

DHCP Lease Expires Not Available

DHCP Lease Obtained Not Available

MAC Address 00:0D:60:15:1A:AB

Service Name b57w2k

IRQ Number 16

Driver c:\winnt\system32\drivers\b57w2k.sys (192215, 7.80.0.0)

[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	SupportsDisconnectData	False
SupportsGracefulClosing	True	SupportsEncryption	True
SupportsGuaranteedBandwidth	False	SupportsExpeditedData	False
SupportsMulticasting	False	SupportsGracefulClosing	False
		SupportsGuaranteedBandwidth	False
Name	MSAFD Tcip [UDP/IP]	SupportsMulticasting	True
ConnectionlessService	True		
GuaranteesDelivery	False	Name	RSVP TCP Service Provider
GuaranteesSequencing	False	ConnectionlessService	False
MaximumAddressSize	16 bytes	GuaranteesDelivery	True
MaximumMessageSize	65467 bytes	GuaranteesSequencing	True
MessageOriented	True	MaximumAddressSize	16 bytes
MinimumAddressSize	16 bytes	MaximumMessageSize	0 bytes
PseudoStreamOriented	False	MessageOriented	False
SupportsBroadcasting	True	MinimumAddressSize	16 bytes
SupportsConnectData	False	PseudoStreamOriented	False
SupportsDisconnectData	False	SupportsBroadcasting	False
SupportsEncryption	False	SupportsConnectData	False
SupportsExpeditedData	False	SupportsDisconnectData	False
SupportsGracefulClosing	False	SupportsEncryption	True
SupportsGuaranteedBandwidth	False	SupportsExpeditedData	True
SupportsMulticasting	True	SupportsGracefulClosing	True
		SupportsGuaranteedBandwidth	False
Name	RSVP UDP Service Provider	SupportsMulticasting	False
ConnectionlessService	True		
GuaranteesDelivery	False	Name	MSAFD NetBIOS [\\Device\\NetBT_Tcpip_{0A116012-8A70-4813-A60F-1179D8F7AE88}] SEQPACKET 8
GuaranteesSequencing	False	ConnectionlessService	False
MaximumAddressSize	16 bytes	GuaranteesDelivery	True
MaximumMessageSize	65467 bytes	GuaranteesSequencing	True
MessageOriented	True	MaximumAddressSize	20 bytes
MinimumAddressSize	16 bytes	MaximumMessageSize	64000 bytes
PseudoStreamOriented	False	MessageOriented	True
SupportsBroadcasting	True	MinimumAddressSize	20 bytes
SupportsConnectData	False		

PseudoStreamOriented	False
SupportsBroadcasting	False
SupportsConnectData	False
SupportsDisconnectData	False
SupportsEncryption	False
SupportsExpeditedData	False
SupportsGracefulClosing	False
SupportsGuaranteedBandwidth	False
SupportsMulticasting	False
Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{0A116012-8A70-4813-A60F-1179D8F7AE88}] DATAGRAM 8	
ConnectionlessService	True
GuaranteesDelivery	False
GuaranteesSequencing	False
MaximumAddressSize	20 bytes
MaximumMessageSize	64000 bytes
MessageOriented	True
MinimumAddressSize	20 bytes
PseudoStreamOriented	False
SupportsBroadcasting	True
SupportsConnectData	False
SupportsDisconnectData	False
SupportsEncryption	False
SupportsExpeditedData	False
SupportsGracefulClosing	False
SupportsGuaranteedBandwidth	False
SupportsMulticasting	False
Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{29FD1170-E61B-4B40-A9F0-4BCEDEDBF8A3}] 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_{29FD1170-E61B-4B40-A9F0-4BCEDEDBF8A3}] 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_{49CD30FF-A43F-4429-B4C6-DC5F2B3F2E53}]
SEQPACKET 6

ConnectionlessService False
GuaranteesDelivery True
GuaranteesSequencing True
MaximumAddressSize 20 bytes
MaximumMessageSize 64000 bytes
MessageOriented True
MinimumAddressSize20 bytes
PseudoStreamOriented False
SupportsBroadcasting False
SupportsConnectData False
SupportsDisconnectData False
SupportsEncryption False
SupportsExpeditedData False
SupportsGracefulClosing False
SupportsGuaranteedBandwidth False
SupportsMulticasting False

Name MSAFD NetBIOS
[\Device\NetBT_Tcpip_{49CD30FF-A43F-4429-B4C6-DC5F2B3F2E53}]
DATAGRAM 6

ConnectionlessService True
GuaranteesDelivery False
GuaranteesSequencing False
MaximumAddressSize 20 bytes
MaximumMessageSize 64000 bytes
MessageOriented True
MinimumAddressSize20 bytes
PseudoStreamOriented False
SupportsBroadcasting True
SupportsConnectData False
SupportsDisconnectData False
SupportsEncryption False
SupportsExpeditedData False

SupportsGracefulClosing False
SupportsGuaranteedBandwidth False
SupportsMulticasting False

Name MSAFD NetBIOS
[\Device\NetBT_Tcpip_{19854BA0-8D07-490E-BE50-A95157216BD2}]
SEQPACKET 1

ConnectionlessService False
GuaranteesDelivery True
GuaranteesSequencing True
MaximumAddressSize 20 bytes
MaximumMessageSize 64000 bytes
MessageOriented True
MinimumAddressSize20 bytes
PseudoStreamOriented False
SupportsBroadcasting False
SupportsConnectData False
SupportsDisconnectData False
SupportsEncryption False
SupportsExpeditedData False
SupportsGracefulClosing False
SupportsGuaranteedBandwidth False
SupportsMulticasting False

Name MSAFD NetBIOS
[\Device\NetBT_Tcpip_{19854BA0-8D07-490E-BE50-A95157216BD2}]
DATAGRAM 1

ConnectionlessService True
GuaranteesDelivery False
GuaranteesSequencing False
MaximumAddressSize 20 bytes
MaximumMessageSize 64000 bytes
MessageOriented True
MinimumAddressSize20 bytes
PseudoStreamOriented False
SupportsBroadcasting True

SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS
 [\Device\NetBT_Tcpip_{E71C5B67-467D-4AAA-BFFB-3896F017D737}]
 SEQPACKET 2

ConnectionlessService False
 GuaranteesDelivery True
 GuaranteesSequencing True
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting False
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS
 [\Device\NetBT_Tcpip_{E71C5B67-467D-4AAA-BFFB-3896F017D737}]
 DATAGRAM 2

ConnectionlessService True
 GuaranteesDelivery False
 GuaranteesSequencing False
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes

MessageOriented True
 MinimumAddressSize20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting True
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

[WinSock]

Item	Value
File	c:\winnt\system32\winsock.dll
Version	3.10
Size	2.80 KB (2,864 bytes)
File	c:\winnt\system32\wsock32.dll
Version	5.00.2195.6603
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	XOn Character	17
Maximum Input Buffer Size	0	XOnXMit Threshold	2048
Maximum Output Buffer Size	False	XOnXOff InFlow Control	0
Settable Baud Rate	True	XOnXOff OutFlow Control	0
Settable Data Bits	True	IRQ Number	4
Settable Flow Control	True	I/O Port	0x03F8-0x03FF
Settable Parity	True	Driver	c:\winnt\system32\drivers\serial.sys (62736, 5.00.2195.6655)
Settable Parity Check	True	Name	COM2
Settable Stop Bits	True	Status	OK
Settable RLSD	True	PNP Device ID	ACPI\PNP0501\2
Supports RLSD	True	Maximum Input Buffer Size	0
Supports 16 Bit Mode	False	Maximum Output Buffer Size	False
Supports Special Characters	False	Settable Baud Rate	True
Baud Rate	9600	Settable Data Bits	True
Bits/Byte	8	Settable Flow Control	True
Stop Bits	1	Settable Parity	True
Parity	None	Settable Parity Check	True
Busy	0	Settable Stop Bits	True
Abort Read/Write on Error	0	Settable RLSD	True
Binary Mode Enabled	-1	Supports RLSD	True
Continue XMit on XOff	0	Supports 16 Bit Mode	False
CTS Outflow Control	0	Supports Special Characters	False
Discard NULL Bytes	0	Baud Rate	9600
DSR Outflow Control	0	Bits/Byte	8
DSR Sensitivity	0	Stop Bits	1
DTR Flow Control Type	Enable	Parity	None
EOF Character	0	Busy	0
Error Replace Character	0	Abort Read/Write on Error	0
Error Replacement Enabled	0	Binary Mode Enabled	-1
Event Character	0	Continue XMit on XOff	0
Parity Check Enabled	0	CTS Outflow Control	0
RTS Flow Control Type	Enable	Discard NULL Bytes	0
XOff Character	19	DSR Outflow Control	0
XOffXMit Threshold	512		

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 (62736, 5.00.2195.6655)

[Parallel]

Item	Value
Name	LPT1
PNP Device ID	ACPI\PNP0401\5&9583612&0

[Storage]

[Following are sub-categories of this main category]

[Drives]

Item	Value
Drive	A:

Description 3 1/2 Inch Floppy Drive
Drive C:
Description Local Fixed Disk
Compressed False
File System NTFS
Size 33.90 GB (36,396,830,720 bytes)
Free Space 28.35 GB (30,439,546,880 bytes)
Volume Name
Volume Serial Number FC4C7C1B
Partition Disk #0, Partition #0
Partition Size 33.90 GB (36,396,831,744 bytes)
Starting Offset 32256 bytes
Drive Description Disk drive
Drive Manufacturer (Standard disk drives)
Drive Model IBM-ESXS ST336607LC FN SCSI Disk Device
Drive BytesPerSector 512
Drive MediaLoaded True
Drive MediaType Fixed hard disk media
Drive Partitions 1
Drive SCSI Bus 0
Drive SCSI LogicalUnit 0
Drive SCSI Port 3
Drive SCSI TargetId 0
Drive SectorsPerTrack 63
Drive Size 36396864000 bytes
Drive TotalCylinders 4425
Drive TotalSectors 71087625
Drive TotalTracks 1128375
Drive TracksPerCylinder 255

[SCSI]

Item Value

Name Adaptec AIC-7902B - Ultra320 SCSI

Caption Adaptec AIC-7902B - Ultra320 SCSI

Driver adpu320

Status OK

PNP Device ID
PCI\VEN_9005&DEV_801D&SUBSYS_02CC1014&REV_10\5&1EA3B137
&0&180018

Device ID
PCI\VEN_9005&DEV_801D&SUBSYS_02CC1014&REV_10\5&1EA3B137
&0&180018

Device Map Not Available

Index Not Available

Max Number Controlled Not Available

IRQ Number 30

I/O Port 0x2400-0x24FF

I/O Port 0x2000-0x4FFF

Driver c:\winnt\system32\drivers\adpu320.sys (132608, 3.0.000.000 built
by: WinDDK)

Name Adaptec AIC-7902B - Ultra320 SCSI

Caption Adaptec AIC-7902B - Ultra320 SCSI

Driver adpu320

Status OK

PNP Device ID
PCI\VEN_9005&DEV_801D&SUBSYS_02CC1014&REV_10\5&1EA3B137
&0&190018

Device ID
PCI\VEN_9005&DEV_801D&SUBSYS_02CC1014&REV_10\5&1EA3B137
&0&190018

Device Map Not Available

Index Not Available

Max Number Controlled Not Available

IRQ Number 31

I/O Port 0x2C00-0x2CFF

I/O Port 0x2800-0x28FF

Driver c:\winnt\system32\drivers\adpu320.sys (132608, 3.0.000.000 built
by: WinDDK)

[Printing]

Name Port Name Server Name

No printing information

[Problem Devices]

Device	PNP Device ID	Error Code
Not Available	ACPI\ASF0001\2&DABA3FF&0	28
Not Available	ACPI\IBM3737\4&369939D9&0	28

[USB]

Device PNP Device ID

Intel(R) 82801EB USB Universal Host Controller - 24D2
PCI\VEN_8086&DEV_24D2&SUBSYS_02ED1014&REV_02\3&61AAA01&
&0&E8

USB Root Hub USB\ROOT_HUB\4&39460DFB&0

Intel(R) 82801EB USB Universal Host Controller - 24D4
PCI\VEN_8086&DEV_24D4&SUBSYS_02ED1014&REV_02\3&61AAA01&
&0&E9

USB Root Hub USB\ROOT_HUB\4&31D97CBA&0

Intel(R) 82801EB USB Universal Host Controller - 24D7
PCI\VEN_8086&DEV_24D7&SUBSYS_02ED1014&REV_02\3&61AAA01&
&0&EA

USB Root Hub USB\ROOT_HUB\4&206D9F09&0

Intel(R) 82801EB USB Universal Host Controller - 24DE
PCI\VEN_8086&DEV_24DE&SUBSYS_02ED1014&REV_02\3&61AAA01
&0&EB

USB Root Hub USB\ROOT_HUB\4&CAEE98E&0

Intel(R) 82801EB USB2 Enhanced Host Controller - 24DD
PCI\VEN_8086&DEV_24DD&SUBSYS_02ED1014&REV_02\3&61AAA01
&0&EF

USB 2.0 Root Hub USB\ROOT_HUB20\4&34B50607&0

[Software Environment]

[Following are sub-categories of this main category]

[Drivers]

Name	Description	File	Type	Started	Start Mode
State	Status	Error Control	Accept Pause	Accept Stop	Accept Stop
abiosdsk	Abiosdsk	Not Available	Kernel Driver	False	False
Disabled	Stopped	OK Ignore	False	False	False
abp480n5	abp480n5	Not Available	Kernel Driver	False	False
Disabled	Stopped	OK Normal	False	False	False
acpi	Microsoft ACPI Driver		c:\winnt\system32\drivers\acpi.sys	Running	OK Normal
Kernel Driver	True	Boot	Running	OK	Normal
False	True				
acpiec	ACPIEC	c:\winnt\system32\drivers\acpiec.sys	Kernel Driver	False	False
Driver	False	Disabled Stopped	OK	Normal	False
False					
adpu160m	adpu160m	Not Available	Kernel Driver	False	False
Disabled	Stopped	OK Normal	False	False	False
adpu320	adpu320	c:\winnt\system32\drivers\adpu320.sys	Kernel Driver	False	False
Driver	True	Boot Running	OK	Normal	False
True					
afd	AFD Networking Support Environment		c:\winnt\system32\drivers\afd.sys	Kernel Driver	True Auto
Running	OK	Normal False	True	True	Auto
aha154x	Aha154x	Not Available	Kernel Driver	False	False
Disabled	Stopped	OK Normal	False	False	False
aic116x	aic116x	Not Available	Kernel Driver	False	False
Disabled	Stopped	OK Normal	False	False	False
aic78u2	aic78u2	Not Available	Kernel Driver	False	False
Disabled	Stopped	OK Normal	False	False	False
aic78xx	aic78xx	Not Available	Kernel Driver	False	False
Disabled	Stopped	OK Normal	False	False	False
ami0nt	ami0nt	Not Available	Kernel Driver	False	False
Disabled	Stopped	OK Normal	False	False	False
amsint	amsint	Not Available	Kernel Driver	False	False
Disabled	Stopped	OK Normal	False	False	False
asc	asc	Not Available	Kernel Driver	False	False
Disabled	Stopped	OK Normal	False	False	False
asc3350p	asc3350p	Not Available	Kernel Driver	False	False
Disabled	Stopped	OK Normal	False	False	False
asc3550	asc3550	Not Available	Kernel Driver	False	False
Disabled	Stopped	OK Normal	False	False	False
asynmac	RAS Asynchronous Media Driver		c:\winnt\system32\drivers\asynmac.sys	Kernel Driver	False
Manual	Stopped	OK Normal	False	False	False
atapi	Standard IDE/ESDI Hard Disk Controller		c:\winnt\system32\drivers\atapi.sys	Kernel Driver	True
Boot	Running	OK Normal	False	True	True

atdisk	Atdisk	Not Available	Kernel Driver	False	False
Disabled	Stopped	OK Ignore	False	False	False
ati2mtag	ati2mtag	c:\winnt\system32\drivers\ati2mtag.sys	Kernel Driver	False	False
Driver	True	Manual Running	OK	Ignore	False
True					
atmarpc	ATM ARP Client Protocol		c:\winnt\system32\drivers\atmarpc.sys	Kernel Driver	False
Manual	Stopped	OK Normal	False	False	False
audstub	Audio Stub Driver		c:\winnt\system32\drivers\audstub.sys	Kernel Driver	Normal
False	True	True	Manual Running	OK	Normal
b57w2k	Broadcom NetXtreme Gigabit Ethernet		c:\winnt\system32\drivers\b57w2k.sys	Kernel Driver	True
Manual	Running	OK Normal	False	True	True
beep	Beep		c:\winnt\system32\drivers\beep.sys	Kernel Driver	False
Driver	True	System Running	OK	Normal	False
True					
buslogic	BusLogic	Not Available	Kernel Driver	False	False
Disabled	Stopped	OK Normal	False	False	False
cd20xrnt	cd20xrnt	Not Available	Kernel Driver	False	False
Disabled	Stopped	OK Normal	False	False	False
cdaudio	Cdaudio		c:\winnt\system32\drivers\cdaudio.sys	Kernel Driver	False
Driver	False	System Stopped	OK	Ignore	False
False					
cdfs	Cdfs		c:\winnt\system32\drivers\cdfs.sys	File System	False
Driver	True	Disabled Running	OK	Normal	False
True					
cdrom	CD-ROM Driver		c:\winnt\system32\drivers\cdrom.sys	Kernel Driver	Normal
False	True	True	System Running	OK	Normal
changer	Changer	Not Available	Kernel Driver	False	False
System	Stopped	OK Ignore	False	False	False
cpqarray	Cpqarray	Not Available	Kernel Driver	False	False
Disabled	Stopped	OK Normal	False	False	False
cpqarray2	cpqarray2	Not Available	Kernel Driver	False	False
Disabled	Stopped	OK Normal	False	False	False
cpqfcalm	cpqfcalm	Not Available	Kernel Driver	False	False
Disabled	Stopped	OK Normal	False	False	False
cpqfws2e	cpqfws2e	Not Available	Kernel Driver	False	False
Disabled	Stopped	OK Normal	False	False	False
dac960nt	dac960nt	Not Available	Kernel Driver	False	False
Disabled	Stopped	OK Normal	False	False	False
deckzpsx	deckzpsx	Not Available	Kernel Driver	False	False
Disabled	Stopped	OK Normal	False	False	False
dfsdriver	DfsDriver		c:\winnt\system32\drivers\dfs.sys	File System Driver	True
True	Boot	Running	OK	Normal	False
					True

disk	Disk Driver	c:\winnt\system32\drivers\disk.sys					ini910u	ini910u	Not Available	Kernel Driver	False	
Kernel Driver	True	Boot	Running	OK	Normal	Disabled	Stopped	OK	Normal	False		
False	True											
diskperf	Diskperf	c:\winnt\system32\drivers\diskperf.sys					intelide	Intellde	Not Available	Kernel Driver	False	
Driver	True	Boot	Running	OK	Normal	Disabled	Stopped	OK	Normal	False		
True												
dmboot	dmboot	c:\winnt\system32\drivers\dmboot.sys					ipfilterdriver	IP Traffic Filter Driver				
Driver	False	Disabled	Stopped	OK	Normal	Manual	Stopped	OK	Normal	False		
False												
dmio	Logical Disk Manager Driver					ipinip	IP in IP Tunnel Driver:c:\winnt\system32\drivers\ipinip.sys					
c:\winnt\system32\drivers\dmio.sys		Kernel Driver			True	Kernel Driver	False	Manual	Stopped	OK	Normal	
Boot	Running	OK	Normal	False	True	False	False					
dmload	dmload	c:\winnt\system32\drivers\dmload.sys					ipnat	IP Network Address Translator c:\winnt\system32\drivers\ipnat.sys				
Driver	True	Boot	Running	OK	Normal	Kernel Driver	False	Manual	Stopped	OK	Normal	
True												
e1000	Intel(R) PRO/1000 Network Connection Driver					ipsecc	IPSEC driver c:\winnt\system32\drivers\ipsecc.sys					
c:\winnt\system32\drivers\e1000nt5.sys		Kernel Driver			True	Kernel Driver	True	Manual	Running	OK	Normal	
Manual	Running	OK	Normal	False	True	False	True					
efs	EFS	c:\winnt\system32\drivers\efs.sys File System Driver					ipsraidn	ipsraidn	Not Available	Kernel Driver	False	
True	Disabled	Running	OK	Normal	False	True	Stopped	OK	Normal	False		
fastfat	Fastfat	c:\winnt\system32\drivers\fastfat.sys					irenum	IR Enumerator Service				
Driver	True	Disabled	Running	OK	Normal	File System	Manual	Stopped	OK	Normal	False	
True												
fd16_700	Fd16_700	Not Available			Kernel Driver	False	isapnp	PnP ISA/EISA Bus Driver				
Disabled	Stopped	OK	Normal	False	False	True	c:\winnt\system32\drivers\isapnp.sys	Kernel Driver	True	True		
fdc	Floppy Disk Controller Driver					kbdclass	Keyboard Class Driver					
Kernel Driver	True	Manual	Running	OK	Normal	c:\winnt\system32\drivers\kbdclass.sys	Kernel Driver	True	True	True		
False	True											
fips	Fips	c:\winnt\system32\drivers\fips.sys					ksecdd	KSecDD	c:\winnt\system32\drivers\ksecdd.sys	Kernel	False	
Driver	True	Auto	Running	OK	Normal	False	Driver	True	Boot	Running	OK	Normal
True												
fireport	fireport	Not Available			Kernel Driver	False	lbtrfdc	lbtrfdc	Not Available	Kernel Driver	False	
Disabled	Stopped	OK	Normal	False	False	System	Stopped	OK	Ignore	False		
flashpnt	flashpnt	Not Available			Kernel Driver	False	lp6nds35	lp6nds35	Not Available	Kernel Driver	False	
Disabled	Stopped	OK	Normal	False	False	Disabled	Stopped	OK	Normal	False		
flpydisk	Floppy Disk Driver					mnmd	mnmdd c:\winnt\system32\drivers\mnmd.sys					
Kernel Driver	True	Manual	Running	OK	Normal	Driver	True	System	Running	OK	Ignore	
False	True											
ftdisk	Volume Manager Driver					modem	Modem c:\winnt\system32\drivers\modem.sys					
c:\winnt\system32\drivers\ftdisk.sys		Kernel Driver			True	Driver	False	Manual	Stopped	OK	Ignore	
Boot	Running	OK	Normal	False	True	False						
gpc	Generic Packet Classifier					mouclass	Mouse Class Driver c:\winnt\system32\drivers\mouclass.sys					
c:\winnt\system32\drivers\msgpc.sys		Kernel Driver			True	Kernel Driver	True	System	Running	OK	Normal	
Manual	Running	OK	Normal	False	True	False	True					
i8042prt	i8042 Keyboard and PS/2 Mouse Port Driver					mountmgr	MountMgr c:\winnt\system32\drivers\mountmgr.sys					
c:\winnt\system32\drivers\i8042prt.sys		Kernel Driver			True	Driver	True	Boot	Running	OK	Normal	
System	Running	OK	Normal	False	True	True						
ibmfe	IBM 10/100 Ethernet PCI Adapter NT Driver					mraid35x	mraid35x	Not Available	Kernel Driver	False		
c:\winnt\system32\drivers\ibmfent5.sys		Kernel Driver			False	Disabled	Stopped	OK	Normal	False		
Manual	Stopped	OK	Normal	False	False							

mrxsmb Driver True	MRXSMB True	c:\winnt\system32\drivers\mrxsmb.sys System Running	OK Normal	File System False	null Driver True	Null True	c:\winnt\system32\drivers\null.sys System Running	OK Normal	Kernel False
msfs Driver True	Msfs True	c:\winnt\system32\drivers\msfs.sys System Running	OK Normal	File System False	nwlnkflt Manual	IPX Traffic Filter Driver Stopped	c:\winnt\system32\drivers\nwlnkflt.sys OK Normal	Kernel Driver False	False
mkserv Manual	Microsoft Streaming Service Proxy Stopped	c:\winnt\system32\drivers\mkserv.sys OK Normal	Kernel Driver False	False	nwlnkfld Manual	IPX Traffic Forwarder Driver Stopped	c:\winnt\system32\drivers\nwlnkfld.sys OK Normal	Kernel Driver False	False
mspclock Manual	Microsoft Streaming Clock Proxy Stopped	c:\winnt\system32\drivers\mspclock.sys OK Normal	Kernel Driver False	False	parallel Kernel Driver	Parallel class driver True	c:\winnt\system32\drivers\parallel.sys Manual Running	OK Normal	False
mispqm Manual	Microsoft Streaming Quality Manager Proxy Stopped	c:\winnt\system32\drivers\mispqm.sys OK Normal	Kernel Driver False	False	parport Kernel Driver	Parallel port driver True	c:\winnt\system32\drivers\parport.sys System Running	OK Ignore	False
mup Driver True	Mup True	c:\winnt\system32\drivers\mup.sys Boot Running	OK Normal	File System False	partmgr Driver True	PartMgr True	c:\winnt\system32\drivers\partmgr.sys Boot Running	OK Normal	Kernel False
nrc710 Disabled	Nrc710 Stopped	Not Available OK Normal	Kernel Driver False	False	parvdm Driver True	ParVdm True	c:\winnt\system32\drivers\parvdm.sys Auto Running	OK Ignore	Kernel False
ndis Kernel Driver False	NDIS System Driver True	c:\winnt\system32\drivers\ndis.sys Boot Running	OK Normal	Kernel Driver False	pci Driver True	PCI Bus Driver True	c:\winnt\system32\drivers\pci.sys Boot Running	OK Critical	Kernel False
ndistapi Manual	Remote Access NDIS TAPI Driver Running	c:\winnt\system32\drivers\ndistapi.sys OK Normal	Kernel Driver False	True	pcidump System	PCIDump Stopped	Not Available OK Ignore	Kernel Driver False	False
ndisuio Manual	NDIS Usermode I/O Protocol Stopped	c:\winnt\system32\drivers\ndisuio.sys OK Normal	Kernel Driver False	False	pciide Driver True	PCIIde True	c:\winnt\system32\drivers\pciide.sys Boot Running	OK Normal	Kernel False
ndiswan Manual	Remote Access NDIS WAN Driver Running	c:\winnt\system32\drivers\ndiswan.sys OK Normal	Kernel Driver False	True	pcmcia Driver False	Pcmcia False	c:\winnt\system32\drivers\pcmcia.sys Disabled Stopped	OK Normal	Kernel False
ndproxy Kernel Driver False	NDIS Proxy True	c:\winnt\system32\drivers\ndproxy.sys Manual Running	OK Normal	Kernel Driver False	pdcomp Manual	PDCOMP Stopped	Not Available OK Ignore	Kernel Driver False	False
netbios File System False	NetBIOS Interface True	c:\winnt\system32\drivers\netbios.sys System Running	OK Normal	Kernel Driver False	pdframe False	PDFRAME Manual	Not Available OK Ignore	Kernel Driver False	False
netbt Kernel Driver False	NetBios over Tcpip True	c:\winnt\system32\drivers\netbt.sys System Running	OK Normal	Kernel Driver False	pdreli Manual	PDRELI Stopped	Not Available OK Ignore	Kernel Driver False	False
netdetect Driver False	NetDetect False	c:\winnt\system32\drivers\netdetect.sys Manual Stopped	OK Normal	Kernel False	pdrframe False	PDRFRAME Manual	Not Available OK Ignore	Kernel Driver False	False
npfs Driver True	Npfs True	c:\winnt\system32\drivers\npfs.sys System Running	OK Normal	File System False	pptpminiport Manual	WAN Miniport (PPTP) Running	c:\winnt\system32\drivers\raspptp.sys OK Normal	Kernel Driver False	True
ntfs Driver True	Ntfs True	c:\winnt\system32\drivers\ntfs.sys Disabled Running	OK Normal	File System False	ptilink Manual	Direct Parallel Link Driver Running	c:\winnt\system32\drivers\ptilink.sys OK Normal	Kernel Driver False	True
					ql1080 Disabled	ql1080 Stopped	Not Available OK Normal	Kernel Driver False	False
					ql10wnt Disabled	Ql10wnt Stopped	Not Available OK Normal	Kernel Driver False	False

ql1240	ql1240	Not Available	Kernel Driver	False	swenum	Software Bus Driver	c:\winnt\system32\drivers\swenum.sys	Kernel Driver	True	Manual	Running	OK	Normal	
Disabled	Stopped	OK	Normal	False	False	True								
ql2100	ql2100	Not Available	Kernel Driver	False	symc810	symc810	Not Available	Kernel Driver	False	Normal	Stopped	OK	Normal	
Disabled	Stopped	OK	Normal	False	Disabled	Stopped	OK	Normal	False	False			False	
rasacd	Remote Access Auto Connection Driver				symc8xx	symc8xx	Not Available	Kernel Driver	False	Normal	Stopped	OK	Normal	
c:\winnt\system32\drivers\rasacd.sys	Kernel Driver	True			Disabled	Stopped	OK	Normal	False	False			False	
System	Running	OK	Normal	False	True									
rasl2tp	WAN Miniport (L2TP)				sym_hi	sym_hi	Not Available	Kernel Driver	False	Normal	Stopped	OK	Normal	
c:\winnt\system32\drivers\rasl2tp.sys	Kernel Driver	True			Disabled	Stopped	OK	Normal	False	False			False	
Manual	Running	OK	Normal	False	True									
raspti	Direct Parallel				tcpip	TCP/IP Protocol Driver		c:\winnt\system32\drivers\tcpip.sys	Kernel Driver	True	System	Running	OK	Normal
Kernel Driver	True	Manual	Running	OK	Normal	False	True		False	True				
False	True				tdasync	TDASYNC		c:\winnt\system32\drivers\tdasync.sys	Kernel Driver	False	Manual	Stopped	OK	Ignore
					False	False			False	False				
rca	Microsoft Streaming Network Raw Channel Access				tdipx	TDIPX		c:\winnt\system32\drivers\tdipx.sys	Kernel Driver	Manual	Stopped	OK	Ignore	Kernel
c:\winnt\system32\drivers\rca.sys	Kernel Driver	False	Manual		Driver	False	Manual	Stopped	OK	Ignore	False		False	
Stopped	OK	Normal	False	False	False									
rdbss	Rdbss				tdnetb	TDNETB		c:\winnt\system32\drivers\tdnetb.sys	Kernel Driver	Manual	Stopped	OK	Ignore	Kernel
Driver	True	System	Running	OK	Normal	False			False	Manual	Stopped	OK	Ignore	False
True					False									
rdpdr	Terminal Server Device Redirector Driver				tdpipe	TDPIPE		c:\winnt\system32\drivers\tdpipe.sys	Kernel Driver	Manual	Stopped	OK	Ignore	Kernel
c:\winnt\system32\drivers\rdpdr.sys	Kernel Driver	True			Driver	False	Manual	Stopped	OK	Ignore	False		False	
Manual	Running	OK	Normal	False	False									
rdpwd	RDPWD				tdspx	TDSPX		c:\winnt\system32\drivers\tdspx.sys	Kernel Driver	Manual	Stopped	OK	Ignore	Kernel
Driver	False	Manual	Stopped	OK	Ignore	False			False	Manual	Stopped	OK	Ignore	False
False					False									
redbook	Digital CD Audio Playback Filter Driver				tdtcp	TDTCP		c:\winnt\system32\drivers\tdtcp.sys	Kernel Driver	Manual	Stopped	OK	Ignore	Kernel
c:\winnt\system32\drivers\redbook.sys	Kernel Driver	False			Driver	False	Manual	Stopped	OK	Ignore	False		False	
System	Stopped	OK	Normal	False	False									
serenum	Serenum Filter Driver				termdd	Terminal Device Driver		c:\winnt\system32\drivers\termdd.sys	Kernel Driver	Auto	Running	OK	Normal	True
Kernel Driver	True	Manual	Running	OK	Normal	False	True		False	True				
False	True				Auto	Running	OK	Normal	False	True				
serial	Serial port driver				tga	tga	Not Available	Kernel Driver	False	Normal	Stopped	OK	Normal	False
Kernel Driver	True	System	Running	OK	Ignore	False			False	False				
False	True				System	Stopped	OK	Ignore	False	False				
sfloppy	Sfloppy				udfs	Udfs		c:\winnt\system32\drivers\udfs.sys	File System	False	Disabled	Stopped	OK	Normal
Driver	False	System	Stopped	OK	Ignore	False			False	False				
False					False									
sglfb	sglfb	Not Available	Kernel Driver	False	uhcd	Microsoft USB Universal Host Controller Driver		c:\winnt\system32\drivers\uhcd.sys	Kernel Driver	Manual	Running	OK	Normal	True
System	Stopped	OK	Normal	False	False				False	True				
Stopped	OK	Normal	False	False	Manual	Running	OK	Normal	False	True				
simbad	Simbad	Not Available	Kernel Driver	False	ultra66	ultra66	Not Available	Kernel Driver	False	Normal	Stopped	OK	Normal	False
Disabled	Stopped	OK	Normal	False	Disabled	Stopped	OK	Normal	False	False				
Stopped	OK	Normal	False	False	System	Stopped	OK	Ignore	False	False				
sparrow	Sparrow	Not Available	Kernel Driver	False	update	Microcode Update Driver		c:\winnt\system32\drivers\update.sys	Kernel Driver	Manual	Running	OK	Normal	True
Disabled	Stopped	OK	Normal	False	False				False	True				
Stopped	OK	Normal	False	False	Manual	Running	OK	Normal	False	True				
spud	Special Purpose Utility Driver				usbehci	Microsoft USB 2.0 Enhanced Host Controller Miniport Driver		c:\winnt\system32\drivers\usbehci.sys	Kernel Driver	Manual	Running	OK	Normal	True
Kernel Driver	True	Manual	Running	OK	Normal	False	True		False	True				
False	True				Manual	Running	OK	Normal	False	True				
srv	Srv				Manual	Running	OK	Normal	False	True				
True	Manual	Running	OK	Normal	False	True								
True	Manual	Running	OK	Normal	False	True								

```

usbhub  Microsoft USB Standard Hub Driver
c:\winnt\system32\drivers\usbhub.sys  Kernel Driver  True
Manual  Running  OK      Normal  False  True

usbhub20 USB 2.0 Root Hub Support
c:\winnt\system32\drivers\usbhub20.sys  Kernel Driver  True
Manual  Running  OK      Normal  False  True

vgasave  VgaSave  c:\winnt\system32\drivers\vga.sys  Kernel
Driver  True  System  Running  OK  Ignore  False
True

wanarp  Remote Access IP ARP Driver
c:\winnt\system32\drivers\wanarp.sys  Kernel Driver  True
Manual  Running  OK      Normal  False  True

wdica  WDICA  Not Available  Kernel Driver  False
Manual  Stopped  OK  Ignore  False  False

nal  Nal Service  \??c:\winnt\system32\drivers\iqvw32.sys  Kernel Driver  True  Disabled  Running  OK  Normal
False  True

```

[Environment Variables]

Variable	Value	User Name
----------	-------	-----------

```

CLASSPATH
.;C:\SQLLIB\java\db2java.zip;C:\SQLLIB\java\db2jcc.jar;C:\SQLLIB\java\sqlj.zip;C:\SQLLIB\java\db2jcc_license_cu.jar;C:\SQLLIB\bin;C:\SQLLIB\java\common.jar <SYSTEM>

ComSpec  %SystemRoot%\system32\cmd.exe  <SYSTEM>

DB2INSTANCE  DB2  <SYSTEM>

DB2TEMPDIR  C:\SQLLIB\  <SYSTEM>

INCLUDE C:\SQLLIB\INCLUDE;C:\SQLLIB\LIB  <SYSTEM>

LIB  C:\SQLLIB\LIB  <SYSTEM>

NUMBER_OF_PROCESSORS  4  <SYSTEM>

OS  Windows_NT  <SYSTEM>

Os2LibPath  %SystemRoot%\system32\os2\dll;  <SYSTEM>

Path
C:\Perl\bin\;%SystemRoot%\system32;%SystemRoot%;%SystemRoot%\System32\Wbem;C:\SQLLIB\BIN;C:\SQLLIB\FUNCTION;c:\tools;c:\tools\util;C:\Program Files\Intel\DMIX  <SYSTEM>

PATHEXT
.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH  <SYSTEM>

PROCESSOR_ARCHITECTURE  x86  <SYSTEM>

PROCESSOR_IDENTIFIER  x86 Family 15 Model 4 Stepping 3, GenuineIntel  <SYSTEM>

PROCESSOR_LEVEL  15  <SYSTEM>

PROCESSOR_REVISION  0403  <SYSTEM>

```

```

TEMP  %SystemRoot%\TEMP  <SYSTEM>

TMP  %SystemRoot%\TEMP  <SYSTEM>

windir  %SystemRoot%  <SYSTEM>

include  C:\Program Files\Microsoft Visual Studio\VC98\at\include;C:\Program Files\Microsoft Visual Studio\VC98\mf\include;C:\Program Files\Microsoft Visual Studio\VC98\include  VCLIENT30\Administrator

lib  C:\Program Files\Microsoft Visual Studio\VC98\mf\lib;C:\Program Files\Microsoft Visual Studio\VC98\lib  VCLIENT30\Administrator

MSDevDirC:\Program Files\Microsoft Visual Studio\Common\MSDev98 VCLIENT30\Administrator

path  C:\Program Files\Microsoft Visual Studio\Common\Tools\WinNT;C:\Program Files\Microsoft Visual Studio\Common\MSDev98\Bin;C:\Program Files\Microsoft Visual Studio\Common\Tools;C:\Program Files\Microsoft Visual Studio\VC98\bin  VCLIENT30\Administrator

```

```
TEMP  %USERPROFILE%\Local Settings\Temp VCLIENT30\Administrator
```

```
TMP  %USERPROFILE%\Local Settings\Temp VCLIENT30\Administrator
```

[Jobs]

[Following are sub-categories of this main category]

[Print]

Document Size	Owner	Notify	Status	Time Submitted
Start Time	Until Time	Elapsed Time	Pages Printed	Job ID
Priority	Parameters	Driver Name	Print Processor	Host Print Queue
	Data Type	Name		
Unknown	Unknown	Unknown	Unknown	Unknown
Unknown	Unknown	Unknown	Unknown	Unknown
Unknown	Unknown	Unknown	Unknown	Unknown

[Network Connections]

Local Name	Remote Name	Type	Status	User Name
No network connections information				

[Running Tasks]

Name	Path	Process ID	Priority	Min Working Set	Max
Working Set	Start Time	Version	Size	File Date	
system idle process	Not Available	Not Available	0	0	Not Available
system	Not Available	Unknown	8	8	0
smss.exe	c:\winnt\system32\smss.exe	1413120	5/24/2005 12:26:04 PM	196	11
(45,840 bytes)			12/7/1999 7:00:00 AM	5.00.2195.6601	204800
csrss.exe	c:\winnt\system32\csrss.exe	1413120	5/24/2005 12:26:07 PM	220	13
(5,392 bytes)			11/3/2004 3:02:31 PM	5.00.2195.6601	204800
winlogon.exe	c:\winnt\system32\winlogon.exe	204800	5/24/2005 12:26:09 PM	216	13
(176,77 KB (181,008 bytes))			11/3/2004 3:03:01 PM	5.00.2195.6714	
services.exe	c:\winnt\system32\services.exe	204800	5/24/2005 12:26:10 PM	268	9
(87,27 KB (89,360 bytes))			12/7/1999 7:00:00 AM	5.00.2195.6700	
lsass.exe	c:\winnt\system32\lsass.exe	204800	5/24/2005 12:26:10 PM	280	9
(33,552 bytes)			12/7/1999 7:00:00 AM	5.00.2195.6695	204800
svchost.exe	c:\winnt\system32\svchost.exe	204800	5/24/2005 12:26:13 PM	472	8
(7,77 KB (7,952 bytes))			12/7/1999 7:00:00 AM	5.00.2134.1	
msdtc.exe	c:\winnt\system32\msdtc.exe	1413120	5/24/2005 12:26:13 PM	504	8
(6,928 bytes)			11/3/2004 8:55:08 AM	1999.9.3421.3	204800
db2jds.exe	c:\sql\lib\bin\db2jds.exe	1413120	5/24/2005 12:26:16 PM	676	8
(193,12 KB (197,752 bytes))			6/17/2004 11:30:56 PM	8.1.6.574	204800
db2sec.exe	c:\sql\lib\bin\db2sec.exe	1413120	5/24/2005 12:26:17 PM	692	8
(29,808 bytes)			6/17/2004 11:32:48 PM	8.1.6.574	204800
svchost.exe	c:\winnt\system32\svchost.exe	204800	5/24/2005 12:26:17 PM	708	8
(7,77 KB (7,952 bytes))			12/7/1999 7:00:00 AM	5.00.2134.1	
llssrv.exe	c:\winnt\system32\llssrv.exe	1413120	5/24/2005 12:26:18 PM	744	9
(83,728 bytes)			6/19/2003 1:05:04 PM	5.00.2195.6697	204800
regsvc.exe	c:\winnt\system32\regsvc.exe	1413120	5/24/2005 12:26:19 PM	796	8
(68,368 bytes)			11/3/2004 3:02:53 PM	5.00.2195.6701	204800
mstask.exe	c:\winnt\system32\mstask.exe	1413120	5/24/2005 12:26:19 PM	888	8
(119,568 bytes)			11/3/2004 3:02:46 PM	4.71.2195.6704	204800
explorer.exe	c:\winnt\explorer.exe	1413120	5/24/2005 12:26:24 PM	1008	8
(243,472 bytes)			11/3/2004 3:03:02 PM	5.00.3700.6690	204800

tcpsvcs.exe	c:\winnt\system32\tcpsvcs.exe	204800	5/24/2005 12:26:28 PM	1120	8
(24,77 KB (25,360 bytes))			12/7/1999 7:00:00 AM	5.00.2134.1	
wingmgt.exe	c:\winnt\system32\wbem\wingmgt.exe	204800	5/24/2005 12:26:28 PM	1152	8
(192,10 KB (196,706 bytes))			11/3/2004 3:03:07 PM	1.50.1085.0100	
inetinfo.exe	c:\winnt\system32\inetrv\inetinfo.exe	204800	5/24/2005 12:26:28 PM	1168	8
(14,608 bytes)			11/3/2004 3:03:22 PM	5.00.0984	14.27 KB
dfssvc.exe	c:\winnt\system32\dfssvc.exe	1413120	5/24/2005 12:27:28 PM	1032	8
(90,896 bytes)			11/3/2004 3:02:32 PM	5.00.2195.6664	204800
svchost.exe	c:\winnt\system32\svchost.exe	204800	5/24/2005 12:27:32 PM	1468	8
(7,77 KB (7,952 bytes))			12/7/1999 7:00:00 AM	5.00.2134.1	
ieexplore.exe	c:\program files\internet explorer\ieexplore.exe	204800	5/24/2005 1:07:12 PM	868	8
(59,27 KB (60,688 bytes))			11/3/2004 1:58:28 PM	5.00.2920.0000	
dllhost.exe	c:\winnt\system32\dllhost.exe	1413120	5/24/2005 1:07:13 PM	1496	8
(5,904 bytes)			11/3/2004 3:02:32 PM	5.00.2195.6692	204800
dllhost.exe	c:\winnt\system32\dllhost.exe	1413120	5/24/2005 1:07:14 PM	1552	8
(5,904 bytes)			11/3/2004 3:02:32 PM	5.00.2195.6692	204800
mdm.exe	c:\winnt\system32\mdm.exe	1413120	5/24/2005 1:07:15 PM	1612	8
(121,29 KB (124,200 bytes))			11/3/2004 8:56:48 AM	6.00.8424	204800
mmc.exe	c:\winnt\system32\mmc.exe	1413120	5/24/2005 4:39:14 PM	2076	8
(603,408 bytes)			11/3/2004 3:02:41 PM	5.00.2195.6601	204800
rsvp.exe	c:\winnt\system32\rsvp.exe	1413120	5/24/2005 4:45:01 PM	2232	8
(176,912 bytes)			11/3/2004 3:02:54 PM	5.00.2195.6663	204800

[Loaded Modules]

Name	Version	Size	File Date	Manufacturer	Path
traffic.dll	5.00.2195.6613	30.77 KB (31,504 bytes)	3:02:58 PM	Microsoft Corporation	c:\winnt\system32\traffic.dll
rsvp.exe	5.00.2195.6663	172.77 KB (176,912 bytes)	3:02:54 PM	Microsoft Corporation	c:\winnt\system32\rsvp.exe
bttagresenu.dll	9.2.4.5	20.00 KB (20,480 bytes)	2:20:07 PM	Intel(R) Corporation	c:\program files\intel\dmix\resource\bttagresenu.dll
btagsrv.dll	9.2.4.5	96.00 KB (98,304 bytes)	2:20:07 PM	Intel(R) Corporation	c:\program files\intel\dmix\btagsrv.dll

teamresenu.dll	9.2.4.4	172.00 KB (176,128 bytes)	3/14/2005	2:20:07 PM	Intel(R) Corporation	c:\program files\intel\dmix\resource\teamresenu.dll
teamsrv.dll	9.2.4.4	256.00 KB (262,144 bytes)	3/14/2005	2:20:07 PM	Intel(R) Corporation	c:\program files\intel\dmix\teamsrv.dll
brandres.dll	9.2.4.7	12.00 KB (12,288 bytes)	3/14/2005	2:20:07 PM	Intel(R) Corporation	c:\program files\intel\dmix\resource\brandres.dll
etcoinst.dll	2.1.3.0 built by: WinDDK	54.50 KB (55,808 bytes)	11/16/2004	4:35:46 PM	Intel Corporation	c:\winnt\system32\etcoinst.dll
intelnic.dll	8.2.3.0 built by: WinDDK	19.00 KB (19,456 bytes)	10/29/2004	5:01:48 PM	Intel(R) Corporation	c:\winnt\system32\intelnic.dll
vlanresenu.dll	9.2.4.4	20.00 KB (20,480 bytes)	3/14/2005	2:20:07 PM	Intel(R) Corporation	c:\program files\intel\dmix\resource\vlanresenu.dll
vlanrv.dll	9.2.4.4	156.00 KB (159,744 bytes)	3/14/2005	2:20:07 PM	Intel(R) Corporation	c:\program files\intel\dmix\vlanrv.dll
wbemprox.dll	1.50.1085.0100	40.10 KB (41,061 bytes)	11/3/2004	3:03:07 PM	Microsoft Corporation	c:\winnt\system32\wbem\wbemprox.dll
dmixresenu.dll	9.2.4.7	40.00 KB (40,960 bytes)	3/14/2005	2:20:07 PM	Intel(R) Corporation	c:\program files\intel\dmix\resource\dmixresenu.dll
msvcp60.dll	6.00.8972.0	392.05 KB (401,462 bytes)	6/17/1998	1:00:00 AM	Microsoft Corporation	c:\winnt\system32\msvcp60.dll
ncs2instutility.dll	9.2.4.0	124.00 KB (126,976 bytes)	3/14/2005	2:20:07 PM	Intel(R) Corporation	c:\winnt\system32\ncs2instutility.dll
acesor.dll	9.2.4.4	284.00 KB (290,816 bytes)	3/14/2005	2:20:07 PM	Intel(R) Corporation	c:\winnt\system32\acesor.dll
ncs2dmix.dll	9.2.4.7	376.00 KB (385,024 bytes)	3/14/2005	2:20:07 PM	Intel(R) Corporation	c:\winnt\system32\ncs2dmix.dll
olepro32.dll	5.0.4522	160.27 KB (164,112 bytes)	11/3/2004	3:02:51 PM	Microsoft Corporation	c:\winnt\system32\olepro32.dll
dmocx.dll	5.00.2134.1	23.27 KB (23,824 bytes)	12/7/1999	7:00:00 AM	Microsoft Corporation	c:\winnt\system32\dmocx.dll
rassapi.dll	5.00.2195.6604	14.27 KB (14,608 bytes)	11/3/2004	3:02:53 PM	Microsoft Corporation	c:\winnt\system32\rassapi.dll
adsnt.dll	5.00.2195.6658	196.77 KB (201,488 bytes)	11/3/2004	3:02:26 PM	Microsoft Corporation	c:\winnt\system32\adsnt.dll
dbghelp.dll	5.00.2195.6613	159.27 KB (163,088 bytes)	6/19/2003	1:05:04 PM	Microsoft Corporation	c:\winnt\system32\dbghelp.dll
localsec.dll	5.00.2195.6623	240.27 KB (246,032 bytes)	11/3/2004	3:02:40 PM	Microsoft Corporation	c:\winnt\system32\localsec.dll
devmgr.dll	5.00.2195.6619	216.77 KB (221,968 bytes)	11/3/2004	3:02:32 PM	Microsoft Corporation	c:\winnt\system32\devmgr.dll
filemgmt.dll	5.00.2195.6601	287.77 KB (294,672 bytes)	11/3/2004	3:02:35 PM	Microsoft Corporation	c:\winnt\system32\filemgmt.dll
pdh.dll	5.00.2195.6660	148.27 KB (151,824 bytes)	11/3/2004	3:02:51 PM	Microsoft Corporation	c:\winnt\system32\pdh.dll
smlogcfg.dll	5.00.2195.6612	278.77 KB (285,456 bytes)	11/3/2004	3:02:56 PM	Microsoft Corporation	c:\winnt\system32\smlogcfg.dll
cabinet.dll	5.00.2147.1	54.77 KB (56,080 bytes)	12/7/1999	7:00:00 AM	Microsoft Corporation	c:\winnt\system32\cabinet.dll
msinfo32.dll	5.00.2195.6601	312.27 KB (319,760 bytes)	11/3/2004	3:03:09 PM	Microsoft Corporation	c:\program files\common files\microsoft shared\msinfo\msinfo32.dll
riched20.dll	5.30.23.1215	421.77 KB (431,888 bytes)	11/3/2004	3:02:53 PM	Microsoft Corporation	c:\winnt\system32\riched20.dll
riched32.dll	5.00.2134.1	3.77 KB (3,856 bytes)	12/7/1999	7:00:00 AM	Microsoft Corporation	c:\winnt\system32\riched32.dll
els.dll	5.00.2195.6610	154.27 KB (157,968 bytes)	11/3/2004	3:02:34 PM	Microsoft Corporation	c:\winnt\system32\els.dll
ntmsmgr.dll	1,0,0,1	427.77 KB (438,032 bytes)	12/7/1999	7:00:00 AM	Microsoft Corporation and HighGround Systems, Inc.	c:\winnt\system32\ntmsmgr.dll
mmfutil.dll	1.50.1085.0000	32.06 KB (32,829 bytes)	12/7/1999	7:00:00 AM	Microsoft Corporation	c:\winnt\system32\mmfutil.dll
logdrive.dll	1.50.1085.0000	200.06 KB (204,863 bytes)	12/7/1999	7:00:00 AM	Microsoft Corporation	c:\winnt\system32\logdrive.dll
dfrgres.dll	5.00.2150.1	27.50 KB (28,160 bytes)	12/7/1999	7:00:00 AM	Executive Software International, Inc.	c:\winnt\system32\dfrgres.dll
dfrgsnap.dll	5.00.2195.6605	41.77 KB (42,768 bytes)	11/3/2004	3:02:32 PM	Executive Software International, Inc.	c:\winnt\system32\dfrgsnap.dll
dmdskres.dll	2195.6605.297.3	119.50 KB (122,368 bytes)	11/3/2004	3:02:33 PM	Microsoft Corp., VERITAS Software	c:\winnt\system32\dmdskres.dll
dmutil.dll	2195.6605.297.3	42.27 KB (43,280 bytes)	11/3/2004	3:02:33 PM	VERITAS Software Corp.	c:\winnt\system32\dmutil.dll

ntmsapi.dll	5.00.1948.1	52.27 KB (53,520 bytes)	11/3/2004 3:02:49 PM	Microsoft Corporation	c:\winnt\system32\ntmsapi.dll	
dmdskmgr.dll	2195.6605.297.3	159.77 KB (163,600 bytes)	11/3/2004 3:02:33 PM	Microsoft Corp., VERITAS Software	c:\winnt\system32\dmdskmgr.dll	
mycomput.dll	5.00.2195.6601	107.77 KB (110,352 bytes)	11/3/2004 3:02:47 PM	Microsoft Corporation	c:\winnt\system32\mycomput.dll	
mmcmdmgr.dll	5.00.2195.6601	816.27 KB (835,856 bytes)	11/3/2004 3:02:41 PM	Microsoft Corporation	c:\winnt\system32\mmcmdmgr.dll	
mmc.exe	5.00.2195.6601	589.27 KB (603,408 bytes)	11/3/2004 3:02:41 PM	Microsoft Corporation	c:\winnt\system32\mmc.exe	
mdm.exe	6.00.8424	121.29 KB (124,200 bytes)	11/3/2004 8:56:48 AM	Microsoft Corporation	c:\winnt\system32\mdm.exe	
txflog.dll	1999.9.3421.3	82.27 KB (84,240 bytes)	11/3/2004 8:55:04 AM	Microsoft Corporation	c:\winnt\system32\txflog.dll	
dllhost.exe	5.00.2195.6692	5.77 KB (5,904 bytes)	11/3/2004 3:02:32 PM	Microsoft Corporation	c:\winnt\system32\dllhost.exe	
sensapi.dll	5.00.2195.6627	7.27 KB (7,440 bytes)	11/3/2004 3:02:54 PM	Microsoft Corporation	c:\winnt\system32\sensapi.dll	
ieexplore.exe	5.00.2920.0000	59.27 KB (60,688 bytes)	11/3/2004 1:58:28 PM	Microsoft Corporation	c:\program files\internet explorer\ieexplore.exe	
h323.tsp	5.00.2195.6699	248.77 KB (254,736 bytes)	11/3/2004 3:02:36 PM	Microsoft Corporation	c:\winnt\system32\h323.tsp	
ipconf.tsp	5.00.2143.1	10.77 KB (11,024 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\ipconf.tsp	
ndptsp.tsp	5.00.2143.1	38.27 KB (39,184 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\ndptsp.tsp	
kmddsp.tsp	5.00.2150.1	17.77 KB (18,192 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\kmddsp.tsp	
uniplat.dll	5.00.2195.6601	14.27 KB (14,608 bytes)	11/3/2004 3:02:59 PM	Microsoft Corporation	c:\winnt\system32\uniplat.dll	
unimdm.tsp	5.00.2195.6601	199.27 KB (204,048 bytes)	11/3/2004 3:02:59 PM	Microsoft Corporation	c:\winnt\system32\unimdm.tsp	
tapisrv.dll	5.00.2195.6666	169.27 KB (173,328 bytes)	11/3/2004 3:02:58 PM	Microsoft Corporation	c:\winnt\system32\tapisrv.dll	
dfssvc.exe	5.00.2195.6664	88.77 KB (90,896 bytes)	11/3/2004 3:02:32 PM	Microsoft Corporation	c:\winnt\system32\dfssvc.exe	
tpcccom.dll	1.0.0.1	49.00 KB (50,176 bytes)	3/15/2005 10:26:03 AM	TODO: <Company name>	c:\inetpub\wwwroot\tpcc\tpcccom.dll	
db2tcp.dll	8.1.6.574	64.06 KB (65,598 bytes)	6/17/2004 8:31:06 PM	International Business Machines Corporation	c:\sql\bin\db2tcp.dll	
tpccdb2glue.dll	Not Available	160.00 KB (163,840 bytes)	3/15/2005 10:26:03 AM	Not Available	c:\inetpub\wwwroot\tpcc\tpccdb2glue.dll	
msvc71.dll	7.10.3077.0	488.00 KB (499,712 bytes)	3/15/2005 10:26:03 AM	Microsoft Corporation	c:\inetpub\wwwroot\tpcc\msvc71.dll	
msvcr71.dll	7.10.3052.4	340.00 KB (348,160 bytes)	3/15/2005 10:26:03 AM	Microsoft Corporation	c:\inetpub\wwwroot\tpcc\msvcr71.dll	
tpccisapi.dll	1.0.0.1	88.00 KB (90,112 bytes)	3/15/2005 10:26:04 AM	TODO: <Company name>	c:\inetpub\wwwroot\tpcc\tpccisapi.dll	
mfc42.dll	6.00.9586.0	992.05 KB (1,015,859 bytes)	11/3/2004 3:02:41 PM	Microsoft Corporation	c:\winnt\system32\mfc42.dll	
wam.dll	5.00.0984	70.77 KB (72,464 bytes)	11/3/2004 3:03:24 PM	Microsoft Corporation	c:\winnt\system32\inetsrv\wam.dll	
ilsdbx.dll	5.00.0984	56.27 KB (57,616 bytes)	3/15/2005 5:39:16 PM	Microsoft Corporation	c:\winnt\system32\inetsrv\ilsdbx.dll	
msrd3x40.dll	4.00.6508.0	308.27 KB (315,664 bytes)	11/3/2004 3:02:45 PM	Microsoft Corporation	c:\winnt\system32\msrd3x40.dll	
odbccp32.dll	3.520.7713.0	92.00 KB (94,208 bytes)	11/3/2004 5:27:34 PM	Microsoft Corporation	c:\winnt\system32\odbccp32.dll	
comsvcs.dll	2000.2.3504.0	1.38 MB (1,448,208 bytes)	11/3/2004 3:02:30 PM	Microsoft Corporation	c:\winnt\system32\comsvcs.dll	
mtxdm.dll	2000.2.3504.0	22.77 KB (23,312 bytes)	11/3/2004 3:02:47 PM	Microsoft Corporation	c:\winnt\system32\mtxdm.dll	
odbj32.dll	4.0.6200.0	52.27 KB (53,520 bytes)	11/3/2004 3:02:50 PM	Microsoft Corporation	c:\winnt\system32\odbj32.dll	
odbjt32.dll	4.0.6200.0	264.27 KB (270,608 bytes)	11/3/2004 3:02:50 PM	Microsoft Corporation	c:\winnt\system32\odbjt32.dll	
msdsqldr.dll	2.70.7713.0	built by: Lab06_N(dagbuild)	16.00 KB (16,384 bytes)	11/3/2004 5:27:34 PM	Microsoft Corporation	c:\program files\common files\system\ole db\msdsqldr.dll
msdatl3.dll	2.70.7713.0	built by: Lab06_N(dagbuild)	84.00 KB (86,016 bytes)	11/3/2004 5:27:34 PM	Microsoft Corporation	c:\program files\common files\system\ole db\msdatl3.dll

msdasql.dll	2.70.7713.0 built by: Lab06_N(dagbuild)	296.00 KB (303,104 bytes)	11/3/2004 5:27:34 PM	Microsoft Corporation	c:\program files\common files\system\ole db\msdasql.dll
ldapdbx.dll	5.00.0984	73.27 KB (75,024 bytes)	3/15/2005 5:39:16 PM	Microsoft Corporation	c:\winnt\system32\inetsrv\ldapdbx.dll
dscomobx.dll	5.00.0984	188.77 KB (193,296 bytes)	3/15/2005 5:39:16 PM	Microsoft Corporation	c:\winnt\system32\inetsrv\dscomobx.dll
httpext.dll	5.00.0984	240.77 KB (246,544 bytes)	11/3/2004 3:03:21 PM	Microsoft Corporation	c:\winnt\system32\inetsrv\httpext.dll
rpcproxy.dll	5.00.2195.6701	16.27 KB (16,656 bytes)	11/3/2004 3:04:06 PM	Microsoft Corporation	c:\winnt\system32\rpcproxy\rpcproxy.dll
fpexedll.dll	4.0.2.7523	20.06 KB (20,541 bytes)	11/3/2004 3:03:13 PM	Microsoft Corporation	c:\program files\common files\microsoft shared\web server extensions\40\bin\fpexedll.dll
md5filt.dll	5.00.0984	32.77 KB (33,552 bytes)	11/3/2004 3:03:23 PM	Microsoft Corporation	c:\winnt\system32\inetsrv\md5filt.dll
gzip.dll	5.00.0984	30.27 KB (30,992 bytes)	11/3/2004 3:03:21 PM	Microsoft Corporation	c:\winnt\system32\inetsrv\gzip.dll
compfilt.dll	5.00.0984	22.77 KB (23,312 bytes)	11/3/2004 3:03:21 PM	Microsoft Corporation	c:\winnt\system32\inetsrv\compfilt.dll
odbcint.dll	3.520.7713.0	88.00 KB (90,112 bytes)	11/3/2004 5:27:33 PM	Microsoft Corporation	c:\winnt\system32\odbcint.dll
odbc32.dll	3.520.7713.0	196.00 KB (200,704 bytes)	11/3/2004 5:27:34 PM	Microsoft Corporation	c:\winnt\system32\odbc32.dll
ldapaclx.dll	5.00.0984	8.27 KB (8,464 bytes)	3/15/2005 5:39:16 PM	Microsoft Corporation	c:\winnt\system32\inetsrv\ldapaclx.dll
storedbx.dll	5.00.0984	251.27 KB (257,296 bytes)	11/3/2004 3:03:26 PM	Microsoft Corporation	c:\winnt\system32\inetsrv\storedbx.dll
ladminx.dll	5.00.0984	61.27 KB (62,736 bytes)	11/3/2004 3:04:05 PM	Microsoft Corporation	c:\winnt\system32\inetsrv\ladminx.dll
sspifilt.dll	5.00.0984	42.77 KB (43,792 bytes)	11/3/2004 3:03:23 PM	Microsoft Corporation	c:\winnt\system32\inetsrv\sspifilt.dll
iscomlog.dll	5.00.0984	24.27 KB (24,848 bytes)	11/3/2004 3:03:22 PM	Microsoft Corporation	c:\winnt\system32\inetsrv\iscomlog.dll
lonsint.dll	5.00.0984	11.77 KB (12,048 bytes)	11/3/2004 3:03:22 PM	Microsoft Corporation	c:\winnt\system32\inetsrv\lonsint.dll
inetsloc.dll	5.00.0984	20.27 KB (20,752 bytes)	11/3/2004 3:02:38 PM	Microsoft Corporation	c:\winnt\system32\inetsloc.dll
w3svc.dll	5.00.0984	338.27 KB (346,384 bytes)	11/3/2004 3:03:23 PM	Microsoft Corporation	c:\winnt\system32\inetsrv\w3svc.dll
staxmem.dll	5.00.0984	8.27 KB (8,464 bytes)	11/3/2004 3:02:57 PM	Microsoft Corporation	c:\winnt\system32\staxmem.dll
exstrace.dll	5.00.0984	13.77 KB (14,096 bytes)	11/3/2004 8:55:28 AM	Microsoft Corporation	c:\winnt\system32\exstrace.dll
iisfecnv.dll	5.00.0984	7.27 KB (7,440 bytes)	11/3/2004 8:55:27 AM	Microsoft Corporation	c:\winnt\system32\inetsrv\iisfecnv.dll
isatq.dll	5.00.0984	61.27 KB (62,736 bytes)	11/3/2004 3:03:22 PM	Microsoft Corporation	c:\winnt\system32\inetsrv\isatq.dll
infocomm.dll	5.00.0984	242.27 KB (248,080 bytes)	11/3/2004 3:03:22 PM	Microsoft Corporation	c:\winnt\system32\inetsrv\infocomm.dll
ldapsvcx.dll	5.00.0984	126.77 KB (129,808 bytes)	11/3/2004 3:03:33 PM	Microsoft Corporation	c:\winnt\system32\inetsrv\ldapsvcx.dll
security.dll	5.00.2154.1	5.77 KB (5,904 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\security.dll
svcxext.dll	5.00.0984	39.77 KB (40,720 bytes)	11/3/2004 3:03:23 PM	Microsoft Corporation	c:\winnt\system32\inetsrv\svcxext.dll
admexs.dll	5.00.0984	27.77 KB (28,432 bytes)	11/3/2004 3:03:20 PM	Microsoft Corporation	c:\winnt\system32\inetsrv\admexs.dll
wamreg.dll	5.00.0984	45.77 KB (46,864 bytes)	11/3/2004 3:03:24 PM	Microsoft Corporation	c:\winnt\system32\inetsrv\wamreg.dll
metadata.dll	5.00.0984	68.77 KB (70,416 bytes)	11/3/2004 3:03:23 PM	Microsoft Corporation	c:\winnt\system32\inetsrv\metadata.dll
iismap.dll	5.00.0984	56.27 KB (57,616 bytes)	11/3/2004 3:02:37 PM	Microsoft Corporation	c:\winnt\system32\iismap.dll
nsepm.dll	5.00.0984	43.27 KB (44,304 bytes)	11/3/2004 3:03:23 PM	Microsoft Corporation	c:\winnt\system32\inetsrv\nsepm.dll
admwprox.dll	5.00.0984	31.77 KB (32,528 bytes)	11/3/2004 8:55:27 AM	Microsoft Corporation	c:\winnt\system32\admwprox.dll
coadmin.dll	5.00.0984	39.77 KB (40,720 bytes)	11/3/2004 3:03:21 PM	Microsoft Corporation	c:\winnt\system32\inetsrv\coadmin.dll
iisadmin.dll	5.00.0984	15.77 KB (16,144 bytes)	11/3/2004 3:03:21 PM	Microsoft Corporation	c:\winnt\system32\inetsrv\iisadmin.dll
rpref.dll	5.00.0984	4.27 KB (4,368 bytes)	11/3/2004 3:03:23 PM	Microsoft Corporation	c:\winnt\system32\inetsrv\rpref.dll
iisrtl.dll	5.00.0984	121.27 KB (124,176 bytes)	11/3/2004 3:02:37 PM	Microsoft Corporation	c:\winnt\system32\iisrtl.dll
inetinfo.exe	5.00.0984	14.27 KB (14,608 bytes)	11/3/2004 3:03:22 PM	Microsoft Corporation	c:\winnt\system32\inetsrv\inetinfo.exe
wshnetbs.dll	5.00.2134.1	7.77 KB (7,952 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\wshnetbs.dll

ntmarta.dll	5.00.2195.6666	100.27 KB (102,672 bytes)	11/3/2004 3:02:49 PM	Microsoft Corporation	c:\winnt\system32\ntmarta.dll
provthrd.dll	1.50.1085.0000	68.07 KB (69,708 bytes)	11/3/2004 1:58:37 PM	Microsoft Corporation	c:\winnt\system32\wbem\provthrd.dll
ntevt.dll	1.50.1085.0072	192.06 KB (196,671 bytes)	11/3/2004 3:03:06 PM	Microsoft Corporation	c:\winnt\system32\wbem\ntevt.dll
perfos.dll	5.00.2155.1	21.27 KB (21,776 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\perfos.dll
psapi.dll	5.00.2134.1	28.27 KB (28,944 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\psapi.dll
framedyn.dll	1.50.1085.0076	164.07 KB (168,009 bytes)	11/3/2004 3:03:06 PM	Microsoft Corporation	c:\winnt\system32\wbem\framedyn.dll
cimwin32.dll	1.50.1085.0103	1.04 MB (1,089,637 bytes)	11/3/2004 3:03:05 PM	Microsoft Corporation	c:\winnt\system32\wbem\cimwin32.dll
wbemsvc.dll	1.50.1085.0007	40.07 KB (41,036 bytes)	11/3/2004 3:03:07 PM	Microsoft Corporation	c:\winnt\system32\wbem\wbemsvc.dll
wbemess.dll	1.50.1085.0100	364.09 KB (372,825 bytes)	11/3/2004 3:03:07 PM	Microsoft Corporation	c:\winnt\system32\wbem\wbemess.dll
fastprox.dll	1.50.1085.0100	152.10 KB (155,749 bytes)	11/3/2004 3:03:06 PM	Microsoft Corporation	c:\winnt\system32\wbem\fastprox.dll
wbemcore.dll	1.50.1085.0100	632.09 KB (647,257 bytes)	11/3/2004 3:03:06 PM	Microsoft Corporation	c:\winnt\system32\wbem\wbemcore.dll
wbemcomn.dll	1.50.1085.0100	692.09 KB (708,696 bytes)	11/3/2004 3:03:06 PM	Microsoft Corporation	c:\winnt\system32\wbem\wbemcomn.dll
winmgmt.exe	1.50.1085.0100	192.10 KB (196,706 bytes)	11/3/2004 3:03:07 PM	Microsoft Corporation	c:\winnt\system32\wbem\winmgmt.exe
simptcp.dll	5.00.2134.1	19.27 KB (19,728 bytes)	5/3/2005 5:39:17 PM	Microsoft Corporation	c:\winnt\system32\simptcp.dll
tcpsvcs.exe	5.00.2134.1	24.77 KB (25,360 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\tcpsvcs.exe
usp10.dll	1.0325.2195.6692	308.27 KB (315,664 bytes)	11/3/2004 3:02:59 PM	Microsoft Corporation	c:\winnt\system32\usp10.dll
thumbvw.dll	5.00.3502.6601	183.27 KB (187,664 bytes)	11/3/2004 3:02:58 PM	Microsoft Corporation	c:\winnt\system32\thumbvw.dll
mshtml.dll	5.00.3700.6699	229.77 KB (235,280 bytes)	11/3/2004 3:02:44 PM	Microsoft Corporation	c:\winnt\system32\mshtml.dll
imgutil.dll	5.00.3700.6682	30.77 KB (31,504 bytes)	11/3/2004 3:02:37 PM	Microsoft Corporation	c:\winnt\system32\imgutil.dll
webvw.dll	5.00.2920.0000	1.06 MB (1,115,408 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\webvw.dll
mshhtml.dll	5.00.3700.6699	2.24 MB (2,353,936 bytes)	11/3/2004 3:02:43 PM	Microsoft Corporation	c:\winnt\system32\mshhtml.dll
faxshell.dll	5.00.2134.1	8.27 KB (8,464 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\faxshell.dll
msacm32.dll	5.00.2134.1	65.27 KB (66,832 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\msacm32.dll
avifil32.dll	5.00.2195.6612	76.77 KB (78,608 bytes)	11/3/2004 3:02:27 PM	Microsoft Corporation	c:\winnt\system32\avifil32.dll
msvfw32.dll	5.00.2195.6612	113.77 KB (116,496 bytes)	11/3/2004 3:02:46 PM	Microsoft Corporation	c:\winnt\system32\msvfw32.dll
docprop2.dll	5.00.2178.1	297.77 KB (304,912 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\docprop2.dll
netplwiz.dll	5.00.2195.6601	169.77 KB (173,840 bytes)	11/3/2004 3:02:48 PM	Microsoft Corporation	c:\winnt\system32\netplwiz.dll
netmsg.dll	5.00.2137.1	152.50 KB (156,160 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\netmsg.dll
netui2.dll	5.00.2134.1	280.27 KB (286,992 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\netui2.dll
mprui.dll	5.00.2195.6601	54.77 KB (56,080 bytes)	11/3/2004 3:02:42 PM	Microsoft Corporation	c:\winnt\system32\mprui.dll
mydocs.dll	5.00.3502.6601	55.77 KB (57,104 bytes)	11/3/2004 3:02:47 PM	Microsoft Corporation	c:\winnt\system32\mydocs.dll
jscrip.dll	5.1.0.8513	476.06 KB (487,481 bytes)	11/3/2004 3:02:40 PM	Microsoft Corporation	c:\winnt\system32\jscrip.dll

imm32.dll	5.00.2195.6655	94.27 KB (96,528 bytes)	11/3/2004 3:02:37 PM	Microsoft Corporation	c:\winnt\system32\imm32.dll
msdbg.dll	6.00.8424	67.50 KB (69,120 bytes)	11/3/2004 8:56:48 AM	Microsoft Corporation	c:\winnt\system32\msdbg.dll
pdm.dll	6.00.8424	179.27 KB (183,574 bytes)	11/3/2004 8:56:49 AM	Microsoft Corporation	c:\winnt\system32\pdm.dll
mlang.dll	5.00.3700.6655	510.77 KB (523,024 bytes)	11/3/2004 3:02:41 PM	Microsoft Corporation	c:\winnt\system32\mlang.dll
urlmon.dll	5.00.3700.6705	442.77 KB (453,392 bytes)	11/3/2004 3:02:59 PM	Microsoft Corporation	c:\winnt\system32\urlmon.dll
wininet.dll	5.00.3700.6713	455.77 KB (466,704 bytes)	11/3/2004 3:03:01 PM	Microsoft Corporation	c:\winnt\system32\wininet.dll
browseui.dll	5.00.3700.6661	34.50 KB (35,328 bytes)	11/3/2004 3:02:28 PM	Microsoft Corporation	c:\winnt\system32\browseui.dll
ntshrui.dll	5.00.2134.1	46.77 KB (47,888 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\ntshrui.dll
linkinfo.dll	5.00.2134.1	15.77 KB (16,144 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\linkinfo.dll
powrprof.dll	5.00.3502.6601	13.27 KB (13,584 bytes)	11/3/2004 3:02:52 PM	Microsoft Corporation	c:\winnt\system32\powrprof.dll
batmeter.dll	5.00.3502.6601	20.27 KB (20,752 bytes)	11/3/2004 3:02:28 PM	Microsoft Corporation	c:\winnt\system32\batmeter.dll
stobject.dll	5.00.2195.6601	79.27 KB (81,168 bytes)	11/3/2004 3:02:57 PM	Microsoft Corporation	c:\winnt\system32\stobject.dll
msi.dll	2.0.2600.1183	1.92 MB (2,017,792 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\msi.dll
webcheck.dll	5.00.3502.6601	251.77 KB (257,808 bytes)	11/3/2004 3:03:00 PM	Microsoft Corporation	c:\winnt\system32\webcheck.dll
netui1.dll	5.00.2134.1	210.27 KB (215,312 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\netui1.dll
netui0.dll	5.00.2195.6601	70.27 KB (71,952 bytes)	11/3/2004 3:02:48 PM	Microsoft Corporation	c:\winnt\system32\netui0.dll
ntlanman.dll	5.00.2195.6601	35.27 KB (36,112 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\ntlanman.dll
hhsetup.dll	5.2.3644.0	37.00 KB (37,888 bytes)	11/3/2004 3:02:37 PM	Microsoft Corporation	c:\winnt\system32\hhsetup.dll
mmschext.dll	5.00.2153.1	24.27 KB (24,848 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\mmschext.dll
browseui.dll	5.00.3700.6661	789.27 KB (808,208 bytes)	11/3/2004 3:02:28 PM	Microsoft Corporation	c:\winnt\system32\browseui.dll
shdocvw.dll	5.00.3700.6668	1.06 MB (1,107,728 bytes)	11/3/2004 3:02:55 PM	Microsoft Corporation	c:\winnt\system32\shdocvw.dll
explorer.exe	5.00.3700.6690	237.77 KB (243,472 bytes)	11/3/2004 3:03:02 PM	Microsoft Corporation	c:\winnt\explorer.exe
msidle.dll	5.00.2920.0000	6.27 KB (6,416 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\msidle.dll
mstask.exe	4.71.2195.6704	116.77 KB (119,568 bytes)	11/3/2004 3:02:46 PM	Microsoft Corporation	c:\winnt\system32\mstask.exe
regsvcs.exe	5.00.2195.6701	66.77 KB (68,368 bytes)	11/3/2004 3:02:53 PM	Microsoft Corporation	c:\winnt\system32\regsvcs.exe
llsrpc.dll	5.00.2195.6601	47.77 KB (48,912 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\llsrpc.dll
llssrv.exe	5.00.2195.6697	81.77 KB (83,728 bytes)	6/19/2003 1:05:04 PM	Microsoft Corporation	c:\winnt\system32\llssrv.exe
ipbootp.dll	5.00.2168.1	33.77 KB (34,576 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\ipbootp.dll
cryptui.dll	5.131.2195.6628	433.27 KB (443,664 bytes)	11/3/2004 3:02:31 PM	Microsoft Corporation	c:\winnt\system32\cryptui.dll
rastls.dll	5.00.2195.6680	98.27 KB (100,624 bytes)	11/3/2004 3:02:53 PM	Microsoft Corporation	c:\winnt\system32\rastls.dll
raschap.dll	5.00.2195.6663	59.27 KB (60,688 bytes)	11/3/2004 3:02:52 PM	Microsoft Corporation	c:\winnt\system32\raschap.dll
rasppp.dll	5.00.2195.6626	194.27 KB (198,928 bytes)	11/3/2004 3:02:53 PM	Microsoft Corporation	c:\winnt\system32\rasppp.dll
rastapi.dll	5.00.2195.6604	52.77 KB (54,032 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\rastapi.dll
rasdlg.dll	5.00.2195.6625	516.77 KB (529,168 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\rasdlg.dll
netcfgx.dll	5.00.2195.6604	534.77 KB (547,600 bytes)	11/3/2004 3:02:47 PM	Microsoft Corporation	c:\winnt\system32\netcfgx.dll

rasmans.dll	5.00.2195.6696	149.77 KB (153,360 bytes)	
11/3/2004 3:02:52 PM	Microsoft Corporation		
c:\winnt\system32\rasmans.dll			
wmi.dll	5.00.2191.1	6.27 KB (6,416 bytes)	12/7/1999 7:00:00 AM
Microsoft Corporation		c:\winnt\system32\wmi.dll	
netshell.dll	5.00.2195.6604	466.27 KB (477,456 bytes)	11/3/2004 3:02:48 PM
Microsoft Corporation			
c:\winnt\system32\netshell.dll			
netman.dll	5.00.2195.6660	93.27 KB (95,504 bytes)	11/3/2004 3:02:48 PM
Microsoft Corporation			
c:\winnt\system32\netman.dll			
ntmsdba.dll	5.00.2195.6655	169.27 KB (173,328 bytes)	
11/3/2004 3:02:49 PM	Microsoft Corporation		
c:\winnt\system32\ntmsdba.dll			
sens.dll	5.00.2195.6627	37.27 KB (38,160 bytes)	11/3/2004 3:02:54 PM
Microsoft Corporation			
c:\winnt\system32\sens.dll			
iashlpr.dll	5.00.2184.1	33.27 KB (34,064 bytes)	12/7/1999 7:00:00 AM
Microsoft Corporation			
c:\winnt\system32\iaslpr.dll			
iasacct.dll	5.00.2195.6603	28.27 KB (28,944 bytes)	11/3/2004 3:02:37 PM
Microsoft Corporation			
c:\winnt\system32\iasacct.dll			
iasuser.dll	5.00.2195.6622	19.77 KB (20,240 bytes)	12/7/1999 7:00:00 AM
Microsoft Corporation			
c:\winnt\system32\iasuser.dll			
iasnap.dll	5.00.2195.6601	58.77 KB (60,176 bytes)	11/3/2004 3:02:37 PM
Microsoft Corporation			
c:\winnt\system32\iasnap.dll			
iaspipe.dll	5.00.2134.1	41.77 KB (42,768 bytes)	12/7/1999 7:00:00 AM
Microsoft Corporation			
c:\winnt\system32\iaspipe.dll			
expsrv.dll	6.0.9589	372.03 KB (380,957 bytes)	11/3/2004 3:02:35 PM
Microsoft Corporation		c:\winnt\system32\expsrv.dll	
vbajet32.dll	6.1.9431	30.03 KB (30,749 bytes)	11/3/2004 3:02:59 PM
Microsoft Corporation			
c:\winnt\system32\vbajet32.dll			
msjtes40.dll	4.00.7328.0	236.27 KB (241,936 bytes)	
11/3/2004 3:02:45 PM	Microsoft Corporation		
c:\winnt\system32\msjtes40.dll			
oledb32r.dll	2.70.7713.0 built by: Lab06_N(dagbuild)	64.00 KB (65,536 bytes)	11/3/2004 5:27:34 PM
Microsoft Corporation			
c:\program files\common files\system\ole db\oledb32r.dll			
comdlg32.dll	5.00.3700.6693	235.77 KB (241,424 bytes)	12/7/1999 7:00:00 AM
Microsoft Corporation			
c:\winnt\system32\comdlg32.dll			
msdart.dll	2.70.7713.0 built by: Lab06_N(dagbuild)	124.00 KB (126,976 bytes)	11/3/2004 5:27:34 PM
Microsoft Corporation			
c:\winnt\system32\msdart.dll			
oledb32.dll	2.70.7713.0 built by: Lab06_N(dagbuild)	404.00 KB (413,696 bytes)	11/3/2004 5:27:34 PM
Microsoft Corporation			
c:\program files\common files\system\ole db\oledb32.dll			
msjint40.dll	4.00.6508.0	148.27 KB (151,824 bytes)	
11/3/2004 3:02:45 PM	Microsoft Corporation		
c:\winnt\system32\msjint40.dll			
msjter40.dll	4.00.6508.0	52.27 KB (53,520 bytes)	
11/3/2004 3:02:45 PM	Microsoft Corporation		
c:\winnt\system32\msjter40.dll			
mswstr10.dll	4.00.6508.0	600.27 KB (614,672 bytes)	
11/3/2004 3:02:47 PM	Microsoft Corporation		
c:\winnt\system32\mswstr10.dll			
msjet40.dll	4.00.7328.0	1.44 MB (1,507,600 bytes)	11/3/2004 3:02:44 PM
Microsoft Corporation			
c:\winnt\system32\msjet40.dll			
msjtoledb40.dll	4.00.6807.0	340.27 KB (348,432 bytes)	
11/3/2004 3:02:45 PM	Microsoft Corporation		
c:\winnt\system32\msjtoledb40.dll			
iasrad.dll	5.00.2195.6601	94.77 KB (97,040 bytes)	11/3/2004 3:02:37 PM
Microsoft Corporation			
c:\winnt\system32\iasrad.dll			
iassam.dll	5.00.2195.6601	98.27 KB (100,624 bytes)	11/3/2004 3:02:37 PM
Microsoft Corporation			
c:\winnt\system32\iassam.dll			
iasads.dll	5.00.2195.6601	73.77 KB (75,536 bytes)	11/3/2004 3:02:37 PM
Microsoft Corporation			
c:\winnt\system32\iasads.dll			
ntmssvc.dll	5.00.2195.6655	391.77 KB (401,168 bytes)	
11/3/2004 3:02:49 PM	Microsoft Corporation		
c:\winnt\system32\ntmssvc.dll			
iaspolcy.dll	5.00.2134.1	25.27 KB (25,872 bytes)	12/7/1999 7:00:00 AM
Microsoft Corporation			
c:\winnt\system32\iaspolcy.dll			
iasvcs.dll	5.00.2195.6601	58.77 KB (60,176 bytes)	11/3/2004 3:02:37 PM
Microsoft Corporation			
c:\winnt\system32\iasvcs.dll			
iassdo.dll	5.00.2195.6601	263.27 KB (269,584 bytes)	11/3/2004 3:02:37 PM
Microsoft Corporation			
c:\winnt\system32\iassdo.dll			
ias.dll	5.00.2134.1	7.27 KB (7,440 bytes)	12/7/1999 7:00:00 AM
Microsoft Corporation			
c:\winnt\system32\ias.dll			
es.dll	2000.2.3504.0	227.77 KB (233,232 bytes)	11/3/2004 3:02:34 PM
Microsoft Corporation			
c:\winnt\system32\es.dll			
db2sec.exe	8.1.6.574	29.11 KB (29,808 bytes)	6/17/2004 11:32:48 PM
International Business Machines Corporation			
c:\sqllib\bin\db2sec.exe			
db2cli.dll	8.1.6.574	2.77 MB (2,908,222 bytes)	6/17/2004 8:30:24 PM
International Business Machines Corporation			
c:\sqllib\bin\db2cli.dll			
db2abind.dll	8.1.6.574	244.06 KB (249,920 bytes)	6/17/2004 8:30:18 PM
International Business Machines Corporation			
c:\sqllib\bin\db2abind.dll			

db2util.dll 8.1.6.574 1.18 MB (1,237,055 bytes) 6/17/2004 8:31:08 PM International Business Machines Corporation c:\sqlib\bin\db2util.dll	msdtcprx.dll 2000.2.3504.0 690.77 KB (707,344 bytes) 11/3/2004 3:02:42 PM Microsoft Corporation c:\winnt\system32\msdtcprx.dll
db2install.dll 8.1.6.574 28.06 KB (28,738 bytes) 6/17/2004 8:28:20 PM International Business Machines Corporation c:\sqlib\bin\db2install.dll	txfaux.dll 2000.2.3504.0 388.27 KB (397,584 bytes) 11/3/2004 3:02:58 PM Microsoft Corporation c:\winnt\system32\txfaux.dll
db2trcapi.dll 8.1.6.574 36.07 KB (36,938 bytes) 6/17/2004 8:28:20 PM International Business Machines Corporation c:\sqlib\bin\db2trcapi.dll	msdtctm.dll 2000.2.3504.0 1.08 MB (1,131,280 bytes) 11/3/2004 3:02:42 PM Microsoft Corporation c:\winnt\system32\msdtctm.dll
db2locale.dll 8.1.6.574 48.06 KB (49,217 bytes) 6/17/2004 8:28:20 PM International Business Machines Corporation c:\sqlib\bin\db2locale.dll	msdtc.exe 1999.9.3421.3 6.77 KB (6,928 bytes) 11/3/2004 8:55:08 AM Microsoft Corporation c:\winnt\system32\msdtc.exe
db2osse.dll 8.1.6.574 312.07 KB (319,561 bytes) 6/17/2004 8:31:10 PM International Business Machines Corporation c:\sqlib\bin\db2osse.dll	clbcatq.dll 2000.2.3504.0 498.27 KB (510,224 bytes) 11/3/2004 3:02:30 PM Microsoft Corporation c:\winnt\system32\clbcatq.dll
db2g11n.dll 8.1.6.574 364.06 KB (372,799 bytes) 6/17/2004 8:28:20 PM International Business Machines Corporation c:\sqlib\bin\db2g11n.dll	rasadhlp.dll 5.00.2168.1 7.27 KB (7,440 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rasadhlp.dll
db2syp.dll 8.1.6.574 88.06 KB (90,176 bytes) 6/17/2004 8:31:06 PM International Business Machines Corporation c:\sqlib\bin\db2syp.dll	winrnr.dll 5.00.2160.1 18.77 KB (19,216 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\winrnr.dll
db2wint.dll 8.1.6.574 48.06 KB (49,215 bytes) 6/17/2004 8:31:08 PM International Business Machines Corporation c:\sqlib\bin\db2wint.dll	rpss.dll 5.00.2195.6702 233.77 KB (239,376 bytes) 11/3/2004 3:02:53 PM Microsoft Corporation c:\winnt\system32\rpss.dll
db2sys.dll 8.1.6.574 2.38 MB (2,490,430 bytes) 6/17/2004 8:31:04 PM International Business Machines Corporation c:\sqlib\bin\db2sys.dll	svchost.exe 5.00.2134.1 7.77 KB (7,952 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\svchost.exe
db2app.dll 8.1.6.574 2.51 MB (2,629,694 bytes) 6/17/2004 8:30:20 PM International Business Machines Corporation c:\sqlib\bin\db2app.dll	iissuba.dll 5.00.0984 9.77 KB (10,000 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\iissuba.dll
db2jds.exe 8.1.6.574 193.12 KB (197,752 bytes) 6/17/2004 11:30:56 PM International Business Machines Corporation c:\sqlib\bin\db2jds.exe	dssenh.dll 5.00.2195.6612 143.77 KB (147,216 bytes) 11/3/2004 3:03:16 PM Microsoft Corporation c:\winnt\system32\dssenh.dll
mtxoci.dll 2000.2.3504.0 103.27 KB (105,744 bytes) 11/3/2004 3:02:47 PM Microsoft Corporation c:\winnt\system32\mtxoci.dll	oakley.dll 5.00.2195.6662 435.77 KB (446,224 bytes) 11/3/2004 3:02:49 PM Microsoft Corporation c:\winnt\system32\oakley.dll
resutils.dll 5.00.2195.6702 39.77 KB (40,720 bytes) 11/3/2004 3:02:53 PM Microsoft Corporation c:\winnt\system32\resutils.dll	mfc42u.dll 6.00.9586.0 988.05 KB (1,011,764 bytes) 11/3/2004 3:02:41 PM Microsoft Corporation c:\winnt\system32\mfc42u.dll
clusapi.dll 5.00.2195.6683 54.27 KB (55,568 bytes) 11/3/2004 3:02:30 PM Microsoft Corporation c:\winnt\system32\clusapi.dll	polagent.dll 5.00.2195.6655 109.27 KB (111,888 bytes) 11/3/2004 3:02:52 PM Microsoft Corporation c:\winnt\system32\polagent.dll
msvcpx50.dll 5.00.7051 552.50 KB (565,760 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\msvcpx50.dll	scecli.dll 5.00.2195.6704 111.77 KB (114,448 bytes) 11/3/2004 3:02:54 PM Microsoft Corporation c:\winnt\system32\scecli.dll
xolehlp.dll 1999.9.3421.3 17.27 KB (17,680 bytes) 11/3/2004 8:55:08 AM Microsoft Corporation c:\winnt\system32\xolehlp.dll	esent.dll 6.1.3940.31 1.08 MB (1,135,376 bytes) 11/3/2004 3:02:34 PM Microsoft Corporation c:\winnt\system32\esent.dll
msdtclog.dll 2000.2.3504.0 86.77 KB (88,848 bytes) 11/3/2004 3:02:42 PM Microsoft Corporation c:\winnt\system32\msdtclog.dll	mswsock.dll 5.00.2195.6603 62.77 KB (64,272 bytes) 11/3/2004 3:02:47 PM Microsoft Corporation c:\winnt\system32\mswsock.dll
mtxclu.dll 2000.2.3504.0 51.27 KB (52,496 bytes) 11/3/2004 3:02:47 PM Microsoft Corporation c:\winnt\system32\mtxclu.dll	

ntdsatq.dll 5.00.2195.6620 31.27 KB (32,016 bytes) 11/3/2004 3:02:49 PM Microsoft Corporation c:\winnt\system32\ntdsatq.dll	seclogon.dll 5.00.2195.6707 16.77 KB (17,168 bytes) 11/3/2004 3:02:54 PM Microsoft Corporation c:\winnt\system32\seclogon.dll
ntdsa.dll 5.00.2195.6697 1016.27 KB (1,040,656 bytes) 11/3/2004 3:02:48 PM Microsoft Corporation c:\winnt\system32\ntdsa.dll	rnrr20.dll 5.00.2195.6603 35.77 KB (36,624 bytes) 11/3/2004 3:02:53 PM Microsoft Corporation c:\winnt\system32\rnrr20.dll
kdcsvc.dll 5.00.2195.6627 144.77 KB (148,240 bytes) 11/3/2004 3:02:40 PM Microsoft Corporation c:\winnt\system32\kdcsvc.dll	psbase.dll 5.00.2195.6661 112.77 KB (115,472 bytes) 11/3/2004 3:02:52 PM Microsoft Corporation c:\winnt\system32\psbase.dll
sfmapi.dll 5.00.2134.1 38.77 KB (39,696 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\sfmapi.dll	cryptsvc.dll 5.00.2195.6661 74.27 KB (76,048 bytes) 11/3/2004 3:02:31 PM Microsoft Corporation c:\winnt\system32\cryptsvc.dll
rassfm.dll 5.00.2195.6604 21.27 KB (21,776 bytes) 11/3/2004 3:02:53 PM Microsoft Corporation c:\winnt\system32\rassfm.dll	cryptdll.dll 5.00.2195.6607 43.27 KB (44,304 bytes) 11/3/2004 3:02:31 PM Microsoft Corporation c:\winnt\system32\cryptdll.dll
rsabase.dll 5.00.2195.6619 129.27 KB (132,368 bytes) 6/19/2003 1:05:04 PM Microsoft Corporation c:\winnt\system32\rsabase.dll	wkssvc.dll 5.00.2195.6692 95.77 KB (98,064 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\wkssvc.dll
schannel.dll 5.00.2195.6705 144.27 KB (147,728 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\schannel.dll	srvsvc.dll 5.00.2195.6697 81.77 KB (83,728 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\srvsvc.dll
netlogon.dll 5.00.2195.6695 363.27 KB (371,984 bytes) 11/3/2004 3:02:48 PM Microsoft Corporation c:\winnt\system32\netlogon.dll	cfgmgr32.dll 5.00.2134.1 16.77 KB (17,168 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\cfgmgr32.dll
kerberos.dll 5.00.2195.6666 207.77 KB (212,752 bytes) 11/3/2004 3:02:40 PM Microsoft Corporation c:\winnt\system32\kerberos.dll	dmserver.dll 2195.6605.297.3 11.77 KB (12,048 bytes) 11/3/2004 3:02:33 PM VERITAS Software Corp. c:\winnt\system32\dmserver.dll
msprivs.dll 5.00.2195.6695 46.00 KB (47,104 bytes) 11/3/2004 3:02:45 PM Microsoft Corporation c:\winnt\system32\msprivs.dll	wshtcpip.dll 5.00.2195.6601 17.27 KB (17,680 bytes) 11/3/2004 3:03:01 PM Microsoft Corporation c:\winnt\system32\wshtcpip.dll
samsrv.dll 5.00.2195.6697 380.77 KB (389,904 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\samsrv.dll	msafd.dll 5.00.2195.6602 106.27 KB (108,816 bytes) 11/3/2004 3:02:42 PM Microsoft Corporation c:\winnt\system32\msafd.dll
lsasrv.dll 5.00.2195.6695 506.77 KB (518,928 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\lsasrv.dll	lmhsvc.dll 5.00.2195.6601 9.77 KB (10,000 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\lmhsvc.dll
lsass.exe 5.00.2195.6695 32.77 KB (33,552 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\lsass.exe	dnssrslvr.dll 5.00.2195.6663 90.27 KB (92,432 bytes) 11/3/2004 3:02:33 PM Microsoft Corporation c:\winnt\system32\dnssrslvr.dll
ntlsapi.dll 5.00.2195.6601 6.77 KB (6,928 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\ntlsapi.dll	tapi32.dll 5.00.2195.6664 123.77 KB (126,736 bytes) 11/3/2004 3:02:58 PM Microsoft Corporation c:\winnt\system32\tapi32.dll
wmicore.dll 5.00.2195.6611 72.77 KB (74,512 bytes) 11/3/2004 3:03:01 PM Microsoft Corporation c:\winnt\system32\wmicore.dll	rasman.dll 5.00.2195.6604 54.77 KB (56,080 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rasman.dll
alrsvc.dll 5.00.2134.1 17.77 KB (18,192 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\alrsvc.dll	rasapi32.dll 5.00.2195.6625 192.77 KB (197,392 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rasapi32.dll
trkwks.dll 5.00.2195.6623 88.27 KB (90,384 bytes) 11/3/2004 3:02:58 PM Microsoft Corporation c:\winnt\system32\trkwks.dll	rtutils.dll 5.00.2168.1 43.77 KB (44,816 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\rtutils.dll

adslrpc.dll	5.00.2195.6701	130.77 KB (133,904 bytes)	11/3/2004 3:02:26 PM	Microsoft Corporation	c:\winnt\system32\adslrpc.dll
activeds.dll	5.00.2195.6601	177.77 KB (182,032 bytes)	11/3/2004 3:02:22 PM	Microsoft Corporation	c:\winnt\system32\activeds.dll
mprapi.dll	5.00.2181.1	79.27 KB (81,168 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\mprapi.dll
iphlpapi.dll	5.00.2195.6602	68.27 KB (69,904 bytes)	11/3/2004 3:02:38 PM	Microsoft Corporation	c:\winnt\system32\iphlpapi.dll
icmp.dll	5.00.2134.1	7.27 KB (7,440 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\icmp.dll
dhcpcsvc.dll	5.00.2195.6685	90.77 KB (92,944 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\dhcpcsvc.dll
eventlog.dll	5.00.2195.6716	46.77 KB (47,888 bytes)	11/3/2004 3:02:35 PM	Microsoft Corporation	c:\winnt\system32\eventlog.dll
ntdsapi.dll	5.00.2195.6666	56.27 KB (57,616 bytes)	11/3/2004 3:02:48 PM	Microsoft Corporation	c:\winnt\system32\ntdsapi.dll
scesrv.dll	5.00.2195.6704	248.77 KB (254,736 bytes)	11/3/2004 3:02:54 PM	Microsoft Corporation	c:\winnt\system32\scesrv.dll
umpnpmgr.dll	5.00.2182.1	86.27 KB (88,336 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\umpnpmgr.dll
services.exe	5.00.2195.6700	87.27 KB (89,360 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\services.exe
msv1_0.dll	5.00.2195.6680	114.77 KB (117,520 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\msv1_0.dll
wzcsapi.dll	5.00.2195.6604	29.27 KB (29,968 bytes)	11/3/2004 3:03:18 PM	Microsoft Corporation	c:\winnt\system32\wzcsapi.dll
oleaut32.dll	2.40.4522	612.27 KB (626,960 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\oleaut32.dll
wzcdlg.dll	5.00.2195.6604	51.27 KB (52,496 bytes)	11/3/2004 3:03:18 PM	Microsoft Corporation	c:\winnt\system32\wzcdlg.dll
cscui.dll	5.00.2195.6705	237.27 KB (242,960 bytes)	11/3/2004 3:02:31 PM	Microsoft Corporation	c:\winnt\system32\cscui.dll
mpr.dll	5.00.2195.6611	53.77 KB (55,056 bytes)	11/3/2004 3:02:42 PM	Microsoft Corporation	c:\winnt\system32\mpr.dll
winspool.drv	5.00.2195.6659	111.27 KB (113,936 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\winspool.drv
winscard.dll	5.00.2195.6609	77.27 KB (79,120 bytes)	11/3/2004 3:03:01 PM	Microsoft Corporation	c:\winnt\system32\winscard.dll
atl.dll	3.00.9435	73.06 KB (74,810 bytes)	11/3/2004 3:02:27 PM	Microsoft Corporation	c:\winnt\system32\atl.dll
certcli.dll	5.00.2195.6619	132.27 KB (135,440 bytes)	11/3/2004 3:02:29 PM	Microsoft Corporation	c:\winnt\system32\certcli.dll
wlnotify.dll	5.00.2195.6706	56.27 KB (57,616 bytes)	11/3/2004 3:03:01 PM	Microsoft Corporation	c:\winnt\system32\wlnotify.dll
cscdll.dll	5.00.2195.6713	98.77 KB (101,136 bytes)	11/3/2004 3:02:31 PM	Microsoft Corporation	c:\winnt\system32\cscdll.dll
lz32.dll	5.00.2195.6611	9.77 KB (10,000 bytes)	11/3/2004 3:02:40 PM	Microsoft Corporation	c:\winnt\system32\lz32.dll
version.dll	5.00.2195.6623	15.77 KB (16,144 bytes)	11/3/2004 3:03:00 PM	Microsoft Corporation	c:\winnt\system32\version.dll
rsaenh.dll	5.00.2195.6611	131.77 KB (134,928 bytes)	11/3/2004 3:03:17 PM	Microsoft Corporation	c:\winnt\system32\rsaenh.dll
mscat32.dll	5.131.2134.1	7.77 KB (7,952 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\mscat32.dll
ole32.dll	5.00.2195.6692	972.77 KB (996,112 bytes)	11/3/2004 3:02:51 PM	Microsoft Corporation	c:\winnt\system32\ole32.dll
imagehlp.dll	5.00.2195.6613	125.77 KB (128,784 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\imagehlp.dll
msasn1.dll	5.00.2195.6666	51.77 KB (53,008 bytes)	12/7/1999 7:00:00 AM	Microsoft Corporation	c:\winnt\system32\msasn1.dll
crypt32.dll	5.131.2195.6661	468.27 KB (479,504 bytes)	11/3/2004 3:02:31 PM	Microsoft Corporation	c:\winnt\system32\crypt32.dll
wintrust.dll	5.131.2195.6624	162.27 KB (166,160 bytes)	11/3/2004 3:03:01 PM	Microsoft Corporation	c:\winnt\system32\wintrust.dll
shlwapi.dll	5.00.3502.6601	282.77 KB (289,552 bytes)	11/3/2004 3:02:56 PM	Microsoft Corporation	c:\winnt\system32\shlwapi.dll
shell32.dll	5.00.3700.6705	2.27 MB (2,383,632 bytes)	11/3/2004 3:02:55 PM	Microsoft Corporation	c:\winnt\system32\shell32.dll

msgina.dll 5.00.2195.6669 326.27 KB (334,096 bytes) 11/3/2004 3:02:43 PM Microsoft Corporation c:\winnt\system32\msgina.dll	userenv.dll 5.00.2195.6711 380.77 KB (389,904 bytes) 11/3/2004 3:02:59 PM Microsoft Corporation c:\winnt\system32\userenv.dll
comctl32.dll 5.81 537.77 KB (550,672 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\comctl32.dll	rpcrt4.dll 5.00.2195.6701 443.77 KB (454,416 bytes) 11/3/2004 3:02:53 PM Microsoft Corporation c:\winnt\system32\rpcrt4.dll
setupapi.dll 5.00.2195.6622 556.77 KB (570,128 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\setupapi.dll	advapi32.dll 5.00.2195.6710 378.27 KB (387,344 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\advapi32.dll
winmm.dll 5.00.2161.1 184.77 KB (189,200 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\winmm.dll	msvcrt.dll 6.10.9844.0 280.05 KB (286,773 bytes) 6/19/2003 1:05:04 PM Microsoft Corporation c:\winnt\system32\msvcrt.dll
winsta.dll 5.00.2195.6701 38.27 KB (39,184 bytes) 11/3/2004 3:03:01 PM Microsoft Corporation c:\winnt\system32\winsta.dll	winlogon.exe 5.00.2195.6714 176.77 KB (181,008 bytes) 11/3/2004 3:03:01 PM Microsoft Corporation c:\winnt\system32\winlogon.exe
wsock32.dll 5.00.2195.6603 21.27 KB (21,776 bytes) 11/3/2004 3:03:02 PM Microsoft Corporation c:\winnt\system32\wsock32.dll	gdi32.dll 5.00.2195.6660 228.27 KB (233,744 bytes) 11/3/2004 3:02:36 PM Microsoft Corporation c:\winnt\system32\gdi32.dll
dnsapi.dll 5.00.2195.6680 131.77 KB (134,928 bytes) 11/3/2004 3:02:33 PM Microsoft Corporation c:\winnt\system32\dnsapi.dll	kernel32.dll 5.00.2195.6688 725.77 KB (743,184 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\kernel32.dll
wldap32.dll 5.00.2195.6666 158.27 KB (162,064 bytes) 11/3/2004 3:03:01 PM Microsoft Corporation c:\winnt\system32\wldap32.dll	user32.dll 5.00.2195.6688 393.77 KB (403,216 bytes) 11/3/2004 3:02:59 PM Microsoft Corporation c:\winnt\system32\user32.dll
ws2help.dll 5.00.2134.1 17.77 KB (18,192 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\ws2help.dll	winsrv.dll 5.00.2195.6699 246.77 KB (252,688 bytes) 11/30/1999 6:39:54 PM Microsoft Corporation c:\winnt\system32\winsrv.dll
ws2_32.dll 5.00.2195.6601 68.27 KB (69,904 bytes) 11/3/2004 3:03:01 PM Microsoft Corporation c:\winnt\system32\ws2_32.dll	basesrv.dll 5.00.2195.6706 41.27 KB (42,256 bytes) 11/3/2004 3:02:28 PM Microsoft Corporation c:\winnt\system32\basesrv.dll
samlib.dll 5.00.2195.6666 48.77 KB (49,936 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\samlib.dll	csrssv.dll 5.00.2195.6601 34.27 KB (35,088 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\csrssv.dll
netrap.dll 5.00.2134.1 11.27 KB (11,536 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\netrap.dll	csrss.exe 5.00.2195.6601 5.27 KB (5,392 bytes) 11/3/2004 3:02:31 PM Microsoft Corporation c:\winnt\system32\csrss.exe
netapi32.dll 5.00.2195.6601 304.27 KB (311,568 bytes) 11/3/2004 3:02:47 PM Microsoft Corporation c:\winnt\system32\netapi32.dll	sfcfiles.dll 5.00.2195.6717 948.27 KB (971,024 bytes) 11/3/2004 3:02:55 PM Microsoft Corporation c:\winnt\system32\sfcfiles.dll
profmap.dll 5.00.2195.6610 29.27 KB (29,968 bytes) 11/3/2004 3:02:52 PM Microsoft Corporation c:\winnt\system32\profmap.dll	ntdll.dll 5.00.2195.6685 480.27 KB (491,792 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\ntdll.dll
secur32.dll 5.00.2195.6695 47.77 KB (48,912 bytes) 11/3/2004 3:02:54 PM Microsoft Corporation c:\winnt\system32\secur32.dll	smss.exe 5.00.2195.6601 44.77 KB (45,840 bytes) 12/7/1999 7:00:00 AM Microsoft Corporation c:\winnt\system32\smss.exe
sfc.dll 5.00.2195.6673 92.80 KB (95,024 bytes) 11/3/2004 3:02:54 PM Microsoft Corporation c:\winnt\system32\sfc.dll	[Services]
nddeapi.dll 5.00.2195.6661 15.77 KB (16,144 bytes) 11/3/2004 3:02:47 PM Microsoft Corporation c:\winnt\system32\nddeapi.dll	Display Name Name State Start ModeService Type Path Error Control Start NameTag ID

Alerter	Alerter	Running	Auto	Share Process		
c:\winnt\system32\services.exe		Normal	LocalSystem		0	
Application Management Process	AppMgmt	Stopped	Manual	Share LocalSystem		0
c:\winnt\system32\services.exe		Normal	LocalSystem			
Background Intelligent Transfer Service	BITS	Stopped	Manual	Share Process		
Share Process	c:\winnt\system32\svchost.exe -k bitsgroup	Normal	LocalSystem		0	
Computer Browser	Browser	Stopped	Manual	Share Process		
c:\winnt\system32\services.exe		Normal	LocalSystem		0	
Indexing Service	cisvc	Stopped	Disabled	Share Process		
c:\winnt\system32\cisvc.exe		Normal	LocalSystem		0	
ClipBook	ClipSrv	Stopped	Manual	Own Process		
c:\winnt\system32\clipsrv.exe		Normal	LocalSystem		0	
DB2 JDBC Applet Server	DB2JDS	Running	Auto	Own Process		
Process	"c:\sqllib\bin\db2jds.exe"	Normal	LocalSystem		0	
DB2 Security Server	DB2NTSECSERVER	Running	Auto	Own Process		
Process	"c:\sqllib\bin\db2sec.exe"	Normal	LocalSystem		0	
Distributed File System	Dfs	Running	Auto	Own Process		
Process	c:\winnt\system32\dfsvc.exe	Normal	LocalSystem		0	
DHCP Client	Dhcp	Running	Auto	Share Process		
c:\winnt\system32\services.exe		Normal	LocalSystem		0	
Logical Disk Manager	Administrative Service			dmadmin	Stopped	
Manual	Share Process			c:\winnt\system32\dmadmin.exe /com		
Normal	LocalSystem				0	
Logical Disk Manager	dmsvr	Running	Auto	Share Process		
c:\winnt\system32\services.exe		Normal	LocalSystem		0	
DNS Client	Dnscache	Running	Auto	Share Process		
c:\winnt\system32\services.exe		Normal	LocalSystem		0	
Event Log	Eventlog	Running	Auto	Share Process		
c:\winnt\system32\services.exe		Normal	LocalSystem		0	
COM+ Event System	EventSystem		Running	Manual	Share	
Process	c:\winnt\system32\svchost.exe -k netsvcs	Normal	LocalSystem		0	
Fax Service	Fax	Stopped	Manual	Own Process		
c:\winnt\system32\faxsvc.exe		Normal	LocalSystem		0	
Internet Authentication Service	IAS	Running	Auto	Share Process		
Process	c:\winnt\system32\svchost.exe -k netsvcs	Normal	LocalSystem		0	
IIS Admin Service	IISADMIN		Running	Auto	Share	
Process	c:\winnt\system32\inetrv\inetinfo.exe	Normal	LocalSystem		0	
Intersite Messaging	IsmServ	Stopped	Disabled	Own Process		
c:\winnt\system32\ismserv.exe		Normal	LocalSystem		0	
Kerberos Key Distribution Center			kdc	Stopped	Disabled	
Share Process	c:\winnt\system32\lsass.exe	Normal	LocalSystem		0	
Server	lanmanserver	Running	Auto	Share Process		
c:\winnt\system32\services.exe		Normal	LocalSystem		0	
Workstation	lanmanworkstation	Running	Auto	Share Process		
c:\winnt\system32\services.exe		Normal	LocalSystem		0	
Site Server ILS Service	LDAPSVCX	Running	Auto	Share Process		
LocalSystem	c:\winnt\system32\inetrv\inetinfo.exe	Normal	LocalSystem		0	
License Logging Service	LicenseService	Running	Auto	Own Process		
LocalSystem	c:\winnt\system32\llssrv.exe	Normal	LocalSystem		0	
TCP/IP NetBIOS Helper Service	LmHosts	Running	Auto	Share Process		
c:\winnt\system32\services.exe		Normal	LocalSystem		0	
Messenger	Messenger	Stopped	Manual	Share Process		
c:\winnt\system32\services.exe		Normal	LocalSystem		0	
NetMeeting Remote Desktop Sharing	mnmsrvc	Stopped	Manual	Own Process		
LocalSystem	c:\winnt\system32\mnmsrvc.exe	Normal	LocalSystem		0	
Distributed Transaction Coordinator	MSDTC	Running	Auto	Own Process		
LocalSystem	c:\winnt\system32\msdtc.exe	Normal	LocalSystem		0	
Windows Installer	MSIServer	Stopped	Manual	Share Process		
c:\winnt\system32\msiexec.exe /v		Normal	LocalSystem		0	
Network DDE	NetDDE	Stopped	Manual	Share Process		
c:\winnt\system32\netdde.exe		Normal	LocalSystem		0	
Network DDE DSDM	NetDDEdsdm	Stopped	Manual	Share Process		
Process	c:\winnt\system32\netdde.exe	Normal	LocalSystem		0	
Net Logon	Netlogon	Stopped	Manual	Share Process		
c:\winnt\system32\lsass.exe		Normal	LocalSystem		0	
Network Connections	Netman	Running	Manual	Share Process		
c:\winnt\system32\svchost.exe -k netsvcs		Normal	LocalSystem		0	
File Replication	NtFrs	Stopped	Manual	Own Process		
c:\winnt\system32\ntfrs.exe		Ignore	LocalSystem		0	
NT LM Security Support Provider			NtLmSsp	Stopped	Manual	
Share Process	c:\winnt\system32\lsass.exe	Normal	LocalSystem		0	
Removable Storage	NtmsSvc	Running	Auto	Share Process		
c:\winnt\system32\svchost.exe -k netsvcs		Normal	LocalSystem		0	
Plug and Play	PlugPlay	Running	Auto	Share Process		
c:\winnt\system32\services.exe		Normal	LocalSystem		0	
IPSEC Policy Agent	PolicyAgent	Running	Auto	Share Process		
Process	c:\winnt\system32\lsass.exe	Normal	LocalSystem		0	
Protected Storage	ProtectedStorage	Running	Auto	Share Process		
Process	c:\winnt\system32\services.exe	Normal	LocalSystem		0	
Remote Access Auto Connection Manager	RasAuto	Stopped	Manual	Share Process		
LocalSystem	c:\winnt\system32\svchost.exe -k netsvcs	Normal	LocalSystem		0	
Remote Access Connection Manager	RasMan	Running	Manual	Share Process		
LocalSystem	c:\winnt\system32\svchost.exe -k netsvcs	Normal	LocalSystem		0	

Routing and Remote Access Share Process LocalSystem	RemoteAccess	Stopped	Disabled	Normal	0	Uninterruptible Power Supply Process	UPS	Stopped	Manual	Own	LocalSystem	0
Remote Registry Service Own Process LocalSystem	RemoteRegistry	Running	Auto	Normal	0	Utility Manager	UtilMan	Stopped	Manual	Own Process	LocalSystem	0
Remote Procedure Call (RPC) Locator Manual Own Process LocalSystem	RpcLocator	Stopped	Normal	0	0	Windows Time	W32Time	Stopped	Manual	Share Process	LocalSystem	0
Remote Procedure Call (RPC) Process LocalSystem	RpcSs	Running	Auto	Share	0	World Wide Web Publishing Service Share Process LocalSystem	W3SVC	Running	Auto	Normal	0	0
QoS Admission Control (RSVP) Process	RSVP	Running	Auto	Own	0	Windows Management Instrumentation LocalSystem	WinMgmt	Running	Auto	Ignore	0	0
Security Accounts Manager Process	SamSs	Running	Auto	Share	0	Windows Management Instrumentation Driver Extensions Running	Wmi	Manual	Share Process	LocalSystem	0	0
Smart Card Helper c:\winnt\system32\scardsvr.exe	SCardDrv	Stopped	Manual	Share Process	0	Automatic Updates c:\winnt\system32\svchost.exe -k wugroup	wuauerv	Stopped	Manual	Share Process	LocalSystem	0
Smart Card c:\winnt\system32\scardsvr.exe	SCardSvr	Stopped	Manual	Share Process	0	Wireless Configuration LocalSystem	WZCSVC	Stopped	Manual	Share	Normal	0
Task Scheduler c:\winnt\system32\mstask.exe	Schedule	Running	Auto	Share Process	0	[Program Groups]						
RunAs Service c:\winnt\system32\services.exe	seclogon	Running	Auto	Share Process	0	Group Name	Name	User Name				
System Event Notification LocalSystem	SENS	Running	Auto	Share	0	Accessories	Default User:Accessories	Default User				
Internet Connection Sharing Share Process LocalSystem	SharedAccess	Stopped	Manual	Normal	0	Accessories\Accessibility	Default User:Accessories\Accessibility	Default User				
Simple TCP/IP Services Process	SimpTcp	Running	Auto	Share	0	Accessories\Entertainment	Default User:Accessories\Entertainment	Default User				
Print Spooler c:\winnt\system32\spoolsv.exe	Spooler	Stopped	Manual	Own Process	0	Accessories\System Tools	Default User:Accessories\System Tools	Default User				
Performance Logs and Alerts Own Process LocalSystem	SysmonLog	Stopped	Manual	Normal	0	Startup	Default User:Startup	Default User				
Telephony TapiSrv c:\winnt\system32\svchost.exe -k tapisrv	Running	Manual	Share Process	Normal	LocalSystem	Accessories	All Users:Accessories	All Users				
Terminal Services Process	TermService	Stopped	Manual	Own	LocalSystem	Accessories\Communications	All Users:Accessories\Communications	All Users				
Telnet c:\winnt\system32\tlntsvr.exe	TlntSvr	Stopped	Manual	Own Process	LocalSystem	Accessories\Entertainment	All Users:Accessories\Entertainment	All Users				
Distributed Link Tracking Server Process	TrkSvr	Stopped	Manual	Share	LocalSystem	Accessories\Microsoft Script Debugger	All Users:Accessories\Microsoft Script Debugger	All Users				
Distributed Link Tracking Client Process	TrkWks	Running	Auto	Share	LocalSystem	Accessories\System Tools	All Users:Accessories\System Tools	All Users				
						ActiveState	ActivePerl 5.8	All Users:ActiveState	ActivePerl 5.8			
						Administrative Tools	All Users:Administrative Tools	All Users				

IBM DB2 All Users:IBM DB2 All Users

IBM DB2\Command Line Tools All Users:IBM DB2\Command Line Tools All Users

IBM DB2\Development Tools All Users:IBM DB2\Development Tools All Users

IBM DB2\General Administration Tools All Users:IBM DB2\General Administration Tools All Users

IBM DB2\Information All Users:IBM DB2\Information All Users

IBM DB2\Monitoring Tools All Users:IBM DB2\Monitoring Tools All Users

IBM DB2\Set-up Tools All Users:IBM DB2\Set-up Tools All Users

Microsoft Visual C++ 6.0 All Users:Microsoft Visual C++ 6.0 All Users

Microsoft Visual C++ 6.0\Microsoft Visual C++ 6.0 Tools All Users:Microsoft Visual C++ 6.0\Microsoft Visual C++ 6.0 Tools All Users

Startup All Users:Startup All Users

Accessories VCLIENT30\Administrator:Accessories VCLIENT30\Administrator

Accessories\Accessibility VCLIENT30\Administrator:Accessories\Accessibility VCLIENT30\Administrator

Accessories\Entertainment VCLIENT30\Administrator:Accessories\Entertainment VCLIENT30\Administrator

Accessories\System Tools VCLIENT30\Administrator:Accessories\System Tools VCLIENT30\Administrator

Administrative Tools VCLIENT30\Administrator:Administrative Tools VCLIENT30\Administrator

Startup VCLIENT30\Administrator:Startup VCLIENT30\Administrator

[Startup Programs]

Program	Command	User Name	Location
synctime	synctime.bat	VCLIENT30\Administrator	Startup

[OLE Registration]

Object	Local Server
Sound (OLE2)	sndrec32.exe

Media Clip mplay32.exe

Video Clip mplay32.exe /avi

MIDI Sequence mplay32.exe /mid

Sound Not Available

Media Clip Not Available

Image Document "C:\Program Files\Windows NT\Accessories\ImageVue\KodakImg.exe"

WordPad Document "%ProgramFiles%\Windows NT\Accessories\WORDPAD.EXE"

Windows Media Services DRM Storage object Not Available

Bitmap Image mspaint.exe

[Internet Explorer 5]

[Following are sub-categories of this main category]

[Summary]

Item	Value
Version	5.00.3700.1000
Build	53700.1000
Product ID	51876-270-4893362-05373
Application Path	C:\Program Files\Internet Explorer
Language	English (United States)
Active Printer	Not Available
Cipher Strength	168-bit
Content Advisor	Disabled
IEAK Install	No

[File Versions]

File	Version	Size	Date	Path	Company
advapi32.dll	5.0.2195.6710	378 KB	6/19/2003 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation

advpack.dll	5.0.3502.6601	87 KB	6/19/2003 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
browsecl.dll	5.0.3700.6661	35 KB	6/19/2003 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
browseui.dll	5.0.3700.6661	789 KB	6/19/2003 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
ckcnv.exe	5.0.2189.1	9 KB	12/7/1999 8:00:00 AM	C:\WINNT\system32	Microsoft Corporation
comctl32.dll	5.81.3502.6601	538 KB	6/19/2003 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
crypt32.dll	5.131.2195.6661	468 KB	6/19/2003 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
ehnsig.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
iemigrat.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
iesetup.dll	5.0.3502.6601	57 KB	6/19/2003 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
iexplore.exe	5.0.2920.0	59 KB	12/7/1999 8:00:00 AM	C:\Program Files\Internet Explorer	Microsoft Corporation
imagehlp.dll	5.0.2195.6613	126 KB	6/19/2003 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
imghelp.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
inseng.dll	5.0.3502.6601	72 KB	6/19/2003 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
jobexec.dll	5.0.0.1	47 KB	12/7/1999 8:00:00 AM	C:\WINNT\system32	Microsoft Corporation
jscript.dll	5.1.0.8513	476 KB	6/19/2003 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
jsproxy.dll	5.0.2920.0	13 KB	12/7/1999 8:00:00 AM	C:\WINNT\system32	Microsoft Corporation
msaahtml.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
mshtml.dll	5.0.3700.6699	2299 KB	6/19/2003 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
msoss.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
msxml.dll	8.0.6730.0	502 KB	6/19/2003 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
occache.dll	5.0.3502.6601	86 KB	6/19/2003 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
ole32.dll	5.0.2195.6692	973 KB	6/19/2003 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
oleaut32.dll	2.40.4522.0	612 KB	6/19/2003 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
olepro32.dll	5.0.4522.0	160 KB	6/19/2003 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
rsabase.dll	5.0.2195.6619	129 KB	6/19/2003 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
rsaenh.dll	5.0.2195.6611	132 KB	6/19/2003 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
rsapi32.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
rsasig.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
schannel.dll	5.1.2195.6705	144 KB	6/19/2003 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
shdoc401.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
shdocvw.dll	5.0.3700.6668	1082 KB	6/19/2003 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
shell32.dll	5.0.3700.6705	2328 KB	6/19/2003 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
shlwapi.dll	5.0.3502.6601	283 KB	6/19/2003 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
url.dll	5.0.3502.6601	82 KB	6/19/2003 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
urlmon.dll	5.0.3700.6705	443 KB	6/19/2003 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
vbscript.dll	5.1.0.7426	428 KB	6/19/2003 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
webcheck.dll	5.0.3502.6601	252 KB	6/19/2003 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
win.com	5.0.2134.1	24 KB	12/7/1999 8:00:00 AM	C:\WINNT\system32	Microsoft Corporation
wininet.dll	5.0.3700.6713	456 KB	6/19/2003 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
winsock.dll	3.10.0.103	3 KB	12/7/1999 8:00:00 AM	C:\WINNT\system32	Microsoft Corporation
wintrust.dll	5.131.2195.6624	162 KB	6/19/2003 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
wsock.vxd	<File Missing>	Not Available	Not Available	Not Available	Not Available
wsock32.dll	5.0.2195.6603	21 KB	6/19/2003 1:05:04 PM	C:\WINNT\system32	Microsoft Corporation
wsock32n.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
[Connectivity]					

Item	Value
Connection Preference	Never dial
EnableHttp1.1	1
ProxyHttp1.1	0
LAN Settings	
AutoConfigProxy	wininet.dll
AutoProxyDetectMode	Disabled
AutoConfigURL	
Proxy	Disabled
ProxyServer	
ProxyOverride	

[Cache]

[Following are sub-categories of this main category]

[Summary]

Item	Value
Page Refresh Type	Automatic
Temporary Internet Files Folder	C:\Documents and Settings\Administrator\Local Settings\Temporary Internet Files
Total Disk Space	34710 MB
Available Disk Space	29029 MB
Maximum Cache Size	1084 MB
Available Cache Size	1084 MB

[List of Objects]

Program File	Status	CodeBase
No cached object information available		

[Content]

[Following are sub-categories of this main category]

[Summary]

Item	Value
Content Advisor	Disabled

[Personal Certificates]

Issued To	Issued By	Validity	Signature	Algorithm
Administrator	Administrator	11/3/2004 to 10/10/2104	sha1RSA	

[Other People Certificates]

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

[Publishers]

Name
No publisher information available

[Security]

Zone	Security Level
Local intranet	Medium-low
Trusted sites	Low
Internet	Medium
Restricted sites	High

Client Configuration Parameters

Microsoft Windows 2000 Client Registry Parameters

Client Configuration Parameters

COM+ Settings

```
tpccCom.tpcc_com.1:
  Activation:
    Enable Object Pooling selected
    Minimum Pool Size: 43
    Maximum Pool Size: 43
    Creating Timeout: 60,000
    Enable Just in Time Activation
  Concurrency:
    Concurrency Required
```

TPCC Application Registry Parameters

```
[HKEY_LOCAL_MACHINE\SOFTWARE\TPCC]
"dbType"="DB2"
"dIvyLogPath"="c:\inetpub\wwwroot\tpcc\dIvy"
"dIvyQueueLen"=dword:00004e20
>nullDB"=dword:00000000
"dbName"="tpcc"
"errorLogFile"="c:\inetpub\wwwroot\tpcc\errorLog.txt"
"htmlTraceLogFile"="c:\inetpub\wwwroot\tpcc\htmlTrace.txt"
"numUsers"=dword:00007530
"dbUserName"="Administrator"
"dbPassword"="tpcc"
"dbInterfacePath"="C:\inetpub\wwwroot\tpcc\db2glue.dll"
"dIvyThreads"=dword:0000000b
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo]
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo\Parameters]
"ListenBackLog"=dword:00000096
"DispatchEntries"=hex(7):4c,00,44,00,41,00,50,00,53,00,56,00,43,00,00,00,4c,00,44,00,41,00,50,00,53,00,56,00,43,00,58,00,00,00,00,00
```

```

"PoolThreadLimit"=dword:000000be
"ThreadTimeout"=dword:00015180

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo\Performance]
"Library"="infectrs.dll"
"Open"="OpenINFOPerformanceData"
"Close"="CloseINFOPerformanceData"
"Collect"="CollectINFOPerformanceData"
"Last Counter"=dword:00000842
"Last Help"=dword:00000843
"First Counter"=dword:00000802
"First Help"=dword:00000803
"Library Validation
Code"=hex:bc,b6,7c,11,e0,c1,c4,01,10,25,00,00,00,00,00,00
"WbemAdapFileTime"=hex:00,a0,38,ed,84,36,c3,01
"WbemAdapFileSize"=dword:00002510
"WbemAdapStatus"=dword:00000000

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\IISADMIN]
"Type"=dword:00000020
"Start"=dword:00000002
"ErrorControl"=dword:00000001
"ImagePath"=hex(2):43,00,3a,00,5c,00,57,00,49,00,4e,00,4e,00,54,00,5c,00,53,00,\
79,00,73,00,74,00,65,00,6d,00,33,00,32,00,5c,00,69,00,6e,00,65,00,74,00,73,\
00,72,00,76,00,5c,00,69,00,6e,00,65,00,74,00,69,00,6e,00,66,00,6f,00,2e,00,\
65,00,78,00,65,00,00,00
"DisplayName"="IIS Admin Service"
"DependOnService"=hex(7):52,00,50,00,43,00,53,00,53,00,00,00,50,00,72,00,6f,00,\
74,00,65,00,63,00,74,00,65,00,64,00,53,00,74,00,6f,00,72,00,61,00,67,00,65,\
00,00,00,00,00
"DependOnGroup"=hex(7):00,00
"ObjectName"="LocalSystem"
"Description"="Allows administration of Web and FTP services through the
Internet Information Services snap-in."
"FailureCommand"="\"C:\\WINNT\\System32\\iisreset.exe\" /fail=%1%"
"FailureActions"=hex:80,51,01,00,88,ca,0a,00,98,ca,0a,00,03,00,00,00,f4,ca,0a,\
00,03,00,00,00,01,00,00,00,03,00,00,00,01,00,00,00,03,00,00,00,01,00,00,00

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\IISADMIN\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,00,01,02,00,00,00,00,05,\
20,00,00,00,20,02,00,00,72,00,73,00,00,00,18,00,8d,01,02,00,01,01,00,00,00,\
00,00,05,0b,00,00,00,20,02,00,00,00,00,1c,00,fd,01,02,00,01,02,00,00,00,00,\
00,05,20,00,00,00,23,02,00,00,72,00,73,00,01,01,00,00,00,00,00,05,12,00,00,\
00,01,01,00,00,00,00,00,05,12,00,00,00

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\IISADMIN\Enum]
"0"="Root\LEGACY_IISADMIN\0000"
"Count"=dword:00000001
"NextInstance"=dword:00000001

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC]
"Type"=dword:00000020
"Start"=dword:00000002
"ErrorControl"=dword:00000001
"ImagePath"=hex(2):43,00,3a,00,5c,00,57,00,49,00,4e,00,4e,00,54,00,5c,00,53,00,\
79,00,73,00,74,00,65,00,6d,00,33,00,32,00,5c,00,69,00,6e,00,65,00,74,00,73,\
00,72,00,76,00,5c,00,69,00,6e,00,65,00,74,00,69,00,6e,00,66,00,6f,00,2e,00,\
65,00,78,00,65,00,00,00
"DisplayName"="World Wide Web Publishing Service"
"DependOnService"=hex(7):49,00,49,00,53,00,41,00,44,00,4d,00,49,00,4e,00,00,00,\
00,00
"DependOnGroup"=hex(7):00,00
"ObjectName"="LocalSystem"
"Description"="Provides Web connectivity and administration through the
Internet Information Services snap-in."

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\ASP]
"NOTE"="This is for backward compatibility only."

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\ASP\LanguageEngines]
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\ASP\LanguageEngines\PerlScript]
"Write"="$Response->write();"
"WriteBlock"="$Response->writeblock();"

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\ASP\Parameters]
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Parameters]
"MajorVersion"=dword:00000005
"MinorVersion"=dword:00000000
"InstallPath"="C:\\WINNT\\System32\\inetsrv"
"CertMapList"="C:\\WINNT\\System32\\inetsrv\\iisrmap.dll"
"AccessDeniedMessage"="Error: Access is Denied."
"Filter DLLs"=""
"LogFileDirectory"="C:\\WINNT\\System32\\LogFiles"
"AcceptExOutstanding"=dword:00000028

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Parameters\ADCLaunch]
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Parameters\ADCLaunch\AdvancedDataFactory]
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Parameters\ADCLaunch\RDSServer.DataFactory]

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Parameters\Script Map]
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Parameters\Virtual Roots]
"/"="c:\\inetpub\\wwwroot,,207"
"/Scripts"="c:\\inetpub\\scripts,,1"
"/IISHelp"="c:\\winnt\\help\\iishelp,,1"
"/IISAdmin"="C:\\WINNT\\System32\\inetsrv\\iisadmin,,1"
"/IISamples"="c:\\inetpub\\iisamples,,1"
"/MSADC"="c:\\program files\\common files\\system\\msadc,,1"
"/_vti_bin"="C:\\Program Files\\Common Files\\Microsoft Shared\\Web Server
Extensions\\40\\isapi,,1"
"/Printers"="C:\\WINNT\\web\\printers,,201"
"/tpcc"="c:\\inetpub\\wwwroot\\tpcc,,207"
"/Rpc"="C:\\WINNT\\System32\\RpcProxy,,4"

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Performance]
"Library"="w3ctr.dll"
"Open"="OpenW3PerformanceData"

```

```
"Close"="CloseW3PerformanceData"
"Collect"="CollectW3PerformanceData"
"Last Counter"=dword:000008e6
"Last Help"=dword:000008e7
"First Counter"=dword:00000844
"First Help"=dword:00000845
"Library Validation
Code"=hex:ba,71,6e,c7,ac,c1,c4,01,10,3d,00,00,00,00,00,00
"WbemAdapFileTime"=hex:00,a0,38,ed,84,36,c3,01
"WbemAdapFileSize"=dword:00001d10
"WbemAdapStatus"=dword:00000000
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Security]
"Security"=hex:01,00,14,80,a0,00,00,00,ac,00,00,00,14,00,00,00,30,00,00,00,02,\
00,1c,00,01,00,00,00,02,80,14,00,ff,01,0f,00,01,01,00,00,00,00,01,00,00,\
00,00,02,00,70,00,04,00,00,00,00,00,18,00,fd,01,02,00,01,01,00,00,00,00,05,\
05,12,00,00,00,74,00,6f,00,00,00,1c,00,ff,01,0f,00,01,02,00,00,00,00,05,\
20,00,00,00,20,02,00,00,72,00,73,00,00,00,18,00,8d,01,02,00,01,01,00,00,\
00,00,05,0b,00,00,00,20,02,00,00,00,00,1c,00,fd,01,02,00,01,02,00,00,00,00,\
00,05,20,00,00,00,23,02,00,00,72,00,73,00,01,01,00,00,00,00,05,12,00,00,\
00,01,01,00,00,00,00,05,12,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Enum]
"0"="Root\LEGACY_W3SVC\0000"
"Count"=dword:00000001
"NextInstance"=dword:00000001
```

RTE Input Parameters

IBM BenchMaster benchmark profile. DO NOT CHANGE THE SPACING IN THIS FILE!

70 ** Number of slaves (all must be defined directly below)

SEGMENT	MACHINE	LOG DIRECTORY	ODBC
WEBSERVER	DB SERVER	STARTWH	ENDWH
#USERS			
v10a	vrte10	c:\rtelogs tpcc	vclient10a node0 1
320	3200		
v10b	vrte10	c:\rtelogs tpcc	vclient10b node0 321
640	3200		
v10c	vrte10	c:\rtelogs tpcc	vclient10a node0 641
960	3200		
v10d	vrte10	c:\rtelogs tpcc	vclient10b node0 961
1280	3200		
v10e	vrte10	c:\rtelogs tpcc	vclient10a node0 1281
1600	3200		
v10f	vrte10	c:\rtelogs tpcc	vclient10b node0 1601
1920	3200		
v20a	vrte20	c:\rtelogs tpcc	vclient20a node0 1921
2240	3200		
v20b	vrte20	c:\rtelogs tpcc	vclient20b node0 2241
2560	3200		
v20c	vrte20	c:\rtelogs tpcc	vclient20a node0 2561
2880	3200		
v20d	vrte20	c:\rtelogs tpcc	vclient20b node0 2881
3200	3200		
v20e	vrte20	c:\rtelogs tpcc	vclient20a node0 3201
3520	3200		
v20f	vrte20	c:\rtelogs tpcc	vclient20b node0 3521
3840	3200		
v30a	vrte30	c:\rtelogs tpcc	vclient30a node0 3841
4160	3200		

v30b	vrte30	c:\rtelogs tpcc	vclient30b node0 4161
4480	3200		
v30c	vrte30	c:\rtelogs tpcc	vclient30a node0 4481
4800	3200		
v30d	vrte30	c:\rtelogs tpcc	vclient30b node0 4801
5120	3200		
v30e	vrte30	c:\rtelogs tpcc	vclient30a node0 5121
5440	3200		
v30f	vrte30	c:\rtelogs tpcc	vclient30b node0 5441
5760	3200		
v40a	vrte40	c:\rtelogs tpcc	vclient40a node0 5761
6080	3200		
v40b	vrte40	c:\rtelogs tpcc	vclient40b node0 6081
6400	3200		
v40c	vrte40	c:\rtelogs tpcc	vclient40a node0 6401
6720	3200		
v40d	vrte40	c:\rtelogs tpcc	vclient40b node0 6721
7040	3200		
v40e	vrte40	c:\rtelogs tpcc	vclient40a node0 7041
7360	3200		
v40f	vrte40	c:\rtelogs tpcc	vclient40b node0 7361
7680	3200		
v50a	vrte50	c:\rtelogs tpcc	vclient50a node0 7681
8000	3200		
v50b	vrte50	c:\rtelogs tpcc	vclient50b node0 8001
8320	3200		
v50c	vrte50	c:\rtelogs tpcc	vclient50a node0 8321
8640	3200		
v50d	vrte50	c:\rtelogs tpcc	vclient50b node0 8641
8960	3200		
v50e	vrte50	c:\rtelogs tpcc	vclient50a node0 8961
9280	3200		
v50f	vrte50	c:\rtelogs tpcc	vclient50b node0 9281
9600	3200		
v110a	vrte110	c:\rtelogs tpcc	vclient110a node1
9601	9920		
v110b	vrte110	c:\rtelogs tpcc	vclient110b node1
9921	10240		
v110c	vrte110	c:\rtelogs tpcc	vclient110a node1
10241	10560		
v110d	vrte110	c:\rtelogs tpcc	vclient110b node1
10561	10880		
v110e	vrte110	c:\rtelogs tpcc	vclient110a node1
10881	11200		
v110f	vrte110	c:\rtelogs tpcc	vclient110b node1
11201	11520		
v120a	vrte120	c:\rtelogs tpcc	vclient120a node1
11521	11840		
v120b	vrte120	c:\rtelogs tpcc	vclient120b node1
11841	12160		
v120c	vrte120	c:\rtelogs tpcc	vclient120a node1
12161	12480		
v120d	vrte120	c:\rtelogs tpcc	vclient120b node1
12481	12800		
v120e	vrte120	c:\rtelogs tpcc	vclient120a node1
12801	13120		
v120f	vrte120	c:\rtelogs tpcc	vclient120b node1
13121	13440		
v160a	vrte160	c:\rtelogs tpcc	vclient160a node1
13441	13760		
v160b	vrte160	c:\rtelogs tpcc	vclient160b node1
13761	14080		
v160c	vrte160	c:\rtelogs tpcc	vclient160a node1
14081	14400		
v160d	vrte160	c:\rtelogs tpcc	vclient160b node1
14401	14720		
v160e	vrte160	c:\rtelogs tpcc	vclient160a node1
14721	15040		

```

v160f  vrte160  c:\rtelogs  tpcc  vclient160b  node1
15041  15360  3200
v140a  vrte140  c:\rtelogs  tpcc  vclient140a  node1
15361  15680  3200
v140b  vrte140  c:\rtelogs  tpcc  vclient140b  node1
15681  16000  3200
v140c  vrte140  c:\rtelogs  tpcc  vclient140a  node1
16001  16320  3200
v140d  vrte140  c:\rtelogs  tpcc  vclient140b  node1
16321  16640  3200
v140e  vrte140  c:\rtelogs  tpcc  vclient140a  node1
16641  16960  3200
v140f  vrte140  c:\rtelogs  tpcc  vclient140b  node1
16961  17280  3200
v150a  vrte150  c:\rtelogs  tpcc  vclient150a  node1
17281  17600  3200
v150b  vrte150  c:\rtelogs  tpcc  vclient150b  node1
17601  17920  3200
v150c  vrte150  c:\rtelogs  tpcc  vclient150a  node1
17921  18240  3200
v150d  vrte150  c:\rtelogs  tpcc  vclient150b  node1
18241  18560  3200
v150e  vrte150  c:\rtelogs  tpcc  vclient150a  node1
18561  18880  3200
v150f  vrte150  c:\rtelogs  tpcc  vclient150b  node1
18881  19200  3200
v10g   vrte10   c:\rtelogs  tpcc  vclient10a node0  19201
19305  1050
v20g   vrte20   c:\rtelogs  tpcc  vclient20a node0  19306
19410  1050
v30g   vrte30   c:\rtelogs  tpcc  vclient30a node0  19411
19515  1050
v40g   vrte40   c:\rtelogs  tpcc  vclient40a node0  19516
19620  1050
v50g   vrte50   c:\rtelogs  tpcc  vclient50a node0  19621
19725  1050
v110g  vrte110  c:\rtelogs  tpcc  vclient110a  node1
19726  19780  550
v120g  vrte120  c:\rtelogs  tpcc  vclient120a  node1
19781  19835  550
v140g  vrte140  c:\rtelogs  tpcc  vclient140a  node1
19836  19890  550
v150g  vrte150  c:\rtelogs  tpcc  vclient150a  node1
19891  19945  550
v160g  vrte160  c:\rtelogs  tpcc  vclient160a  node1
19946  20000  550

```

1000 ** Connect rate - rate users log in to the database (users per minute)

250 ** Run rate- rate users ramp in (users per minute)

1 *** Ramp-in type (0 = linear, 1 = 5 step descending rate)"

1 *** Web client (ignored for 2-tier; 0 = Microsoft's web client, 1 = IBM pSeries web client)"

20000 ** Total number of warehouses

0 *** Run type (0 = 3-tier, 1 = 2-tier)"

173 *** C_LOAD (0-255) - NURAND ""C"" value that WAS used for customer last name generation during database LOAD, usually 123 for SQL Server"

88 *** C_RUN (0-255) - NURAND ""C"" value to be used for customer last name generation when running. abs(C_LOAD - C_RUN) must be 65 to 119, inclusive, but not 96 or 112."

208 *** C_C_ID (0-1023) - NURAND ""C"" value to be used for customer ID generation when running"

208 *** C_OL_I_ID (0-8191) - NURAND ""C"" value to be used for orderline item ID generation when running" administrator ** Database user name

tpcc ** Database password

TOTAL	NEWORDPAYMENT	DELI	STCKLVLORDSTAT
0	44950	43020	4010
Transaction mix percentages (must add to 100,000)"			
0	12030	12030	5030
think times (milliseconds)			
0	0	0	0
think times (milliseconds)			
0	18000	3000	2000
times (milliseconds)			
0	0	0	0
times (milliseconds)			
0	5000	5000	20000
percentile values (milliseconds)			
0	100	100	100
painting menu delay (milliseconds)			
0	100	100	100
painting response time delay (milliseconds)			

2000 ** 90th percentile value for menu transactions (milliseconds)

Appendix D: 60-Day Space

60-Day Space Computation

All data sizes in MB unless otherwise stated

Warehouses	20,000
Measured TpmC	250,975

Table	Rows	Table	Index	5% Space	Total Space
Warehouse	20,000		9	0	10
District	200,000		38	0	39
Item	100,000		10	0	11
Stock	2,000,000,000	651,150		0	683,708
Customer	600,000,000	468,875	29,100	24,899	522,874
New-Order	180,000,000	7,025		0	7,025
Orders	600,000,000	22,204	17,000	0	39,204
Order-Line	9,000,000,000	591,224		0	591,224
History	600,000,000	37,400		0	37,400

Free Space	289,302	<u>30 Minute log Computations</u>	
Dynamic Space	650,829	Log Written (KB)	17,821,087
Static Space	1,230,666	Total New-Order Txns	7,529,250
Daily Growth	130,673	Log Written per New-Order (KB)	2.37
Daily Spread	93,292		

Data Storage Requirement

60 Days (MB)	14,668,600
60 Days (GB)	14,325

Log Storage Requirement

8 Hours (GB)	271.93
--------------	--------

Disk Sizing

Disk Type	Formatted		SUT		Priced	
	Capacity (GB)	# of Disks	Capacity (GB)	# of Disks	Capacity (GB)	# of Disks
DB DS4500 36.4GB	36.40	500	18,200	500	18,200	500
LOG SCSI RAID1	36.40	20	364	20	364	20
OS SCSI 36GB	36.40	1	36	1	36	1

Total Capacity	18,600
-----------------------	--------

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

May 26, 2005

IBM Corporation
Chris King
3079 Cornwallis Road
Durham, NC 27709

Ms. King:

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

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

Part Number	Description	Unit Price	Quantity	Price
P72-00264	Windows Server 2003, Enterprise x64 Edition <i>Server License Only - No CALs</i> <i>Discount Schedule: No Level</i> <i>Unit Price reflects a 42% discount from the retail unit price of \$3,999.</i>	\$2,334	1	\$2,334
C11-00821	Windows 2000 Server <i>Server License Only - No CALs</i> <i>Discount Schedule: No Level</i> <i>Unit Price reflects a 8% discount from the retail unit price of \$799.</i>	\$738	10	\$7,380
254-00170	Visual C++ Standard Edition <i>Discount Schedule: No Discounts Applied</i>	\$109	1	\$109
	Microsoft Problem Resolution Services <i>Professional Support</i> <i>(1 incident)</i>	\$245	1	\$245

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: PCchki0526055751.

Please include this Reference ID in any correspondence regarding this price quote.



> My Shopping Cart

MY SHOPPING CART

MY SAVED ITEMS

[Returns](#) | [Privacy](#) | [Security](#)

Clear Cart Move Cart to Wish List Save Cart Print Cart Email Cart

Network - Cables

Qty.	Product Description	Unit Price	Savings	Total Price
<input type="text" value="1"/>	AMC CC5E-B14B 14 FT Cat 5E Blue Cat 5E Blue Cable - OEM Model #: CC5E-B14B	\$3.29		\$3.29
<input type="button" value="Update"/> Remove Save Move To Wishlist				

Switches

Qty.	Product Description	Unit Price	Savings	Total Price
<input type="text" value="1"/>	D-Link DGS-1008TL 10/100/1000Mbps Unmanaged Gigabit Switch - Retail Model #: DGS-1008TL <i>**This item is warranted through the product manufacturer only.</i>	\$189.00		\$189.00
<input type="button" value="Update"/> Remove Save Move To Wishlist				

Subtotal: \$192.29

Shipping: \$0.00

Shipping

Zip Code:

*Enter your Zip Code and select a shipping option to determine your shipping cost.

Redeem Gift Certificates

Gift Certificates \$0.00

Claim Code:

Security Code:

[Click here to add another Gift Certificate.](#)

Apply Promo Code

Promo Code:

Total(before tax): \$192.29

If you're experiencing problems with your shopping cart, please click here and try again.

Note: Your shopping cart will be emptied.

Important Ordering Information

- Newegg.com will not be responsible for typographical or other errors or omissions regarding prices or other information.
- All sales are subject to Newegg.com's Terms and Conditions of Sale.
- Newegg.com reserves the right to refuse any order.
- All parts come with the original manufacturers warranty.
- Newegg.com offers No technical support.