



TPC Benchmark™ C
Full Disclosure Report

NEC Express5800/1160Xe (8 SMP)

**with IBM DB2 Universal Database 8.2
and
Microsoft® Windows® Server 2003, Datacenter Edition
for 64-bit Itanium-based Systems**

**First Edition
Submitted for Review
January 25, 2005**

NEC, the Sponsor of this benchmark test, believes that the information in this document is accurate as of the publication date. The information in this document is subject to change without notice. The Sponsor assumes no responsibility for any errors that may appear in this document. The pricing information in this document is believed to accurately reflect the current prices as of the publication date. However, the Sponsor provides no warranty of the pricing information in this document. Benchmark results are highly dependent upon workload, specific application requirements, and system design and implementation. Relative system performance will vary as a result of these and other factors. Therefore, TPC Benchmark™ C should not be used as a substitute for a specific customer application benchmark when critical capacity planning and/or product evaluation decisions are contemplated.

All performance data contained in this report were obtained in a rigorously controlled environment. Results obtained in other operating environments may vary significantly. NEC does not warrant or represent that a user can or will achieve similar performance expressed in transactions per minute (tpmC) or normalized price/performance (\$/tpmC). No warranty of system performance or price/performance is expressed or implied in this report.

Copyright 2005 NEC Corporation.

All rights reserved.

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

Printed in USA, 2005

NEC and Express5800 are registered trademarks of **NEC** Corporation.

IBM, DB2, DB2 UDB, DB2 Universal Database are trademarks of **International Business Machines** Corporation in the United States and/or other countries

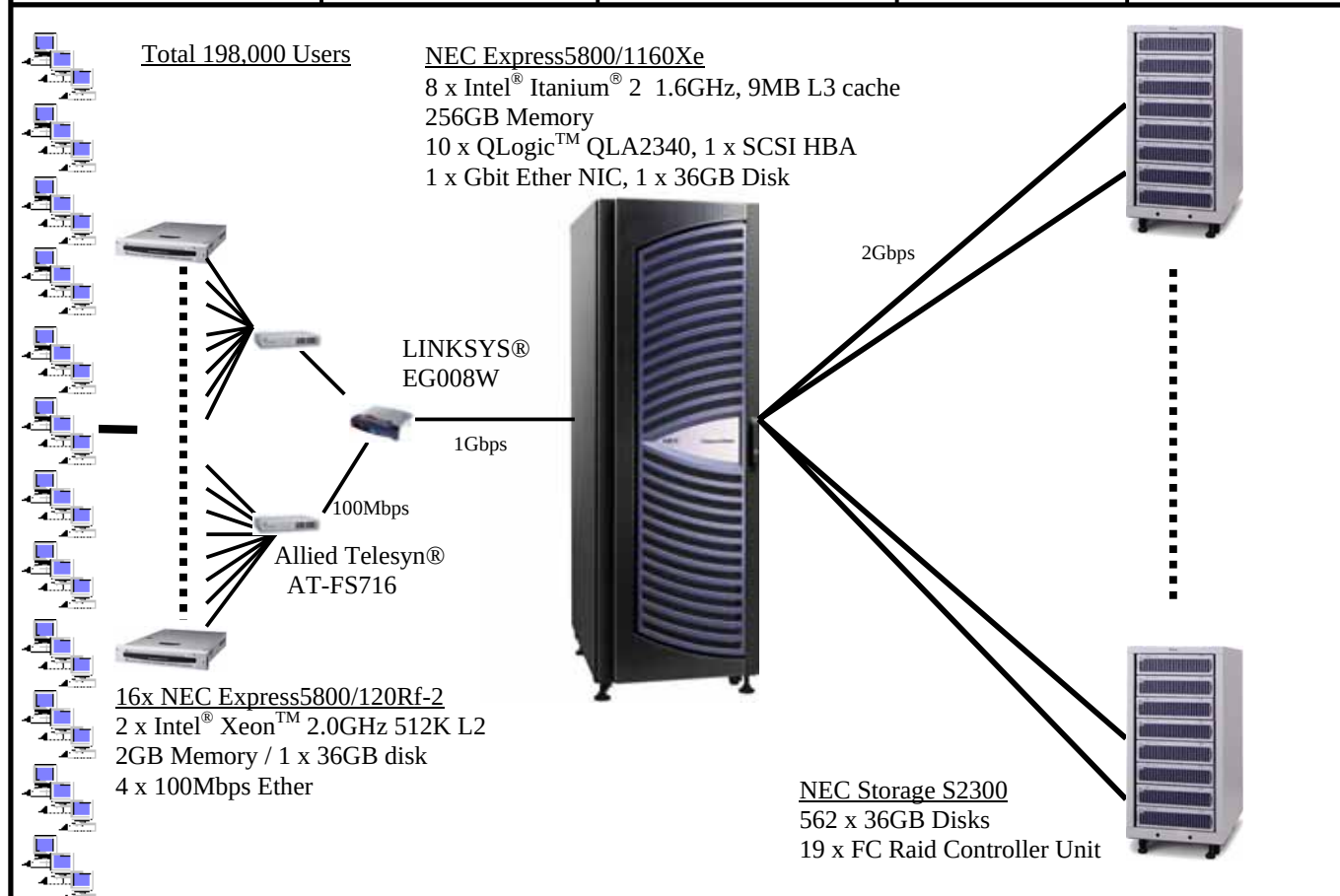
Microsoft®, Windows® Server 2003 is registered trademarks of **Microsoft®** Corporation.

TPC Benchmark, TPC-C and tpmC are trademarks of the **Transaction Processing Performance Council**.

Intel®, Itanium® and Xeon™ are registered trademarks and trademark of **Intel®** Corporation.

Other product names mentioned in this document may be trademarks and/or registered trademarks of their respective companies.

		NEC Express5800/1160Xe (8 SMP) C/S with Express5800/120Rf-2		TPC-C Rev.5.3 Reported Date January 25, 2005	
Total System Cost		TPC-C Throughput	Price/Performance		Availability Date
\$1,356,338 (USD)		247,650 tpmC	\$5.48 per tpmC		July 24, 2005
Processors		Database Manager	Operating System	Other Software	Number of Users
8 Intel® Itanium® 2 1.6GHz for Server 32 Intel® Xeon™ 2.0GHz for Clients		IBM DB2 UDB 8.2	Microsoft® Windows® Server 2003, Datacenter Edition for 64-bit Itanium-based Systems	Windows® 2000 Server Microsoft® VC++ Microsoft® COM+	198,000



System Component	Server		Each Client	
Processors	8	Intel® Itanium® 2 1.6GHz	2	Intel® Xeon™ 2.0GHz
Cache		9MB L3 Cache		512KB L2 Cache
Memory		256GB		2GB
Disk Controllers	10 1	QLogic™ QLA2340 SCSI HBA	1	On-board SCSI
Disk Drives	1 562	36GB 36GB	1	36GB
Total Storage		18,426GB		36GB
Others	1 1	DVD-ROM Drive 1Gbps Ether NIC	1 2 2	CD-ROM Drive On-board Ether controller 100Mbps Ether NIC

Numerical Quantities Summary

MQTh, Computed Maximum Qualified Throughput						247,650 tpmC	
<u>Response Times(in seconds)</u>		<u>Average</u>		<u>Maximum</u>		<u>90%</u>	
New-Order		0.35		29.44		0.94	
Payment		0.34		21.94		0.93	
Delivery(interactive portion)		0.12		3.57		0.11	
Stock-Level		0.33		17.64		0.91	
Order-status		0.35		16.07		0.93	
Delivery(deferred portion)		0.15		12.66		0.19	
Menu		0.12		73.29		0.11	
Response time delay added for emulated components						0.1	
<u>Transaction Mix , in percent of total transaction</u>							
New-Order						44.80%	
Payment						43.08%	
Delivery						4.04%	
Stock-Level						4.04%	
Order-Status						4.04%	
<u>Keying/Think Times (in seconds)</u>		<u>Average</u>		<u>Min</u>		<u>Max</u>	
New-Order		18.01	12.07	18.01	0.00	18.01	120.7
Payment		3.01	12.07	3.01	0.00	3.01	120.7
Delivery		2.01	5.07	2.01	0.00	2.01	50.7
Stock-Level		2.01	5.08	2.01	0.00	2.01	50.7
Order-Status		2.01	10.09	2.01	0.00	2.01	100.7
<u>Test Duration</u>							
Ramp-up time						5400 seconds	
Measurement interval						9000 seconds	
Number of checkpoints						N/A	
Checkpoint interval						N/A	
Number of transactions (all types) completed in measurement interval						82,934,549	

ABSTRACT	8
TPC BENCHMARK TM C METRICS	8
STANDARD AND EXECUTIVE SUMMARY STATEMENTS	8
AUDITOR	8
PREFACE	9
TPC BENCHMARK TM C OVERVIEW	9
DOCUMENT STRUCTURE	9
GENERAL ITEMS	11
ORDER AND TITLES	11
SUMMARY STATEMENT	11
NUMERICAL QUANTITIES SUMMARY	11
APPLICATION PROGRAM	11
SPONSOR	11
PARAMETERS AND OPTIONS	12
CONFIGURATION DIAGRAMS	12
MEASURED CONFIGURATION	13
PRICED SYSTEM CONFIGURATION	14
CLAUSE 1 : LOGICAL DATABASE DESIGN AND RELATED ITEMS	15
TABLE DEFINITIONS	15
TABLE ORGANIZATION	15
INSERT AND DELETE OPERATIONS	15
DISCLOSURE OF PARTITIONING	15
REPLICATION OF TABLES	15
ADDITIONAL AND/OR DUPLICATED ATTRIBUTES IN ANY TABLE	15
CLAUSE 2 : TRANSACTION AND TERMINAL PROFILES RELATED ITEMS	16
RANDOM NUMBER GENERATION	16
TERMINAL INPUT/OUTPUT SCREEN LAYOUT	16
TERMINAL FEATURE VERIFICATION	16
PRESENTATION MANAGER OR INTELLIGENT TERMINAL	16
TRANSACTION PROFILES	16
TRANSACTION MIX	16
QUEUEING MECHANISM	17
CLAUSE 3 : TRANSACTION AND SYSTEM PROPERTIES RELATED ITEMS	18
TRANSACTION SYSTEM PROPERTIES (ACID)	18
ATOMICITY TESTS	18
<i>Completed Transactions</i>	18
<i>Aborted Transactions</i>	18
CONSISTENCY TESTS	18
ISOLATION TESTS	19
DURABILITY TESTS	19
<i>Instantaneous Interruption and Loss of Memory</i>	19
<i>Loss of Log and Data Disk</i>	19
<i>Loss of mirrored write-back cache</i>	19
CLAUSE 4 : SCALING AND DATABASE POPULATION RELATED ITEMS	21
INITIAL CARDINALITY OF TABLES	21
CONSTANT VALUE FOR THE NURAND FUNCTION	21
DISTRIBUTION OF TABLES AND LOGS	22
TYPE OF DATABASE	23
DATABASE MAPPING	23
60-DAYS SPACE	23
CLAUSE 5 : PERFORMANCE METRICS AND RESPONSE TIME RELATED ITEMS	24

THROUGHPUT	24
RESPONSE TIMES.....	24
KEYING AND THINK TIMES	24
RESPONSE TIME FREQUENCY DISTRIBUTION CURVES	25
RESPONSE TIME VERSUS THROUGHPUT CURVE	27
NEW-ORDER THINK TIME FREQUENCY DISTRIBUTION.....	28
NEW-ORDER THROUGHPUT VS. ELAPSED TIME.....	28
STEADY STATE	29
WORK PERFORMED DURING STEADY STATE	29
REGULATION OF TRANSACTION MIX	29
TRANSACTION STATISTICS	29
CHECKPOINT COUNT AND LOCATION	29
CLAUSE 6 : SUT, DRIVER, AND COMMUNICATION DEFINITION RELATED ITEMS	29
DESCRIPTIONS OF RTE.....	29
LOSS OF TERMINAL CONNECTIONS.....	30
EMULATED COMPONENTS	30
FUNCTIONAL DIAGRAMS AND DETAIL OF DRIVER SYSTEM.....	30
NETWORK CONFIGURATIONS AND DRIVER SYSTEM.....	30
NETWORK BANDWIDTH.....	30
OPERATOR INTERVENTION	30
CLAUSE 7 : PRICING RELATED ITEMS	30
HARDWARE AND SOFTWARE COMPONENTS.....	30
AVAILABILITY	31
THROUGHPUT, AND PRICE PERFORMANCE	31
COUNTRY SPECIFIC PRICING	31
USAGE PRICING.....	31
SYSTEM PRICING	31
CLAUSE 8 : AUDIT RELATED ITEMS.....	32
AUDITOR'S REPORT.....	32
AVAILABILITY OF THE FULL DISCLOSURE REPORT	32
AUDITOR'S LETTER	33
<u>APPENDIX A : APPLICATION SOURCE CODE</u>	<u>35</u>
<u>APPENDIX B : DATABASE DESIGN.....</u>	<u>214</u>
<u>APPENDIX C : TUNABLE PARAMETERS</u>	<u>295</u>
<u>APPENDIX D : SPACE CALCULATION</u>	<u>397</u>
<u>APPENDIX E : PRICE QUOTATION</u>	<u>398</u>

Abstract

This report documents the compliance of NEC Corporation's TPC Benchmark™ C tests on the NEC Express5800/1160Xe client/server system with version 5.3 of the TPC Benchmark C Standard Specification. 16 Clients (NEC Express5800/120Rf-2) were used as the front-end clients.

The operating system and the DBMS used on the server were Microsoft® Windows® Server 2003, Datacenter Edition and IBM DB2 Universal Database 8.2. The operating system on the clients was Microsoft® Windows® 2000 Server SP2. Those clients ran Microsoft® IIS server 5.0 and COM+.

Two standard metrics, transaction-per-minute-C(tpmC) and price per tpmC(\$/tpmC) are reported, in accordance with the TPC Benchmark™ C Standard. The independent auditor's report by Francois Raab appears at the end of this report.

TPC Benchmark™ C Metrics

The standard TPC Benchmark™ C metrics, tpmC (transactions per minute), price per tpmC (three year capital cost per measured tpmC) are reported.

System	SW	Total System Cost	TpmC	\$ per tpmC	Availability Date
NEC Express5800 /1160Xe	IBM DB2 Universal Database 8.2 Microsoft® Windows® Server 2003, Datacenter Edition	\$1,356,338 (USD)	247,650	\$5.48	July 24, 2005

Standard and Executive Summary Statements

The following pages contain executive summary of results for this benchmark.

Auditor

The benchmark configuration, environment and methodology were audited by Francois Raab of Info Sizing Inc. to verify compliance with the relevant TPC specifications.

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 Specifications Version 5.3

TPC Benchmark™ C Overview

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

TPC Benchmark™ C (TPC-C) is an 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 such 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 TPC-C results must include the tpmC rate, the associated price-per-tpmC, and the availability date of the priced configuration.

Although these specifications express implementation in terms of a relational data model with conventional locking scheme, the database may be implemented using any commercially available database management system (DBMS), database server, file system, or other data repository that provides a functionally equivalent implementation. The terms “table”, “row”, and “column” are used in this document only as examples of logical data structures.

TPC-C uses terminology and metrics that are similar to other benchmarks, originated by the TPC or others. Such similarity in terminology does not in any way imply that TPC-C results are comparable to other benchmarks. The only benchmark results comparable to TPC-C are other TPC-C results conformant with the same revision.

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.

Benchmark sponsors are permitted several possible system designs, insofar as they adhere to the model described and pictorially illustrated in Clause 6. A Full Disclosure Report of the implementation details, as specified in Clause 8, must be made available along with the reported results.

Document Structure

This TPC Benchmark™ C Full Disclosure Report is organized as follows:

- The main body of the document lists each item in Clause 8 of the TPC-C Standard and explains how each requirement is satisfied.
- Appendix A contains the source code of the TPC-C application code used to implement the TPC-C transactions.
- Appendix B contains the database definition and population code used in the tests.
- Appendix C contains the tunable parameters used in the TPC-C tests.
- Appendix D contains space calculation table.
- Appendix E contains third-party price quotations.

TPC Benchmark™ C Full Disclosure

The TPC Benchmark™ C Standard Specification requires test sponsors to publish, and make available to the public, a full disclosure report for the results to be considered compliant with the Standard. The required contents of the full disclosure report are specified in Clause 8. This report is intended to satisfy the Standard's requirement for full disclosure. It documents the compliance of the benchmark tests with each item listed in Clause 8 of the TPC Benchmark™ C Standard Specification.

In the Standard Specification, the main headings in Clause 8 are keyed to the other clauses. The headings in this report use the same sequence, so that they correspond to the titles or subjects referred to in Clause 8.

Each section in this report begins with the text of the corresponding item from Clause 8 of the Standard Specification, printed in italic type. The plain text that follows explains how the tests comply with the TPC Benchmark™ C requirement. In sections where Clause 8 requires extensive listings, the section refers to the appropriate appendix at the end of this report.

General Items

Order and titles

The order and titles of sections in the Test Sponsor's Full Disclosure report must correspond with the order and titles of sections from the TPC-C standard specification (i.e., this document). The intent is to make it as easy as possible for readers to compare and contrast material in different Full Disclosure reports.

The order and titles of sections in this report correspond with that of the TPC-C standard specification.

Summary Statement

The TPC Executive Summary Statement must be included near the beginning of the Full Disclosure report.

The TPC Executive Summary Statement is included at the beginning of this report.

Numerical Quantities Summary

The numerical quantities listed below must be summarized near the beginning of the Full Disclosure report :

- *measurement interval in minutes,*
- *number of checkpoints in the measurement interval,*
- *longest checkpoint interval in minutes,*
- *number of transactions (all types) completed within the measurement interval,*
- *computed Maximum Qualified Throughput in tpmC,*
- *ninetieth percentile, average and maximum response times for the New-Order, Payment, Order-Status, Stock-Level, Delivery (deferred and interactive) and Menu transactions,*
- *time in seconds added to response time to compensate for delays associated with emulated components,*
- *percentage of transaction mix for each transaction type.*

These numerical quantities are summarized at the beginning of this report.

Application Program

The application program (as defined in 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 the application source codes used in the TPC-C benchmark.

Sponsor

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

This benchmark test was sponsored by NEC Corporation. NEC has authorized NEC Corp. to publish TPC-C performance and price/performance results for the NEC Express5800/1160Xe. Price quotations contained in Appendix E correspond to the NEC Express5800/1160Xe server.

Parameters and Options

Settings must be provided for all customer-tunable parameters and options which 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.

Appendix C contains the tunable parameters used in the TPC-C tests.

Configuration Diagrams

Diagrams of both measured and priced configurations must be provided, accompanied by a description of the differences. This includes, but is not limited to:

- Number and type of processors
- Size of allocated memory, and any specific mapping/partitioning of memory unique to the test.
- Number and type of disk units (and controllers, if applicable).
- Number of channels or bus connections to disk units, including their protocol type.
- Number of LAN (e.g., Ethernet) connections, including routers, workstations, terminals, etc., that were physically used in the test or are incorporated into the pricing structure (see Clause 8.1.8).
- Type and the run-time execution location of software components (e.g., DBMS, client processes, transaction monitors, software drivers, etc.).

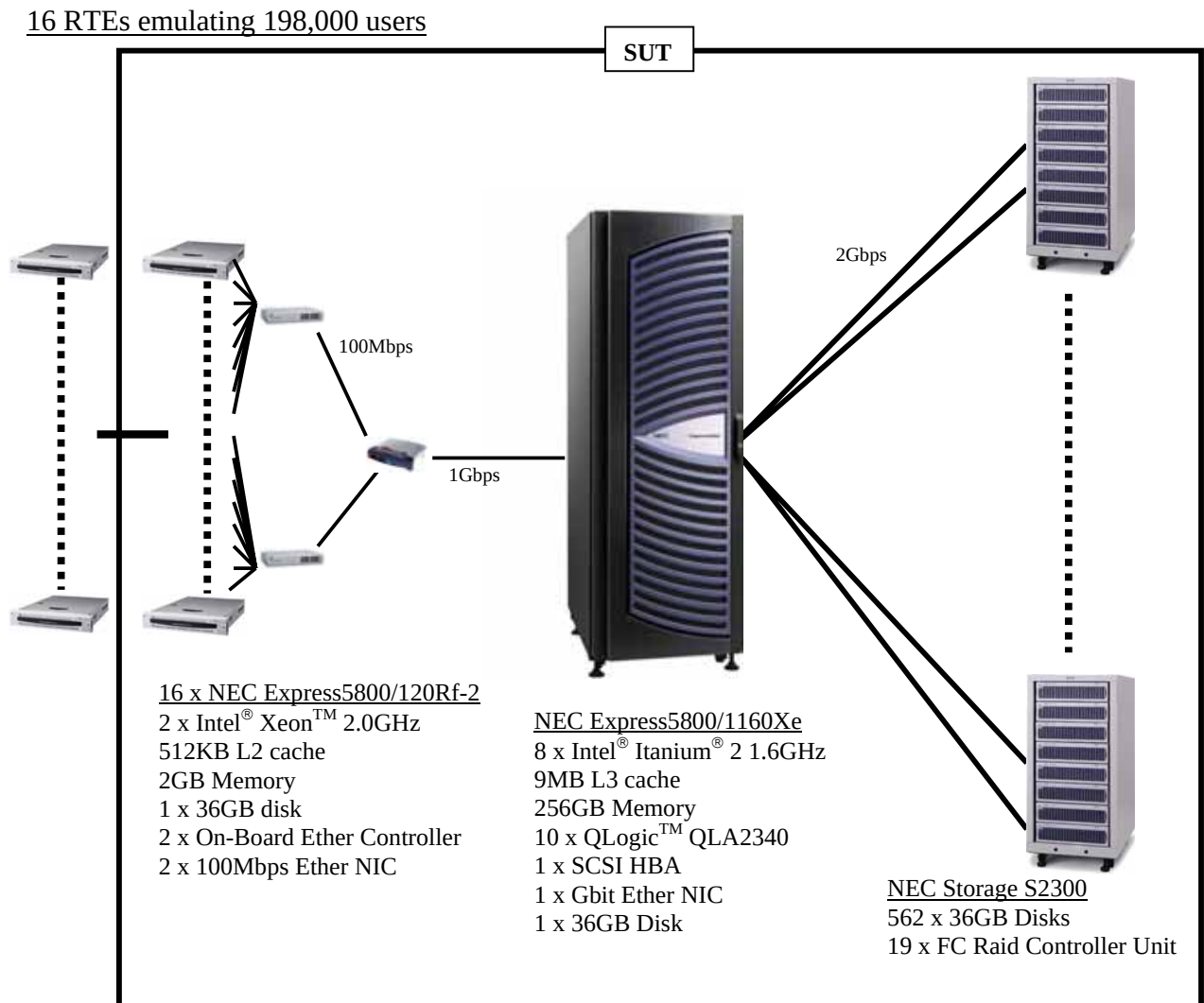
Figure 1.1 shows the measured configuration diagram.

Figure 1.2 shows the priced configuration diagram.

Measured Configuration

The following figure represents the measured configuration. The benchmark system used a remote terminal emulator (RTE) to initiate transactions and measure response times of transactions, as well as record various data for each transaction.

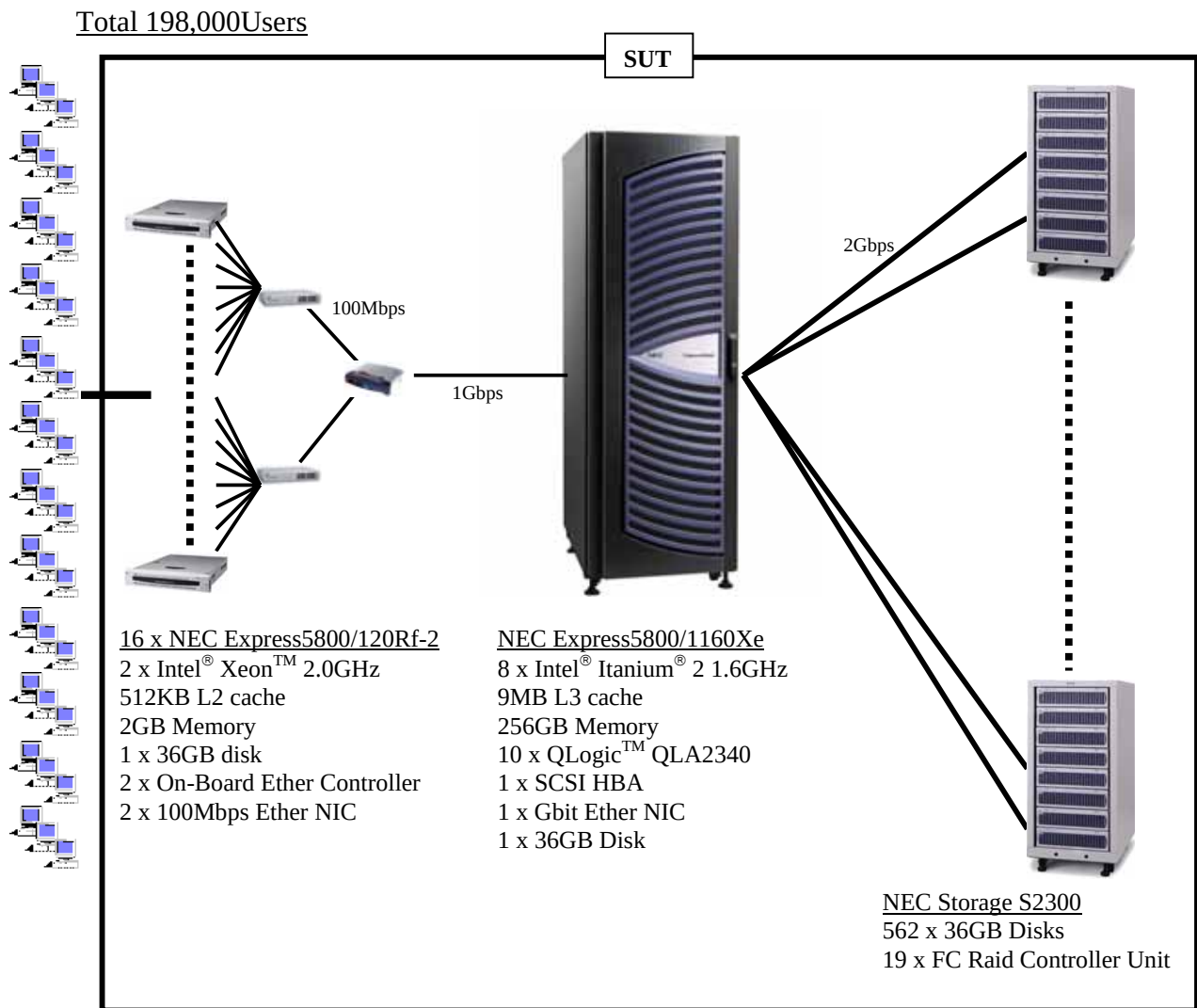
Figure 1.1 Express5800/1160Xe, Measured Configuration Diagram



Priced System Configuration

The following figure depicts the priced system, whose cost determines the normalized price per tpmC reported for the test.

Figure1.2: Express5800/1160Xe, Priced Configuration Diagram



Clause 1 : Logical Database Design and 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.

Table Organization

The physical organization of tables and indices, within the database, must be disclosed.

Appendix B contains the code used to define the physical organization of tables and indices

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 entire benchmark.

Disclosure of Partitioning

While there are a 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 but ITEM were horizontally partitioned into multiple tables. Each (STOCK, CUSTOMER, ORDERS, ORDERLINE, WAREHOUSE, DISTRICT, NEWORDER, HISTORY) table partition contains data associated with a range of 1650 warehouses. For each partitioned table, a view was created over all table partitions to provide full transparency of data manipulation.

Replication of Tables

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

No tables were replicated in this benchmark test.

Additional and/or Duplicated Attributes in any Table

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 duplications or 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 described.

Random numbers generation in RTE and DB generator were verified by auditor independently.

Terminal Input/Output Screen Layout

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

All screen layouts were verified to be exactly followed the specification by auditor.

Terminal feature 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 for the demonstration in 8.1.3.3 must be disclosed and commercially available (including supporting software and maintenance).

Each of five transaction types was tested by the auditor. The auditor verified that all the features specified in Clause 2.2.2.4 were provided.

Presentation Manager or Intelligent Terminal

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

Application code running on the client machines implemented the TPC-C user interface. No presentation manager software or intelligent terminal features were used. The source code for the applications is listed in Appendix A.

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.*

Table 1 shows the numerical quantities required by Clause 8.1.3.5 through 8.1.3.10.

Transaction Mix

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

Table 1 shows the mix of transaction types seen by the SUT during the reported measurement interval.

Following table summarizes the data required for disclosure in Clause 8.1.3.5 through 8.1.3.11

Table 1 Transaction Statistics

Statistic		Value
New Order	Home warehouse order lines	98.99%
	Remote warehouse order lines	1.01%
	Rolled back transactions	0.99%
	Average items per order	9.9
Payment	Home warehouse payments	85.02%
	Remote warehouse payments	14.98%
	Accessed by last name	59.98%
Order Status	Accessed by last name	59.96%
Delivery	Skipped deliveries	0
Transaction Mix	New Order	44.80%
	Payment	43.08%
	Delivery	4.04%
	Stock Level	4.04%
	Order Status	4.04%

Queuing Mechanism

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

The client application processes submitted delivery transactions to server software running on the client machines. These delivery servers were responsible for processing queued deliveries to submit to the database server.

The source code is listed in Appendix A.

Clause 3 : Transaction and System Properties Related Items

Transaction System Properties (ACID)

The results of the ACID tests 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.

The TPC Benchmark™ C Standard Specification defines a set of transaction processing system properties that a system under test (SUT) must support during the execution of the benchmark. Those properties are Atomicity, Consistency, Isolation and Durability (ACID). This section quotes the specification definition of each of those properties and describes the tests done as specified and monitored by the auditor, to demonstrate compliance.

Atomicity Tests

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.

Completed Transactions

Perform the Payment transaction for a randomly selected warehouse, district and customer (by customer number as specified in Clause 2.5.1.2) and verify that the records in the CUSTOMER, DISTRICT and WAREHOUSE tables have been changed appropriately.

The value of w_ytd, d_ytd, c_balance, c_ytd_payment and c_payment_cnt of a randomly selected warehouse, district, and customer were retrieved. The Payment transaction was executed on the same warehouse, district, and customer. The transaction was committed. The values w_ytd, d_ytd, c_balance, c_ytd_payment, and c_payment_cnt were retrieved again. It was verified that all values had been changed appropriately.

Aborted Transactions

Perform the Payment transaction for a randomly selected warehouse, district and customer (by customer number as specified in Clause 2.5.1.2) and substitute a ROLLBACK of the transaction for the COMMIT of the transaction. Verify that the records in CUSTOMER, DISTRICT and WAREHOUSE tables have Not been changed.

The value of w_ytd, d_ytd, c_balance, c_ytd_payment and c_payment_cnt of randomly selected warehouse, district, and customer were retrieved. The Payment transaction was executed on the same warehouse, district, and customer. The transaction was rolled back. The values of w_ytd, d_ytd, c_balance, c_ytd_payment, c_payment_cnt were retrieved again. It was verified that none of the values had changed.

Consistency Tests

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.

The specification defines 12 consistency conditions of which the following four are required to be explicitly demonstrated:

1. The sum of balances (d_ytd) for all Districts within a specific Warehouse is equal to the balance (w_ytd) of that Warehouse.
2. For each District within a Warehouse, the next available Order ID (d_next_o_id) minus one is equal to the most recent Order ID [max(o_id)] for the Order table associated with the preceeding District and Warehouse. Additionally, that same relationship exists for the most recent Order ID [max(o_id)] for the New Order table associated with the same District and Warehouse. Those relationships can be illustrated as follows:
$$d_next_o_id - 1 = max(o_id) = max(no_o_id)$$

where $(d_w_id = o_w_id = no_w_id)$ and $(d_id = o_d_id = no_d_id)$
3. For each District within a Warehouse, the value of the most recent Order ID [max(no_o_id)] minus the first Order ID [min(no_o_id)] plus one, for the New Order table associated with the District and Warehouse equals the number of rows in that New Order table. That relationship can be illustrated as follows:
$$max(no_o_id) - min(no_o_id) + 1 = \text{number of rows in New Order for the Warehouse/District}$$
4. For each District within a Warehouse, the sum of Order Line counts [sum(o_ol_cnt)] for the Order table associated with the District equals the number of rows in the Order Line table associated with the same District. That relationship can be illustrated as follows:

sum(o_ol_cnt) = number of rows in the Order Line table for the Warehouse/District

An RTE driven run was executed against a freshly loaded database. After the run the 4 consistency conditions defined above were tested using a script to issue queries to the database. All queries showed that the database was still in a consistent state.

Isolation Tests

Sufficient conditions must be enabled at either the system or application level to ensure the required isolation level is obtained.

Isolation tests one through nine were executed using shell scripts to issue queries to the database. Each script included timestamps to demonstrate the concurrency of operations. The results of the queries were captured to files. The captured files were verified to demonstrate the required isolation had been met.

Case D was followed for Isolation Test 7.

Durability Tests

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 during the Measurement Interval containing TPC-C database tables or recovery log data.*
- *Instantaneous interruption (system or subsystem crash/system hang) in processing which causes all or part of the processing of atomic transactions to halt.*
- *Failure of all or part of memory(loss of contents)*

Instantaneous Interruption, Loss of Memory and Loss of Log

Because the loss of power erases the contents of memory, both of instantaneous interruption and loss of memory were combined into a single test. And Loss of log also was combined to the test.

The following steps were performed on the full scaled database under the full load of users.

1. A sum of D_NEXT_O_ID of all rows in the district table was taken.
2. Full load of users were logged in to the database and kept running transactions about 5 minutes in steady state.
3. Removed one of RAID5 log disk in the steady state. The running couninued without any interruptions.
4. The system was powered off.
5. The RTE was shut down.
6. The system was powered up. IBM DB2 UDB 8.2 was restarted and automatically recovered.
7. A new count of D_NEXT_O_ID was taken.
8. This number was compared with the number of new orders reported by the RTE.

Loss of Data Disk

This test was demonstrated on a 19,800 warehouse database. The standard driving mechanism was used to generate the transaction load of 49,500 users for the test. To demonstrate recovery from a permanent failure of durable media containing TPC-C tables, the following steps were performed.

1. A sum of D_NEXT_O_ID of all rows in the district table was taken.
2. 49,500 users were logged in to the database and kept running transactions about 5 minutes in steady state.
3. One disk drive for data part in the array was removed causing IBM DB2 UDB 8.2 error. Shut down IBM DB2 UDB 8.2.
4. The 19,800 warehouse database was restored from backup and the transaction log was rolled forward.
5. A new count of D_NEXT_O_ID was taken.
6. This number was compared with the number of new orders reported by the RTE.

Loss of mirrored write-back cache

The Fibre Array system used for this benchmark has integrated feature of mirrored write-back cache. When a LUN is configured to enable write-back caching, the data on the cache is automatically mirrored on the RAMs in two controller modules, which are powered and protected from loss of power by the controller BBU. Loss of write-back cache was demonstrated on 19,800 warehouse database, by pulling off one of the controller module. A fully scaled database would also pass this test.

1. A sum of D_NEXT_O_ID of all rows in the district table was taken.
2. 198,000 users were logged in to the database and kept running transactions about 5 minutes in steady state.
3. A controller module, which manages write-back cache of mirrored drives, was pulled off.
4. Fibre system reported system error resulted in IO error for IBM DB2 UDB 8.2.
5. Contents on mirrored cache on another controller were automatically saved to temporal disk area and the Fibre Array house was eventually stopped. System was shut down.
6. The Fibre array was re-powered.
7. Saved data was automatically restored to mirrored cache and flushed to drives.
8. IBM DB2 UDB 8.2 was started up and database was recovered.
9. A new count of D_NEXT_O_ID was taken.
10. This number was compared with the number of new orders reported by the RTE.

Clause 4 : Scaling and Database Population Related Items

Initial 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 TPC-C database was originally built with 19,800 warehouses.

Table 2 Number of Rows for Server

Table	Cardinality as benchmarked
Warehouse	19,800
District	198,000
Customer	594,000,000
History	594,000,000
Orders	594,000,000
New Order	178,200,000
Order Line	5,940,241,980
Stock	1,980,000,000
Item	100,000
Deleted Warehouse Rows	0

Constant Value for the NURand function

The following values were used as constant value inputs to the NURand function for this benchmark.

C_LAST (Build)	173
C_LAST (RUN)	66

Distribution of Tables and Logs

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

Table 3 depicts the distribution of the database over the disks of the tested and priced system.

Figure 1.1, 1.2 shows the disk configuration for measured and priced system.

Table 3 : Data Distribution

HBA#	storage system#	SP#	DEU#	# of Disks	RAID Level	Capacity (GB)	Disk Partition1
Partitions for DB Log							
0	0	0	0	7	5	199.70	S: (9.77GB) for junction point
			1	15	5	465.66	Q: (465.66GB)

HBA#	storage system#	SP#	DEU#	# of Disks	RAID Level	Capacity (GB)	Disk Partition1-12	Container ID#	Partitioned Table ID#	Buffer pool Group ID#	
Partitions for DB Data								DB Mapping			
1	1	0	0	15	0	498.91	*1	1	1	1	
			1	15	0	498.91		2			
	2	0	0	15	0	498.91		3			
			1	15	0	498.91		4			
2	3	0	0	15	0	498.91		5	2		1
			1	15	0	498.91		6			
	4	0	0	15	0	498.91		7			
			1	15	0	498.91		8			
3	5	0	0	15	0	498.91		9	3		
			1	15	0	498.91		10			
	6	0	0	15	0	498.91		11			
			1	15	0	498.91		12			
4	7	0	0	15	0	498.91		13	5		2
			1	15	0	498.91		14			
	8	0	0	15	0	498.91		15			
			1	15	0	498.91		16			
5	9	0	0	15	0	498.91		17	6		
			1	15	0	498.91		18			
	10	0	0	15	0	498.91		19			
			1	15	0	498.91		20			
6	11	0	0	15	0	498.91		21	7		3
			1	15	0	498.91		22			
	12	0	0	15	0	498.91		23			
			1	15	0	498.91		24			
7	13	0	0	15	0	498.91		25	9		
			1	15	0	498.91		26			
	14	0	0	15	0	498.91		27			
			1	15	0	498.91		28			
8	15	0	0	15	0	498.91		29	10		4
			1	15	0	498.91		30			
	16	0	0	15	0	498.91		31			
			1	15	0	498.91		32			
9	17	0	0	15	0	498.91		33	12		
			1	15	0	498.91		34			
	18	0	0	15	0	498.91		35			
			1	15	0	498.91		36			

*1. Composition from Disk Partition1 to Disk Partition12 is shown in the following table.
Partition1-12

Disk Partiton#	Container name
Partition1	w are (2.01GB)
Partition2	dist (2.01GB)
Partition3	item (2.01GB)
Partition4	stock (24.01GB)
Partition5	cust (14.00GB)
Partition6	cust_idx (2.01GB)
Partition7	norderA (4.01GB)
Partition8	order (2.01GB)
Partition9	order_idx (2.01GB)
Partition10	orderl (28.01GB)
Partition11	history (20.00GB)
Partition12	norderB (4.01GB)

Type of Database

A statement must be provided that describes:

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

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

Database Mapping

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

The Warehouse, District, Customer, Order, Order-Line, New Order, History and Stock tables were horizontally partitioned into multiple tables. The specifics of the distribution of partitioned tables across the physical media can be found in Table 3.

60-Days Space

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).

To calculate the space required to sustain the database log for 8 hours of growth at steady state, the following steps were followed:

- Before the run IBM DB2 monitoring counters were reset.
- After the run "Log pages written" IBM DB2 counter was recorded.
- The space used was divided by the number of NEW-ORDERS giving a space used per NEW-ORDER transaction.
- The space used per transaction was multiplied by the measured tpmC rate times 480 minutes.

The details of both the 8-hour transaction log space requirement and the 60-day space requirement is shown in Appendix D.

Clause 5 : Performance Metrics and Response Time Related Items

Throughput

Measured tpmC must be reported

Table 4 : Measured tpmC

247,650 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.

Table 5 : Response Times (in seconds)

Type	Average	Maximum	90 th %
New-Order	0.35	29.44	0.94
Payment	0.34	21.94	0.93
Interactive Delivery	0.12	3.57	0.11
Stock Level	0.33	17.64	0.91
Order Status	0.35	16.07	0.93
Deferred Delivery	0.15	12.66	0.19
Menu	0.12	73.29	0.11

Keying and Think Times

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

Table 6 : Keying Times

Type	Average	Minimum	Maximum
New-Order	18.01	18.01	18.01
Payment	3.01	3.01	3.01
Interactive Delivery	2.01	2.01	2.01
Stock Level	2.01	2.01	2.01
Order Status	2.01	2.01	2.01

Table 7 : Think Times

Type	Average	Minimum	Maximum
New-Order	12.07	0.00	120.7
Payment	12.07	0.00	120.7
Interactive Delivery	5.07	0.00	50.7
Stock Level	5.08	0.00	50.7
Order Status	10.09	0.00	100.7

Response Time Frequency Distribution Curves

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

Figure 2.1 : New-Order Response Time Distribution

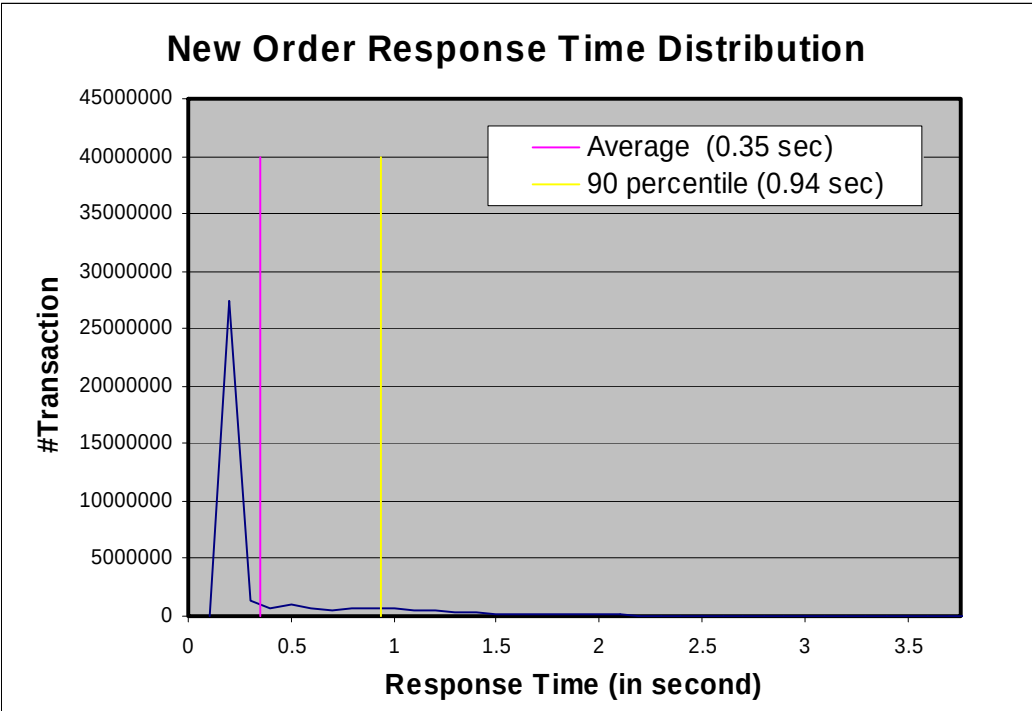


Figure 2.2 : Payment Response Time Distribution

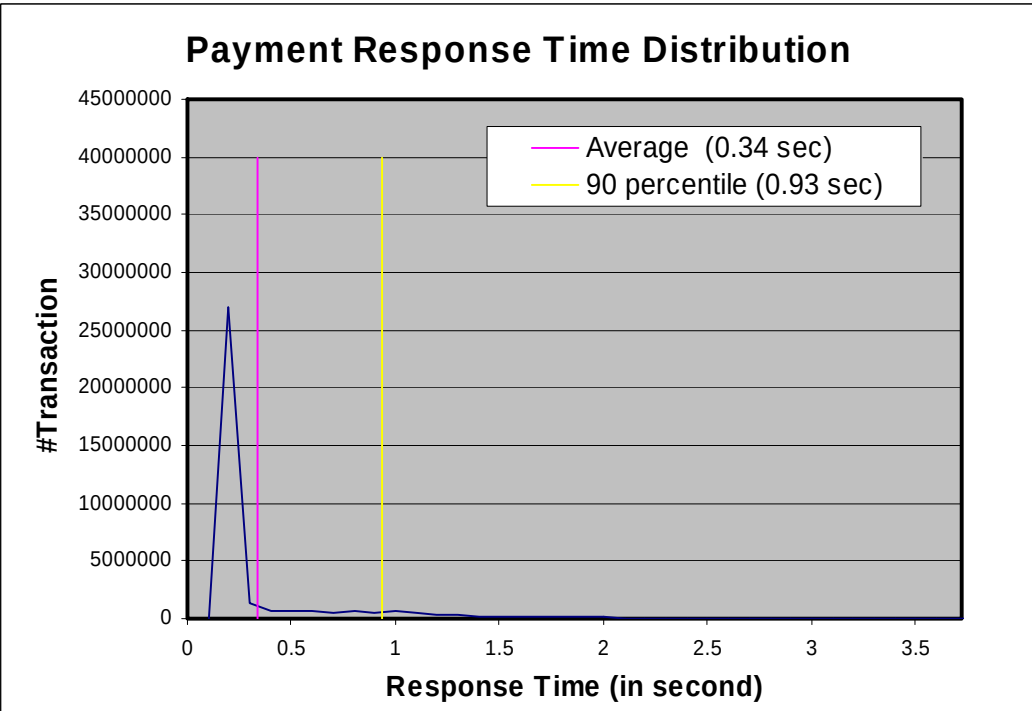


Figure 2.3 : Order-Status Response Time Distribution

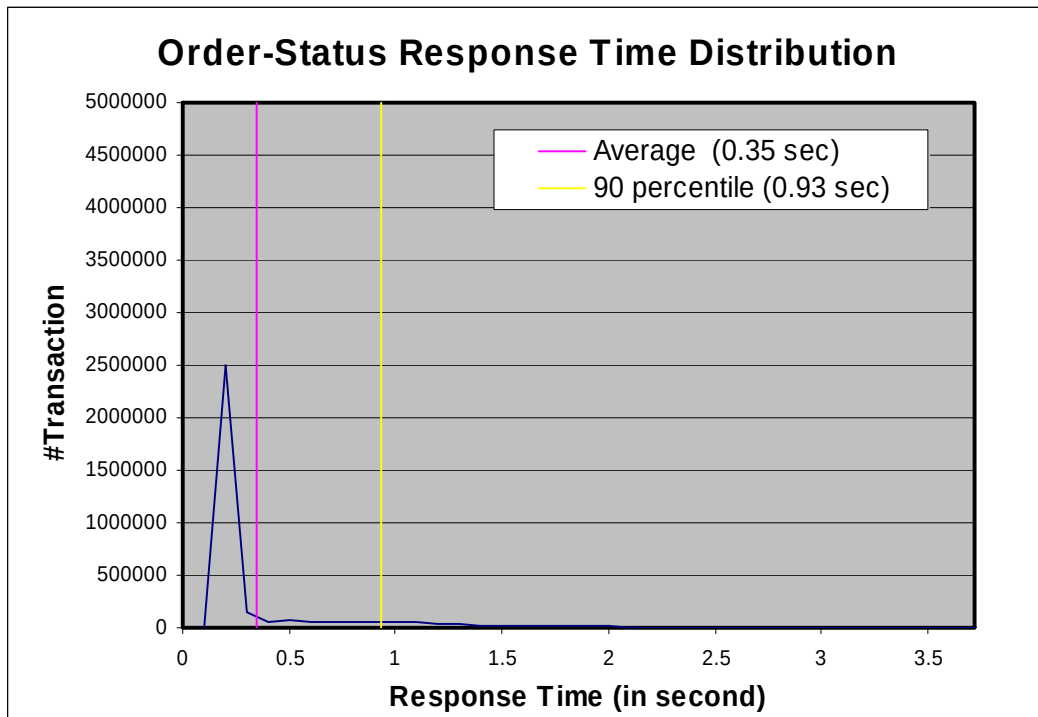


Figure 2.4 : Delivery Response Time Distribution

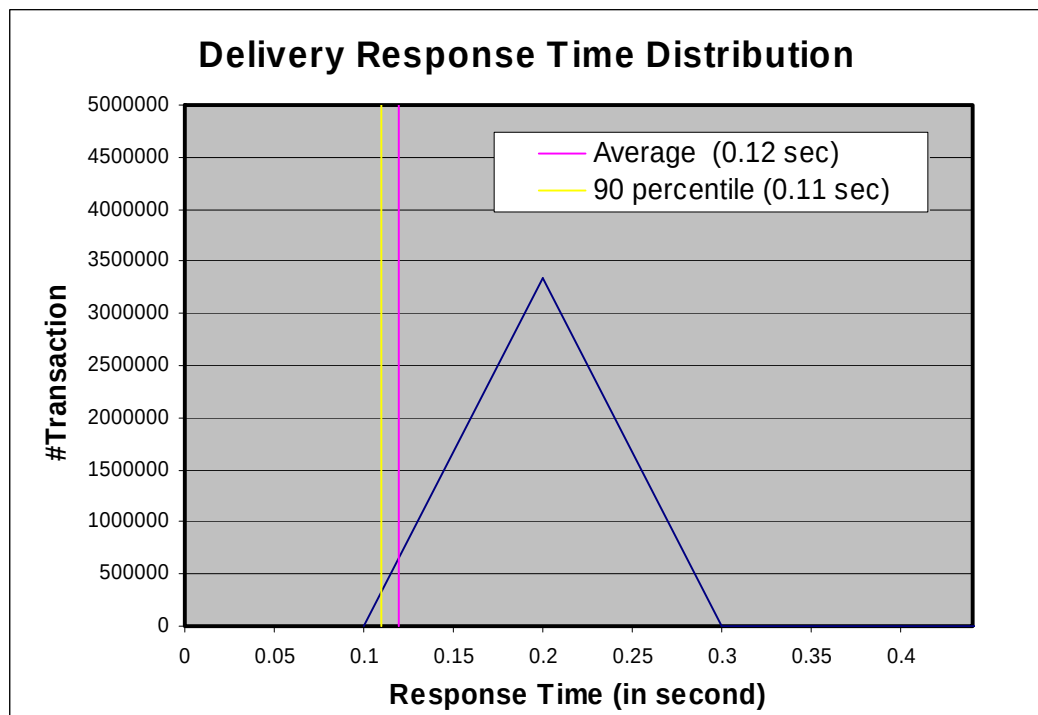
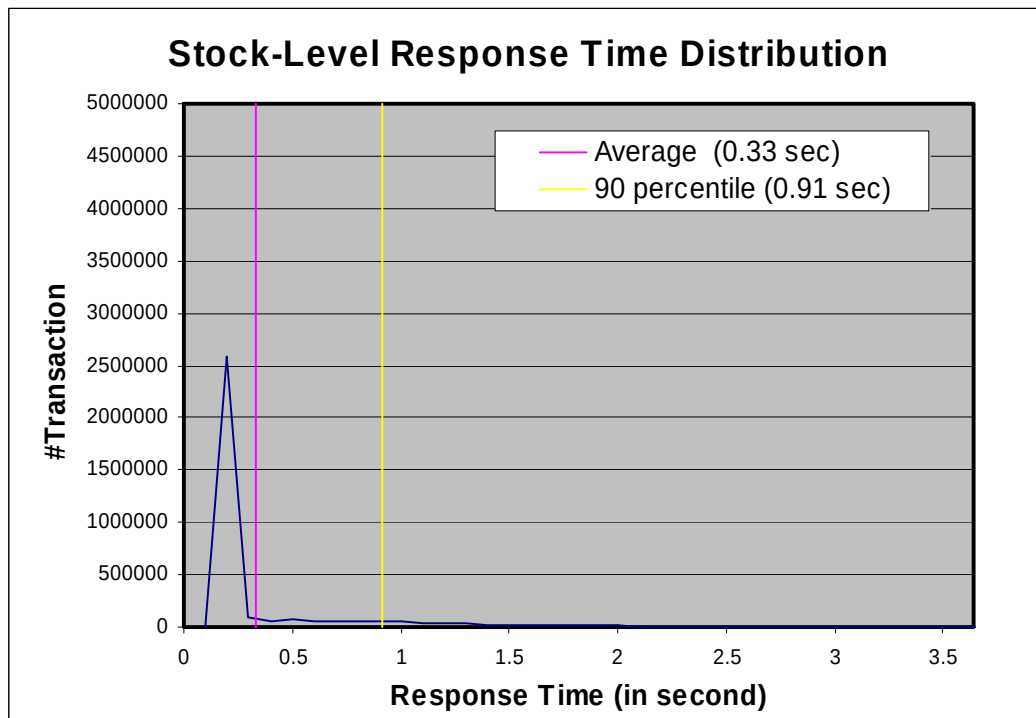


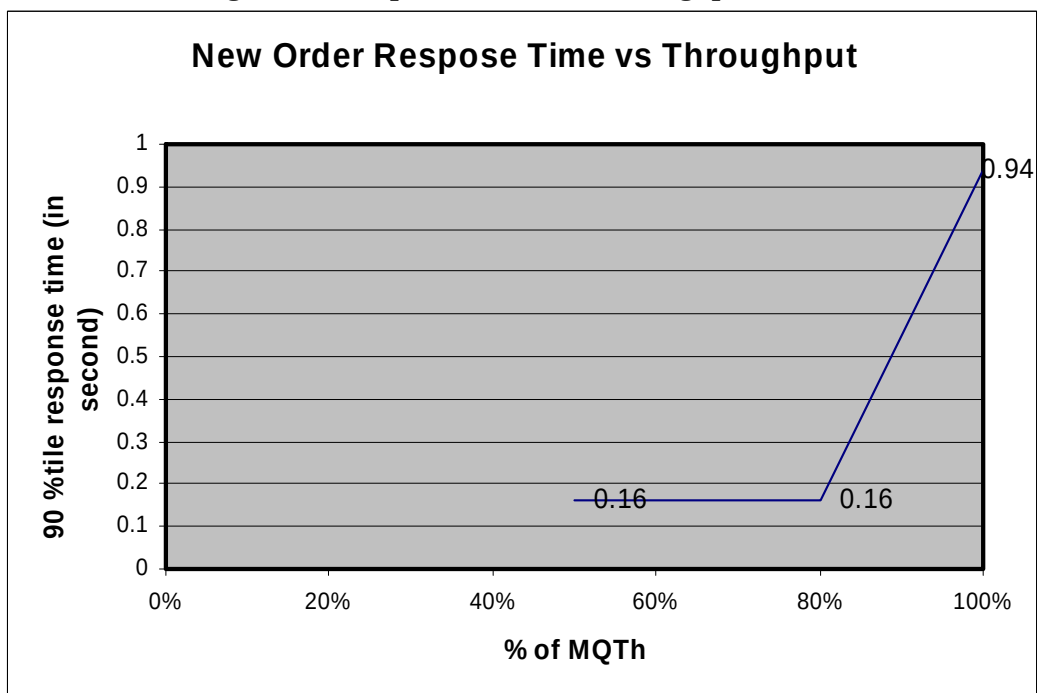
Figure 2.5 : Stock-Level Response Time Distribution



Response time versus Throughput Curve

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

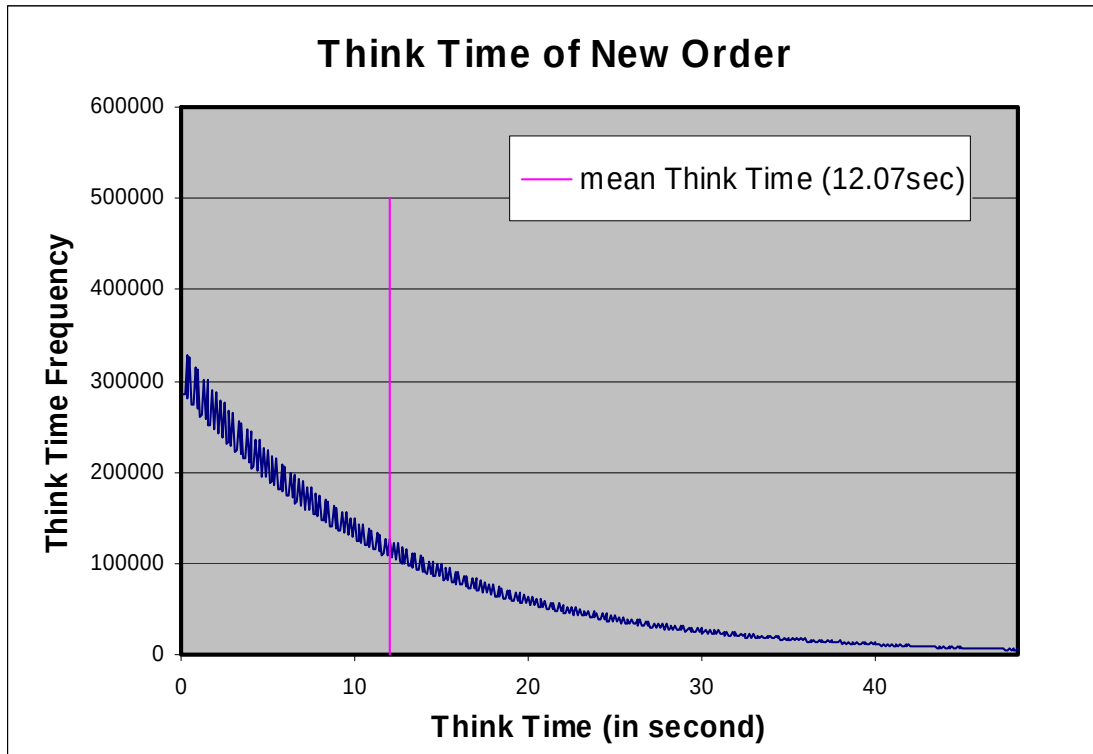
Figure 2.6 Response Time vs. Throughput Curve



New-Order Think Time Frequency Distribution

Think Time frequency distribution curves (see Clause 5.6.3) must be reported for the New-Order transaction.

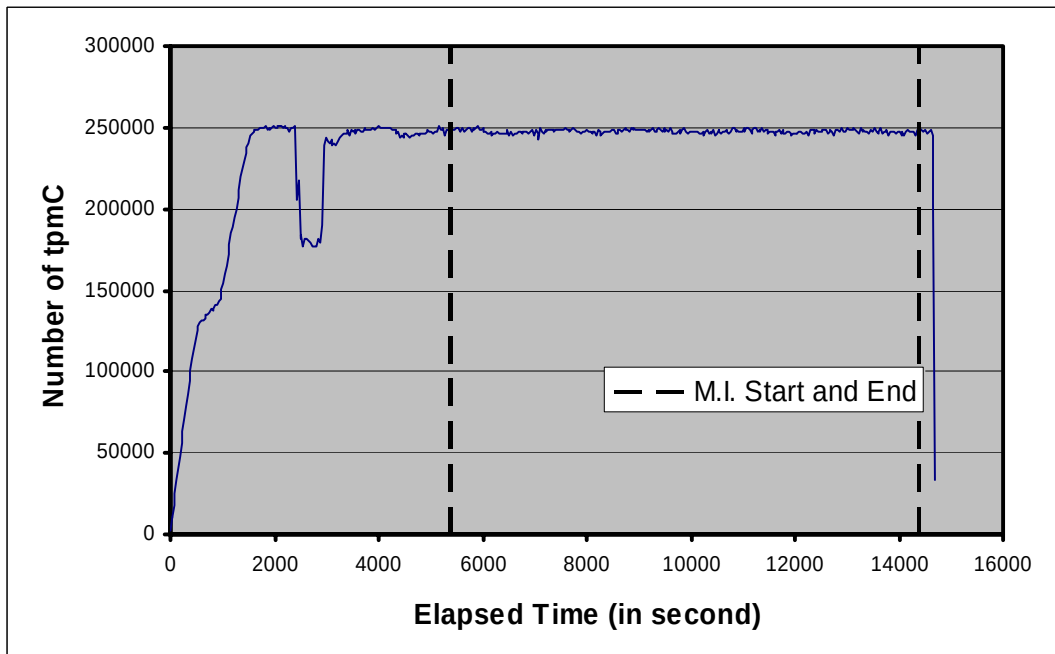
Figure 2.7 New-Order Think Time



New-Order Throughput vs. Elapsed Time

A graph of throughput versus elapsed time (see Clause 5.6.4) must be reported for the New-Order transaction.

Figure 2.8 New Order Throughput vs. Elapsed Time



Steady State

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.

Steady state was confirmed by the throughput data collected during the run and graphed in Figure 2.8.

Work Performed During Steady State

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

A 2-hour 30-minute measurement interval was used to guaranty that all work normally performed during an 8-hour sustained test are included in the reported throughput.

Regulation of 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.

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

Transaction Statistics

- . *The percentage of the total mix for each transaction type must be disclosed.*
- . *The percentage of New-Order transactions rolled back as a result of invalid item number must be disclosed.*
- . *The average number of order-lines entered per New-Order transaction must be disclosed.*
- . *The percentage of remote order-lines entered per New-Order transaction must be disclosed.*
- . *The percentage of remote Payment transactions must be disclosed.*
- . *The percentage of customer selections by customer last name in the Payment and Order-Status transactions must be disclosed.*
- . *The percentage of Delivery transactions skipped due to there being fewer than necessary orders in the New-Order table must be disclosed.*

The above statistics are disclosed in Table 1.

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.

IBM 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 IBM 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](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

Descriptions of RTE

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

The RTE used was NEC proprieta1ly. The RTE input parameters are listed in Appendix C.

Loss of Terminal Connections

The number of terminal connections lost during the Measurement Interval must be disclosed (see Clause 6.6.2).

No terminal connections were lost.

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.

As configured for this test, the driver software emulates the traffic that would be observed from the users' PCs connected by Ethernet to the front-end clients using HTTP (HyperText Transfer Protocol) over TCP/IP. One tenth of a second (100 milli seconds) was added to each transaction time to compensate for the overhead of the Web browser.

Functional Diagrams and Detail of Driver System

A complete functional diagram of both the benchmark 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 diagrams in figure 1.1 and 1.2 show the tested and priced benchmark configurations.

Network configurations and Driver system

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).

Figure 1.1 and 1.2 in this report has the network configurations of both the tested system and the priced system.

Network Bandwidth

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

The database server contains a Gigabit Ethernet adapter. This adapter is connected to a Gigabit Ethernet switch. This Gigabit Ethernet switch is cascaded to two Ethernet switches with 100Mbps bandwidth. 16 front-end clients are connected to these cascaded switches with 100Mbps bandwidth. Each client has two Ethernet adapters to be connected to RTE system. The network bandwidth between RTE system and the front-end clients is 100Mbps. 48 segments are used for the connection of this tested configuration.

Operator Intervention

If the configuration requires operator intervention (see Clause 6.6.6), the mechanism and the frequency of this intervention must be disclosed.¹¹¹

This configuration does not require any operator intervention to sustain eight hours of the reported throughput.

Clause 7 : Pricing Related Items

Hardware and Software Components

A detailed list of hardware and software used in the priced system must be reported. Each separately orderable item must have 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) of price(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 justification of any discounts applied must be disclosed in the price sheet. Sufficient detail of what items are being discounted and by how much they are being discounted must be provided so that the discount amount used in the computation of the total system cost can be independently reproduced.

The detailed list of all hardware and software for the priced configuration is listed in the system pricing summary.

Availability

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 date for the priced system must be the date at which all components are committed to be available. This single date must be reported on the first page of the Executive Summary. All availability dates, whether for individual components or for the SUT as a whole, must be disclosed to a precision of one day.

The total system as priced will be available July 24, 2005.

- “IBM DB2 Universal Database 8.2” will be available by July 24, 2005.
- The other products are already available.

Throughput, and Price Performance

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

- Maximum Qualified Throughput : 247,650 tpmC
- Price per tpmC : \$5.48 per tpmC
- Total 3-year cost of ownership : \$1,356,338 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.

This system 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:

- 8 IBM DB2 UDB 8.2 Enterprise Server Edition Proc Lic/1 yr Maint
- 1 Windows® Server 2003, Datacenter Edition
- 17 Windows® 2000 Server, Server License
- 1 Visual C++ Standard Edition

System Pricing

System pricing should include subtotals for the following components: Server Hardware, Server Software, Client Hardware, Client Software, and Network Components. Clause 6.1 describes the Server and Client components.

System pricing must include line item indication where non-sponsoring companies' part numbers 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 8 : Audit Related Items

Auditor's Report

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.

Next page contains the complete independent auditor's report by Francois Raab of Info Sizing Inc. for the test described in this report.

Availability of the Full Disclosure Report

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

Requests for this TPC Benchmark™ C Full Disclosure Report should be sent to:

Transaction Processing Performance Council

Presidio of San Francisco

Building 572B (surface)

P.O. Box 29920 (mail) San Francisco, CA 94129-0920

Voice: 415-561-6272

Fax: 415-561-6120

Email: info@tpc.org

Auditor's letter



Sponsor: Masami Shiraishi
NEC Corporation
1-10 Nisshincho
Fuchu-City, Tokyo 183-8501, Japan

January 6, 2005

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

Platform: NEC Express5800/1160Xe c/s
Operating system: Microsoft Windows Server 2003, Datacenter Edition for 64-bit
Itanium-based Systems
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: NEC Express5800/1160Xe				
8 x Itanium2 (1.6 GHz)	256 GB (9 MB Cache/cpu)	1 x 36 GB int. 562 x 36 GB ext.	0.94 Seconds	247,650.67
Sixteen (16) Clients: NEC Express 5800/120Rf-2 (Specification for each)				
2 x Xeon (2.0 GHz)	2 GB (512 KB cache/cpu)	1 x 36 GB	n/a	n/a

In my opinion, these performance results were produced in compliance with the TPC requirements for version 5.3 of the benchmark.

1373 North Franklin Street • Colorado Springs, CO 80903-2527 • Office: 719/473-7555 • Fax: 719/473-7554

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 2.5 hours
- 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', with a stylized flourish at the end.

François Raab, President

1373 North Franklin Street • Colorado Springs, CO 80903-2527 • **Office:** 719/473-7555 • **Fax:** 719/473-7554

Appendix A : Application Source Code

A1. Front-End Source Code, Scripts

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 -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=$(LD_FLAGS_SHLIB) /DEF:rpctpcc.def
LD_FLAGS_LIB=/LIBPATH:$(TPCC_SQLLIB)\lib /LIBPATH:"C:\Program
Files\Microsoft Visual Studio\VC98\Lib" db2api.lib winmm.lib
LD_FLAGS_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=CK040607

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

@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=%HOME%\tpc-c.ibm
set TPCC_SQLLIB=%HOME%\SQLLIB
set TPCC_RUNDATA=%HOME%\tpc-c.ibm\RUNS

@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

```

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      /* May be ignored by RTE */
#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);

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

```

lval.h

```

#ifdef __LVAL_H
#define __LVAL_H
#define WAREHOUSES 19800
#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

```

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

```

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 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

```

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.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) tpcccli.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 -1 -n TPCCCLI -g -# 0 -o TPCCCLI.exfmt.plan
- db2expln -d $(TPCC_DBNAME) -c $(TPCC_SCHEMA) -p TPCCCLI -s 0 -g
-o TPCCCLI.expln.plan

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

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

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

tpcccli.c:
@echo "Prepping $.sqc"
-db2 prep $.sqc $(PRP_OPTS) ISOLATION RR
@echo "Binding $.bnd"
db2 bind $.bnd $(BND_OPTS) QUERYOPT 7

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

# Client Library:
tpcccli$(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

```

tpcccli.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.
*****/

```

```

*****/

/*
 * 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 "lval.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_0_ENTRY_D_time = time(NULL) ;

// Sort the item list ; This helps eliminate duplicates anyway, and
since invalid item
// IDs == 100001 , we will remain compliant with 2.4.2.3 Comment 1.
// !! So DON'T sort or change the order of the items in any
subsequent code, including SQL.

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 ;
    }
}

// Pre-duplicate processing count

in_neword->s_0_OL_CNT = itemIndex ;

// Sort the original array

qsort( in_neword->in_item, in_neword->s_0_OL_CNT
      , sizeof ( in_neword->in_item[ 0 ] )
      , itemComparison
    ) ;

// Now purge the duplicates.

actualItemIndex = -1 ;

for ( itemIndex = 0
      ; itemIndex < in_neword->s_0_OL_CNT
      ; itemIndex++ )
{
    // If duplicate, just increment the item entry order count

    if ( actualItemIndex >= 0
        && 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 ;
    }
    else
    {
        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_0_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_0_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_0_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_0_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_0_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_0_OL_CNT);
    SWAP_BYTE(in_neword->s_all_local);
    SWAP_BYTE(in_neword->duplicate_items);
    for (itemIndex=0; itemIndex<in_neword->s_0_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 deferered 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

if ( in_c_id == 0 )
{
    // Pre-built c_data prefix for BC does not include c_id in this
    // instance

    // Tne strange $04.4d.%02.2d printf modifier is to print a (4,2)
    // 0-padded floating-point value -- %f won't 0-pad by default.
    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) ) ;

    // 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 ;

```

```

        // Pre-built c_data prefix for BC does include c_id in this
instance
        // The strange $04.4d.%02.2d printf modifier is to print a (4,2)
        // 0-padded floating-point value -- %f won't 0-pad by default.
        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_id , c_d_id , c_w_id ,
d_id , w_id , (int)(h_amount / 100), (int)(h_amount % 100) ) ;

    }

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_s
treet_2 , :c_city , :c_state
         , :c_zip , :c_phone , :c_since , :c_credit , :c_credi
t_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_st
reet_2 , :c_city , :c_state
         , :c_zip , :c_phone , :c_since , :c_credit , :c_credi
t_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 )
    {
        // Propagate 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    w_id ;
        ///short        d_id ;
        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

    SELECT ITEMS_BELOW_THRESHOLD

        INTO :low_stock

        FROM TABLE( STOCK_LEVEL( :w_id, :d_id, :threshold ) ) AS T

        WITH CS;

*/

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 ) ;
}

```

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 tpccctx.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

```

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 - 2005
** 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; }

/* Copy 9 characters - 8 for dbname, 1 for NULL */
strncpy(dbname,in_dbname,9);
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;

    sqleSetTypeCtx(SQL_CTX_MULTI_MANUAL);

```

```

    sqleBeginCtx(&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;

    sqleAttachToCtx(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;

    sqleDetachFromCtx(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; }

    sqleEndCtx(&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;

```

```

    sqlcGetCurrentCtx(ctx, NULL, &sqlca);

    if (SQLCODE != 0) {
        sqlerror( CLIENT_SQL, "GETCTX", __FILE__, __LINE__, &sqlca);
        return SQLCODE;
    }

    return 0;
}

```

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 - 2005
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****/

/*
 * tccdbg.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_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);

if ((err_fp = fopen(err_fn, "a+")) == NULL)
{
    return;
}

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_tmstp(&timeStamp[0]);
    timeStamp[19] = (char)NULL;

    if ((debug_fp = fopen(filename, "a+")) == NULL)
    {
        return;
    }
}

```



```

        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 (%llx)\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\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);
}

```

error.h

```

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

```



```

#pragma once

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

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

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

#define ERR_FATAL_LEVEL          1
#define ERR_WARNING_LEVEL       2
#define ERR_INFORMATION_LEVEL   3

#define ERR_TYPE_LOGIC           -1
    //logic error in program; internal error
#define ERR_SUCCESS              0
    //success (a non-error error)
#define ERR_BAD_ITEM_ID         1
    //expected abort record in txnRecord
#define ERR_TYPE_DELIVERY_POST   2
    //expected delivery post failed
#define ERR_TYPE_WEBDLL          3
    //tpcc web generated error
#define ERR_TYPE_SQL             4
    //sql server generated error
#define ERR_TYPE_DBLIB           5
    //dblib generated error
#define ERR_TYPE_ODBC            6
    //odbc generated error
#define ERR_TYPE_SOCKET          7
    //error on communication socket client rte only
#define ERR_TYPE_DEADLOCK        8
    //dblib and odbc only deadlock condition
#define ERR_TYPE_COM             9
    //error from COM call
#define ERR_TYPE_TUXEDO          10
    //tuxedo error
#define ERR_TYPE_OS              11
    //operating system error
#define ERR_TYPE_MEMORY          12
    //memory allocation error
#define ERR_TYPE_TPCC_ODBC       13
    //error from tpcc odbc txn module
#define ERR_TYPE_TPCC_DBLIB      14
    //error from tpcc dblib txn module
#define ERR_TYPE_DELISRV         15
    //delivery server error
#define ERR_TYPE_TXNLOG          16
    //txn log error
#define ERR_TYPE_BCCONN          17
    //Benchcraft connection class

```

```

#define ERR_TYPE_TPCC_CONN      18
    //Benchcraft connection class
#define ERR_TYPE_ENCINA         19
    //Encina error
#define ERR_TYPE_COMPONENT      20
    //error from COM component
#define ERR_TYPE_RTE            21
    //Benchcraft rte
#define ERR_TYPE_AUTOMATION     22
    //Benchcraft automation errors

#define WEB_ERR(rc)             c=wsprintf(szForm,"<HTML><HEAD><TITLE>TPC-C
Error %d</TITLE></HEAD><BODY>" \
    "<BOLD>An Error Occurred</BOLD><BR><BR>",rc);

class CBaseErr
{
public:
    char    *m_szApp;
    char    *m_szMsg;
    char    *m_szLoc;    // code location where the error occurred
    int      m_idMsg;

    CBaseErr(void)
    {
        m_idMsg          = 0;
        m_szMsg           = new char[m_szMsg_size];
        m_szApp           = new char[m_szApp_size];
        m_szLoc           = NULL;

        m_szMsg[0]        = 0;
        m_szApp[0]         = 0;

        GetModuleFileName(GetModuleHandle(NULL), m_szApp,
m_szApp_size);
    }

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

    CBaseErr(int idMsg)
    {
        m_idMsg           = idMsg;
        m_szApp           = new char[m_szApp_size];
        m_szMsg           = new char[m_szMsg_size];
        m_szLoc           = NULL;

        GetModuleFileName(GetModuleHandle(NULL), m_szApp,
m_szApp_size);
        LoadString(GetModuleHandle(NULL), idMsg, m_szMsg,
m_szMsg_size);
    }

    CBaseErr(LPCTSTR szMsg)
    {
        m_idMsg           = 0;

```

```

        m_szApp      = new char[m_szApp_size];
        m_szMsg      = new char[m_szMsg_size];
        m_szLoc      = NULL;

        GetModuleFileName(GetModuleHandle(NULL), m_szApp,
m_szApp_size);
        strcpy(m_szMsg, szMsg);
    }

    void SetError(char *szMsg, LPCTSTR szLocation)
    {
        if (szMsg != NULL)
            strcpy(m_szMsg, szMsg);
        else
            m_szMsg[0] = 0;

        if (szLocation != NULL)
        {
            delete [] m_szLoc;
            m_szLoc = new char[strlen(szLocation)+1];
            strcpy(m_szLoc, szLocation);
        }
        else
        {
            delete [] m_szLoc;
            m_szLoc = NULL;
        }
    }

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

        if (szStr)
            j = wsprintf(szTmp, "%s\n", szStr);
        if (m_szLoc)
            j += wsprintf(szTmp+j, "Location=%s\n", m_szLoc);
        if (m_szMsg)
            j += wsprintf(szTmp+j, "%s\n", m_szMsg);

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

    char *GetApp(void) { return m_szApp; }
    char *GetMsg(void) { return m_szMsg; }
    char *GetLocation(void) { return m_szLoc; }

    virtual int ErrorType() = 0; // a value which distinguishes the
kind of error that occurred
    virtual int ErrorNum() = 0; // an error value specific
to the error type
    virtual char *ErrorText() = 0; // a string (i.e., human
readable) representation of the error
};

class CSocketErr : public CBaseErr
{
public:
    enum Action
    {
        eNone,

```

```

        eSend,
        eSocket,
        eConnect
    };

    CSocketErr(Action eAction, LPCTSTR szLocation);
    CSocketErr(int iError) { m_errId = iError; };
    int      m_errId;
    Action   m_eAction;

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

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

    CSystemErr(Action eAction, LPCTSTR szLocation);

    void Draw(HWND hwnd, LPCTSTR szStr = NULL);

    int      m_errId;
    Action   m_eAction;

    int ErrorType() { return ERR_TYPE_OS; }
    int ErrorNum() { return m_errId; }
    char *ErrorText() { return m_szMsg; }
};

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

    int ErrorType() { return ERR_TYPE_MEMORY; }
    int ErrorNum() { return 0; }
    char *ErrorText() { return "Insufficient Memory to continue.";}
};

```

```
};
```

ReadRegistry.cpp

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

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

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

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

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

```
    else if ( !strcmp(szTmp, szTxnMonNames[COM]) )
        pReg->eTxnMon = COM;
    }

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

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

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

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

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

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

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

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

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

    RegCloseKey(hKey);

    return FALSE;
}
```

ReadRegistry.h

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

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

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

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

BOOL ReadTPCCRegistrySettings( TPCCREGISTRYDATA *pReg );
```

trans.h

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

#include "db2tpcc.h"

// String length constants
```

```
#define SERVER_NAME_LEN 20
#define DATABASE_NAME_LEN 20
#define USER_NAME_LEN 20
#define PASSWORD_LEN 20
#define TABLE_NAME_LEN 20
#define I_DATA_LEN 50
#define I_NAME_LEN 24
#define BRAND_LEN 1
#define LAST_NAME_LEN 16
#define W_NAME_LEN 10
#define ADDRESS_LEN 20
#define STATE_LEN 2
#define ZIP_LEN 9
#define S_DIST_LEN 24
#define S_DATA_LEN 50
#define D_NAME_LEN 10
#define FIRST_NAME_LEN 16
#define MIDDLE_NAME_LEN 2
#define PHONE_LEN 16
#define DATETIME_LEN 30
#define CREDIT_LEN 2
#define C_DATA_LEN 250
#define H_DATA_LEN 24
#define DIST_INFO_LEN 24
#define MAX_OL_NEW_ORDER_ITEMS 15
#define MAX_OL_ORDER_STATUS_ITEMS 15
#define STATUS_LEN 25
#define OL_DIST_INFO_LEN 24

// TIMESTAMP_STRUCT is provided by the ODBC header file sqltypes.h, but
is not available
// when compiling with dblib, so redefined here. Note: we are using the
symbol "__SQLTYPES"
// (declared in sqltypes.h) as a way to determine if TIMESTAMP_STRUCT
has been declared.
#ifndef __SQLTYPES
typedef struct
{
    short /* SQLSMALLINT */ year;
    unsigned short /* SQLUSMALLINT */ month;
    unsigned short /* SQLUSMALLINT */ day;
    unsigned short /* SQLUSMALLINT */ hour;
    unsigned short /* SQLUSMALLINT */ minute;
    unsigned short /* SQLUSMALLINT */ second;
    unsigned long /* SQLINTEGER */ fraction;
} TIMESTAMP_STRUCT;
#endif

// possible values for exec_status_code after transaction completes
enum EXEC_STATUS
{
    eSqlError = -1, // -1 "Sql error."
    eOK = 0, // 0 "Transaction committed."
    eDeliveryFailed = 1, // 1 "Delivery Post Failed."
    eInvalidItem = 100 // 100 "Item number is not valid."
};

#define ASSIGN_STATUS(es, i) switch (i) {
    \
    case -1:
    case 0: es=eOK;
    \
    es=eSqlError;break;
    \
    break;
```

```

case 1:
case 100:

es=eDeliveryFailed; break; \
es=eInvalidItem; break; }

// transaction structures
typedef struct
{
    // input params
    short          ol_supply_w_id;
    long           ol_i_id;
    short          ol_quantity;

    // output params
    char           ol_i_name[I_NAME_LEN+1];
    char           ol_brand_generic[BRAND_LEN+1];
    double         ol_i_price;
    double         ol_amount;
    short          ol_stock;
} OL_NEW_ORDER_DATA;

typedef struct
{
    // input params
    short          w_id;
    short          d_id;
    long           c_id;
    short          o_ol_cnt;

    // output params
    EXEC_STATUS    exec_status_code;
    char           c_last[LAST_NAME_LEN+1];
    char           c_credit[CREDIT_LEN+1];
    double         c_discount;
    double         w_tax;
    double         d_tax;
    long           o_id;
    short          o_commit_flag;
    TIMESTAMP_STRUCT o_entry_d;
    short          o_all_local;
    double         total_amount;
    OL_NEW_ORDER_DATA OL[MAX_OL_NEW_ORDER_ITEMS];
} NEW_ORDER_DATA, *PNEW_ORDER_DATA;

typedef struct
{
    // input params
    short          w_id;
    short          d_id;
    long           c_id;
    short          c_d_id;
    short          c_w_id;
    double         h_amount;
    char           c_last[LAST_NAME_LEN+1];

    // output params
    EXEC_STATUS    exec_status_code;
    TIMESTAMP_STRUCT h_date;
    char           w_street_1[ADDRESS_LEN+1];
    char           w_street_2[ADDRESS_LEN+1];
    char           w_city[ADDRESS_LEN+1];
    char           w_state[STATE_LEN+1];
    char           w_zip[ZIP_LEN+1];
    char           d_street_1[ADDRESS_LEN+1];

```

```

char           d_street_2[ADDRESS_LEN+1];
char           d_city[ADDRESS_LEN+1];
char           d_state[STATE_LEN+1];
char           d_zip[ZIP_LEN+1];
char           c_first[FIRST_NAME_LEN+1];
char           c_middle[MIDDLE_NAME_LEN + 1];
char           c_street_1[ADDRESS_LEN+1];
char           c_street_2[ADDRESS_LEN+1];
char           c_city[ADDRESS_LEN+1];
char           c_state[STATE_LEN+1];
char           c_zip[ZIP_LEN+1];
char           c_phone[PHONE_LEN+1];
TIMESTAMP_STRUCT c_since;
char           c_credit[CREDIT_LEN+1];
double         c_credit_lim;
double         c_discount;
double         c_balance;
char           c_data[200+1];
} PAYMENT_DATA, *PPAYMENT_DATA;

typedef struct
{
    long           ol_i_id;
    short          ol_supply_w_id;
    short          ol_quantity;
    double         ol_amount;
    TIMESTAMP_STRUCT ol_delivery_d;
} OL_ORDER_STATUS_DATA;

typedef struct
{
    // input params
    short          w_id;
    short          d_id;
    long           c_id;
    char           c_last[LAST_NAME_LEN+1];

    // output params
    EXEC_STATUS    exec_status_code;
    char           c_first[FIRST_NAME_LEN+1];
    char           c_middle[MIDDLE_NAME_LEN+1];
    double         c_balance;
    long           o_id;
    TIMESTAMP_STRUCT o_entry_d;
    short          o_carrier_id;
    OL_ORDER_STATUS_DATA OL[MAX_OL_ORDER_STATUS_ITEMS];
    short          o_ol_cnt;
} ORDER_STATUS_DATA, *PORDER_STATUS_DATA;

typedef struct
{
    // input params
    short          w_id;
    short          o_carrier_id;

    // output params
    EXEC_STATUS    exec_status_code;
    SYSTEMTIME     queue_time;
    long           o_id[10];
} DELIVERY_DATA, *PDELIVERY_DATA;

//This structure is used for posting delivery transactions and for
writing them to the delivery server.

```

```

typedef struct _DELIVERY_TRANSACTION
{
    __int64        queue;           //time delivery transaction
    short          w_id;            //delivery warehouse
    short          o_carrier_id;    //carrier id
} DELIVERY_TRANSACTION;

typedef struct
{
    // input params
    short          w_id;
    short          d_id;
    short          threshold;

    // output params
    EXEC_STATUS    exec_status_code;
    long           low_stock;
} STOCK_LEVEL_DATA, *PSTOCK_LEVEL_DATA;

```

txn_base.h

```

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

#pragma once

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

class DllDecl CTPCC_BASE
{
public:
    CTPCC_BASE(void) {};
    virtual ~CTPCC_BASE(void) {};

    virtual PNEW_ORDER_DATA      BuffAddr_NewOrder()
    = 0;
    virtual PPAYMENT_DATA        BuffAddr_Payment()
    = 0;
    virtual PDELIVERY_DATA        BuffAddr_Delivery()
    = 0;
    virtual PSTOCK_LEVEL_DATA      BuffAddr_StockLevel() = 0;
    virtual PORDER_STATUS_DATA    BuffAddr_OrderStatus() = 0;

    virtual void NewOrder        () = 0;
    virtual void Payment         () = 0;
    virtual void Delivery        () = 0;
    virtual void StockLevel      () = 0;

```

```

};
virtual void OrderStatus        () = 0;

```

tpcc_dblib.cpp

```

#include <windows.h>
#include <stdio.h>
#include <time.h>
#define DBNTWIN32

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

#include "..\..\common\src\error.h"
#include "..\..\common\src\trans.h"
#include "..\..\common\src\txn_base.h"
#include "tpcc_dblib.h"

static CRITICAL_SECTION          DBLIBCriticalSection;

BOOL WINAPI DllMain(HMODULE hModule, DWORD ul_reason_for_call, LPVOID
lpReserved)
{
    // DebugBreak();
    switch( ul_reason_for_call )
    {
    case DLL_PROCESS_ATTACH:
        DisableThreadLibraryCalls(hModule);
        InitializeCriticalSection(&DBLIBCriticalSection);
        break;
    case DLL_PROCESS_DETACH:
        DeleteCriticalSection(&DBLIBCriticalSection);
        break;
    default:
        /* nothing */;
    }
    return TRUE;
}

// wrapper routine for class constructor
__declspec(dllexport) CTPCC_DBLIB* CTPCC_DBLIB_new(
    LPCSTR szUser,
    LPCSTR szPassword,
    LPCSTR szDatabase )
{
    // user name for login
    // password for login
    // name of database to use
    {
        return new CTPCC_DBLIB( szUser, szPassword, szDatabase );
    }
}

CTPCC_DBLIB::CTPCC_DBLIB (
    LPCSTR szUser,
    LPCSTR szPassword,    // password for
    LPCSTR szDatabase )    // name of database
{
    to use
    {
        EnterCriticalSection(&DBLIBCriticalSection);
    }
}

```

```

    int rc = connect_to_TM_auth((char *) szDatabase, (char *)
szUser, (char *) szPassword);
    if(rc != 0)
    {
        /* throw error */
        throw CDBLIBERR( "DBLIB: CONNECT TO DB2" );
    }

    if ( (rc = get_context(&connection_ctx)) != 0)
    {
        /*throw error */
        throw CDBLIBERR( "DBLIB: GET CTX" );
    }

    if ( (rc = detach_context(connection_ctx)) != 0)
    {
        /* throw error */
        throw CDBLIBERR( "DBLIB: CONNECT TO DB2" );
    }

    LeaveCriticalSection(&DBLIBCriticalSection);
}

CTPCC_DBLIB::~CTPCC_DBLIB( void )
{
    // close db connection and deallocate resources
    EnterCriticalSection(&DBLIBCriticalSection);
    int rc = attach_context(connection_ctx);
    if(rc != 0)
    {
        throw CDBLIBERR( "DBLIB: DESTRUCTOR ATTACH" );
    }
    disconnect_from_TM();
    LeaveCriticalSection(&DBLIBCriticalSection);
}

#define STR_CPY(target, source) memcpy(target,source, sizeof(source)); \
target[sizeof(source)-1]='\0'

#define int_to_dbl(i,d) d=(double)(i/100.0)

#define SQL_TIMESTAMP(ts,pts,dr)    dr = localtime ( ((long *)pts) ); \
ts.year    = dr->tm_year+1900;    \
ts.month   = ++(dr->tm_mon);       \
ts.day     = dr->tm_mday;          \
ts.hour    = dr->tm_hour;          \
ts.minute  = dr->tm_min;           \
ts.second  = dr->tm_sec;

/*
The mapping
year  = daterec->tm_year+1900; //years since 1900
month = ++(daterec->tm_mon);   //0-11
day   = daterec->tm_mday;      //1-31
hour  = daterec->tm_hour;      //0-23
minute = daterec->tm_min;      //0-59
second = daterec->tm_sec;      //0-59
*/

```

```

#define SET0(s) ZeroMemory(&s, sizeof(s))

void CTPCC_DBLIB::StockLevel()
{
    int rc = attach_context(connection_ctx);
    if(rc != 0)
    {
        //throw error;
        throw CDBLIBERR( "DBLIB: ATTACH STK" );
    }
    SET0(m_in_txn.stk);
    SET0(m_out_txn.stk);

    m_in_txn.stk.s_W_ID=m_txn.StockLevel.w_id;
    m_in_txn.stk.s_D_ID=m_txn.StockLevel.d_id;
    m_in_txn.stk.s_threshold=m_txn.StockLevel.threshold;

    stocklev_sql(&m_in_txn.stk, &m_out_txn.stk);

    m_txn.StockLevel.low_stock=m_out_txn.stk.s_low_stock;

    ASSIGN_STATUS(m_txn.StockLevel.exec_status_code,m_out_txn.stk.s_transtat
us);

    rc = detach_context(connection_ctx);
    if(rc != 0)
    {
        //throw error
        throw CDBLIBERR( "DBLIB: DETACH STK" );
    }
    return;
}

void CTPCC_DBLIB::NewOrder()
{
    int i;
    tm *daterec;

    int rc = attach_context(connection_ctx);
    if(rc != 0)
    {
        //throw error;
        throw CDBLIBERR( "DBLIB: ATTACH NO" );
    }

    //zero out the new ord struct, takes care of unused items
    SET0(m_in_txn.no);
    SET0(m_out_txn.no);

    m_in_txn.no.s_W_ID=m_txn.NewOrder.w_id;
    m_in_txn.no.s_D_ID=m_txn.NewOrder.d_id;
    m_in_txn.no.s_C_ID=m_txn.NewOrder.c_id;
    m_in_txn.no.s_O_OL_CNT=m_txn.NewOrder.o_ol_cnt;

    for (i = 0; i<m_in_txn.no.s_O_OL_CNT; i++)
    {
        m_in_txn.no.in_item[i].s_OL_I_ID=m_txn.NewOrder.OL[i].ol_i_id;
    }
}

```

```

m_in_txn.no.in_item[i].s_OL_SUPPLY_W_ID=m_txn.NewOrder.OL[i].ol_supply_w
_id;

m_in_txn.no.in_item[i].s_OL_QUANTITY=m_txn.NewOrder.OL[i].ol_quantity;
}

//call new order
neword_sql(&m_in_txn.no,&m_out_txn.no);

STR_CPY(m_txn.NewOrder.c_last,m_out_txn.no.s_C_LAST);
STR_CPY(m_txn.NewOrder.c_credit,m_out_txn.no.s_C_CREDIT);

int_to_dbl(m_out_txn.no.s_C_DISCOUNT,m_txn.NewOrder.c_discount);
int_to_dbl(m_out_txn.no.s_W_TAX,m_txn.NewOrder.w_tax);
int_to_dbl(m_out_txn.no.s_D_TAX,m_txn.NewOrder.d_tax);

m_txn.NewOrder.o_id=m_out_txn.no.s_O_ID;

SQL_TIMESTAMP( m_txn.NewOrder.o_entry_d,
&m_out_txn.no.s_O_ENTRY_D_time,daterec)

int_to_dbl(m_out_txn.no.s_total_amount,m_txn.NewOrder.total_amount);
m_txn.NewOrder.o_ol_cnt=m_in_txn.no.s_O_OL_CNT;
for (i = 0; i<m_txn.NewOrder.o_ol_cnt; i++)
{
    m_txn.NewOrder.OL[i].ol_i_id=m_in_txn.no.in_item[i].s_OL_I_ID;
m_txn.NewOrder.OL[i].ol_supply_w_id=m_in_txn.no.in_item[i].s_OL_SUPPLY_W
_ID;
m_txn.NewOrder.OL[i].ol_quantity=m_in_txn.no.in_item[i].s_OL_QUANTITY;
    STR_CPY(m_txn.NewOrder.OL[i].ol_i_name,
m_out_txn.no.item[i].s_I_NAME);
    STR_CPY(m_txn.NewOrder.OL[i].ol_brand_generic,
&m_out_txn.no.item[i].s_brand_generic);
int_to_dbl(m_out_txn.no.item[i].s_I_PRICE,m_txn.NewOrder.OL[i].ol_i_pric
e) ;
int_to_dbl(m_out_txn.no.item[i].s_OL_AMOUNT,m_txn.NewOrder.OL[i].ol_amou
nt) ;
    m_txn.NewOrder.OL[i].ol_stock =
m_out_txn.no.item[i].s_S_QUANTITY;
}

ASSIGN_STATUS(m_txn.NewOrder.exec_status_code,m_out_txn.no.s_transtatus)
;

rc = detach_context(connection_ctx);
if(rc != 0)
{
    //throw error
    throw CDBLIBERR( "DBLIB: DETACH NO" );
}

```

```

}
return;
}

void CTPCC_DBLIB::Payment()
{
    tm *daterec;
    int rc = attach_context(connection_ctx);
    if(rc != 0)
    {
        //throw error;
        throw CDBLIBERR( "DBLIB: ATTACH PMT" );
    }

    SET0(m_in_txn.pay);
    SET0(m_out_txn.pay);

    m_in_txn.pay.s_W_ID=m_txn.Payment.w_id;
    m_in_txn.pay.s_D_ID=m_txn.Payment.d_id;
    m_in_txn.pay.s_C_ID=m_txn.Payment.c_id;
    m_in_txn.pay.s_C_D_ID=m_txn.Payment.c_d_id;
    m_in_txn.pay.s_C_W_ID=m_txn.Payment.c_w_id;

    m_in_txn.pay.s_H_AMOUNT=(__int64)(m_txn.Payment.h_amount*100);
    // if customer id is zero, then payment is by name
    if (m_txn.Payment.c_id == 0) {
        memcpy(m_in_txn.pay.s_C_LAST,m_txn.Payment.c_last,sizeof(m_txn.Pa
yment.c_last));
    }
    payment_sql(&m_in_txn.pay, &m_out_txn.pay);

    SQL_TIMESTAMP(m_txn.Payment.h_date,&m_out_txn.pay.s_H_DATE_time,d
aterec)

    STR_CPY(m_txn.Payment.w_street_1,m_out_txn.pay.s_W_STREET_1);

    STR_CPY(m_txn.Payment.w_street_2,m_out_txn.pay.s_W_STREET_2);
    STR_CPY(m_txn.Payment.w_city,m_out_txn.pay.s_W_CITY);
    STR_CPY(m_txn.Payment.w_state,m_out_txn.pay.s_W_STATE);
    STR_CPY(m_txn.Payment.w_zip,m_out_txn.pay.s_W_ZIP);
    STR_CPY(m_txn.Payment.d_street_1,m_out_txn.pay.s_D_STREET_1);
    STR_CPY(m_txn.Payment.d_street_2,m_out_txn.pay.s_D_STREET_2);
    STR_CPY(m_txn.Payment.d_city,m_out_txn.pay.s_D_CITY);
    STR_CPY(m_txn.Payment.d_state,m_out_txn.pay.s_D_STATE);
    STR_CPY(m_txn.Payment.d_zip,m_out_txn.pay.s_D_ZIP);
    STR_CPY(m_txn.Payment.c_first,m_out_txn.pay.s_C_FIRST);
    STR_CPY(m_txn.Payment.c_middle,m_out_txn.pay.s_C_MIDDLE);
    // if customer id is zero, then payment is by name
    if (m_in_txn.pay.s_C_ID != 0){
        STR_CPY(m_txn.Payment.c_last,m_out_txn.pay.s_C_LAST);
    }
    STR_CPY(m_txn.Payment.c_street_1,m_out_txn.pay.s_C_STREET_1);
    STR_CPY(m_txn.Payment.c_street_2,m_out_txn.pay.s_C_STREET_2);
    STR_CPY(m_txn.Payment.c_city,m_out_txn.pay.s_C_CITY);
    STR_CPY(m_txn.Payment.c_state,m_out_txn.pay.s_C_STATE);
    STR_CPY(m_txn.Payment.c_zip,m_out_txn.pay.s_C_ZIP);
    STR_CPY(m_txn.Payment.c_phone,m_out_txn.pay.s_C_PHONE);
}

```



```

        SQL_TIMESTAMP(m_txn.Payment.c_since,&m_out_txn.pay.s_C_SINCE_time
,daterec)
        STR_CPY(m_txn.Payment.c_credit,m_out_txn.pay.s_C_CREDIT);
        if (m_in_txn.pay.s_C_ID == 0){
            m_txn.Payment.c_id=m_out_txn.pay.s_C_ID;
        }
        int_to_dbl(m_out_txn.pay.s_C_CREDIT_LIM,m_txn.Payment.c_credit_li
m);
        int_to_dbl(m_out_txn.pay.s_C_DISCOUNT,m_txn.Payment.c_discount);
        int_to_dbl(m_out_txn.pay.s_C_BALANCE,m_txn.Payment.c_balance);
        STR_CPY(m_txn.Payment.c_data,m_out_txn.pay.s_C_DATA);

        ASSIGN_STATUS(m_txn.Payment.exec_status_code,m_out_txn.pay.s_tran
status);

        rc = detach_context(connection_ctx);
        if(rc != 0)
        {
            //throw error
            throw CDBLIBERR( "DBLIB: DETACH PMT" );
        }

        /*if (m_txn.Payment.c_id == 0)
            throw CDBLIBERR( "DBLIB: INVALID CUSTOMER PMT" );
        taken care of in isapi*/

        return;
    }

```

```

void CTPCC_DBLIB::OrderStatus()
{
    int i;
    tm* daterec;
    int rc = attach_context(connection_ctx);
    if(rc != 0)
    {
        //throw error;
        throw CDBLIBERR( "DBLIB: ATTATCH OS" );
    }

    SET0(m_in_txn.os);
    SET0(m_out_txn.os);

    m_in_txn.os.s_W_ID=m_txn.OrderStatus.w_id;
    m_in_txn.os.s_D_ID=m_txn.OrderStatus.d_id;
    m_in_txn.os.s_C_ID=m_txn.OrderStatus.c_id;
    //look up by last
    if(m_txn.OrderStatus.c_id == 0){
        memcpy(m_in_txn.os.s_C_LAST,
m_txn.OrderStatus.c_last,sizeof(m_txn.OrderStatus.c_last));
    }
    ordstat_sql(&m_in_txn.os,&m_out_txn.os);

    STR_CPY(m_txn.OrderStatus.c_first,m_out_txn.os.s_C_FIRST);
    STR_CPY(m_txn.OrderStatus.c_middle,m_out_txn.os.s_C_MIDDLE);
    if(m_txn.OrderStatus.c_id != 0){
        STR_CPY(m_txn.OrderStatus.c_last,m_out_txn.os.s_C_LAST);
    }
}

```

```

    else {
        m_txn.OrderStatus.c_id=m_out_txn.os.s_C_ID;
    }
    int_to_dbl(m_out_txn.os.s_C_BALANCE,m_txn.OrderStatus.c_balance);
    m_txn.OrderStatus.o_id=m_out_txn.os.s_O_ID;
    SQL_TIMESTAMP(m_txn.OrderStatus.o_entry_d,&m_out_txn.os.s_O_ENTRY
_D_time,daterec)
    m_txn.OrderStatus.o_carrier_id=m_out_txn.os.
s_O_CARRIER_ID;
    m_txn.OrderStatus.o_ol_cnt=m_out_txn.os.s_ol_cnt;

    for (i=0; i < m_out_txn.os.s_ol_cnt; i++) {
        m_txn.OrderStatus.OL[i].ol_i_id=m_out_txn.os.item[i].s_OL_I_ID;
        m_txn.OrderStatus.OL[i].ol_supply_w_id=m_out_txn.os.item[i].s_OL_
SUPPLY_W_ID;
        m_txn.OrderStatus.OL[i].ol_quantity=m_out_txn.os.item[i].s_OL_QUA
NTITY;
        int_to_dbl(m_out_txn.os.item[i].s_OL_AMOUNT,m_txn.OrderStatus.OL[
i].ol_amount);
        if (m_out_txn.os.item[i].s_OL_DELIVERY_D_time != 0 ){
            SQL_TIMESTAMP(m_txn.OrderStatus.OL[i].ol_delivery_d,&m_out_txn.os
.item[i].s_OL_DELIVERY_D_time,daterec)
        }
        else
            m_txn.OrderStatus.OL[i].ol_delivery_d.year=0;
    }
}

```

```

        rc = detach_context(connection_ctx);
        if(rc != 0)
        {
            //throw error
            throw CDBLIBERR( "DBLIB: DETACH OS" );
        }

        ASSIGN_STATUS(m_txn.OrderStatus.exec_status_code,m_out_txn.os.s_t
ranstatus);
        /*
        if (m_txn.OrderStatus.o_ol_cnt == 0)
            throw CDBLIBERR( "DBLIB: ERR NO SUCH ORDER OS" );
        else if (m_txn.OrderStatus.c_id == 0 &&
m_txn.OrderStatus.c_last[0] == 0)
            throw CDBLIBERR( "DBLIB: INVALID CUSTOMER OS" );
        */

        return;
    }
}

```

```

void CTPCC_DBLIB::Delivery()
{
    int i;
    int rc = attach_context(connection_ctx);
    if(rc != 0)
    {

```

```

        //throw error;
        throw CDBLIBERR( "DBLIB: ATTACH DEL" );
    }

    SET0(m_in_txn.del);
    SET0(m_out_txn.del);

    m_in_txn.del.s_W_ID=m_txn.Delivery.w_id;
    m_in_txn.del.s_O_CARRIER_ID=m_txn.Delivery.o_carrier_id;
    delivery_sql(&m_in_txn.del,&m_out_txn.del);

    for (i=0; i<10; i++)
    {
        m_txn.Delivery.o_id[i] = m_out_txn.del.s_O_ID[i];
    }

    ASSIGN_STATUS(m_txn.Delivery.exec_status_code,
m_out_txn.del.s_transtatus);

    rc = detach_context(connection_ctx);
    if(rc != 0)
    {
        //throw error
        throw CDBLIBERR( "DBLIB: DETACH DEL" );
    }

    return;
}

```

tpcc_dblib.h

```
#pragma once
```

```

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

```

```

#define ERR_ATTACH 1
#define ERR_DETACH 2
#define ERR_CONNECT 3
#define ERR_GET_CTX 4
#define ERR_DEST 5

```

```

class CDBLIBERR : public CBaseErr
{
public:

```

```

    CDBLIBERR(char * dberrstr)
    {
        m_dberror=0;
        m_dberrstr = dberrstr;
        if (strstr(m_dberrstr,"DETACH"))
            m_dberror=ERR_DETACH;
        else if(strstr(m_dberrstr,"ATTACH"))

```

```

        m_dberror=ERR_ATTACH;
    else if(strstr(m_dberrstr,"CONNECT"))
        m_dberror=ERR_CONNECT;
    else if(strstr(m_dberrstr,"GET_CTX"))
        m_dberror=ERR_GET_CTX;
    else if (strstr(m_dberrstr,"DESTRUCTOR"))
        m_dberror=ERR_DEST;
    };

    ~CDBLIBERR()
    {
        delete [] m_dberrstr;
    };

    int ErrorType() {return ERR_TYPE_DBLIB;};
    int ErrorNum() {return m_dberror;};
    char *ErrorText() {return m_dberrstr;};

private:
    char *m_dberrstr;
    int m_dberror;
};

```

```

class DllDecl CTPCC_DBLIB : public CTPCC_BASE
{
private:
    void *connection_ctx; //connection context

    union
    {
        NEW_ORDER_DATA NewOrder;
        PAYMENT_DATA Payment;
        DELIVERY_DATA Delivery;
        STOCK_LEVEL_DATA StockLevel;
        ORDER_STATUS_DATA OrderStatus;
        m_txn;
    }

    union
    {
        in_neword_struct no;
        in_payment_struct pay;
        in_delivery_struct del;
        in_stocklev_struct stk;
        in_ordstat_struct os;
        m_in_txn;
    }

    union
    {
        out_neword_struct no;
        out_payment_struct pay;
        out_delivery_struct del;
        out_stocklev_struct stk;
        out_ordstat_struct os;
        m_out_txn;
    }
}

```

```

public:
    CTPCC_DBLIB(LPCSTR szUser, LPCSTR szPassword, LPCSTR
szDatabase );
    ~CTPCC_DBLIB(void);

    inline PNEW_ORDER_DATA      BuffAddr_NewOrder()
    { return &m_txn.NewOrder;    };
    inline PPAYMENT_DATA        BuffAddr_Payment()
    { return &m_txn.Payment;      };
    inline PDELIVERY_DATA        BuffAddr_Delivery()
    { return &m_txn.Delivery;     };
    inline PSTOCK_LEVEL_DATA     BuffAddr_StockLevel() { return
&m_txn.StockLevel; };
    inline PORDER_STATUS_DATA    BuffAddr_OrderStatus() { return
&m_txn.OrderStatus; };

    void NewOrder      ();
    void Payment        ();
    void Delivery       ();
    void StockLevel     ();
    void OrderStatus    ();

};

extern "C" DllDecl CTPCC_DBLIB* CTPCC_DBLIB_new
( LPCSTR szUser, LPCSTR szPassword, LPCSTR szDatabase );

typedef CTPCC_DBLIB* (TYPE_CTPCC_DBLIB)( LPCSTR, LPCSTR, LPCSTR);

```

install.c

```

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

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

#include "resource.h"

#define WM_INITTEXT      WM_USER+100

HICON      hIcon;
HINSTANCE  hInst;

DWORD      versionExeMS;
DWORD      versionExeLS;

```

```

DWORD      versionExeMM;
DWORD      versionDllMS;
DWORD      versionDllLS;

// TPC-C registry settings
TPCCREGISTRYDATA    Reg;

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

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

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

BOOL install_com(char *szDllPath);

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

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

    hInst = hInstance;

    InitCommonControls();

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

    iRc = DialogBox(hInstance, MAKEINTRESOURCE(IDD_DIALOG4),
GetDesktopWindow(), LicenseDlgProc);
    if ( iRc )
    {
        iRc = DialogBox(hInstance, MAKEINTRESOURCE(IDD_DIALOG1),
GetDesktopWindow(), MainDlgProc);
        if ( iRc )
        {

```

```

        DialogBoxParam(hInstance,
MAKEINTRESOURCE(IDD_DIALOG2), GetDesktopWindow(), UpdatedDlgProc,
(LPARAM)iRc);
    }
    DestroyIcon(hIcon);
    return 0;
}

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

    switch(uMsg)
    {
        case WM_INITDIALOG:
            hFont = CreateFont(-12, 0, 0, 0, 400, 0, 0, 0, 0,
0, 0, 0, 0, "Arial");
            SendMessage( GetDlgItem(hwnd, IDR_LICENSE1),
WM_SETFONT, (WPARAM)hFont, MAKELPARAM(0, 0) );
            PostMessage(hwnd, WM_INITTEXT, (WPARAM)0,
(LPARAM)0);
            return TRUE;
        case WM_INITTEXT:
            hResInfo = FindResource(hInst,
MAKEINTRESOURCE(IDR_LICENSE1), "LICENSE");
            dwSize = SizeofResource(hInst, hResInfo);
            hRes = LoadResource(hInst, hResInfo );
            pSrc = (BYTE *)LockResource(hRes);
            pDst = (unsigned char *)malloc(dwSize+1);
            if ( pDst )
            {
                memcpy(pDst, pSrc, dwSize);
                pDst[dwSize] = 0;
                SetDlgItemText(hwnd, IDC_LICENSE, (const
char *)pDst);
                free(pDst);
            }
            else
                SetDlgItemText(hwnd, IDC_LICENSE, (const
char *)pSrc);
            return TRUE;
        case WM_DESTROY:
            DeleteObject(hFont);
            return TRUE;
        case WM_COMMAND:
            if ( wParam == IDOK )
                EndDialog(hwnd, TRUE);
            if ( wParam == IDCANCEL )
                EndDialog(hwnd, FALSE);
            default:
                break;
    }
    return FALSE;
}

```

```

BOOL CALLBACK UpdatedDlgProc(HWND hwnd, UINT uMsg, WPARAM wParam, LPARAM
lParam)
{
    switch(uMsg)
    {
        case WM_INITDIALOG:
            switch(lParam)
            {
                case 1:
                case 2:
                    SetDlgItemText(hwnd, IDC_RESULTS,
"TPC-C Web Client Installed");
                    break;
            }
            return TRUE;
        case WM_COMMAND:
            if ( wParam == IDOK )
                EndDialog(hwnd, TRUE);
            break;
        default:
            break;
    }
    return FALSE;
}

BOOL CALLBACK MainDlgProc(HWND hwnd, UINT uMsg, WPARAM wParam, LPARAM
lParam)
{
    PAINTSTRUCT        ps;
    MEMORYSTATUS        memoryStatus;
    OSVERSIONINFO       VI;
    char                szTmp[256];
    static char          szDllPath[256];
    static char          szExePath[256];

    switch(uMsg)
    {
        case WM_INITDIALOG:
            GlobalMemoryStatus(&memoryStatus);
            iMaxPhysicalMemory =
(memoryStatus.dwTotalPhys/ 1048576);

            if ( GetInstallPath(szDllPath) )
            {
                MessageBox(hwnd, "Error internet service
inetsrv is not installed.", NULL, MB_ICONSTOP | MB_OK);
                EndDialog(hwnd, FALSE);
                return TRUE;
            }

            // set default values
            ZeroMemory( &Reg, sizeof(Reg) );
            Reg.dwNumberOfDeliveryThreads = 4;
            Reg.dwMaxConnections = 100;
            Reg.dwMaxPendingDeliveries = 100;
            Reg.eDB_Protocol = DBLIB;
            Reg.eTxnMon = None;
            strcpy(Reg.szDbServer, "");
            strcpy(Reg.szDbName, "tpcc");
            strcpy(Reg.szDbUser, "sa");
            strcpy(Reg.szDbPassword, "");

            iPoolThreadLimit = iMaxPhysicalMemory * 2;
    }
}

```

```

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

        ReadTPCCRegistrySettings( &Reg );
        ReadRegistrySettings();

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

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

        SetDlgItemText(hwnd, IDC_PATH, szDllPath);

        SetDlgItemText(hwnd, ED_DB_SERVER,
Reg.szDbServer);
        SetDlgItemText(hwnd, ED_DB_USER_ID, Reg.szDbUser);
        SetDlgItemText(hwnd, ED_DB_PASSWORD,
Reg.szDbPassword);
        SetDlgItemText(hwnd, ED_DB_NAME, Reg.szDbName);

        SetDlgItemInt(hwnd, ED_THREADS,
Reg.dwNumberOfDeliveryThreads, FALSE);
        SetDlgItemInt(hwnd, ED_MAXCONNECTION,
Reg.dwMaxConnections, FALSE);
        SetDlgItemInt(hwnd, ED_MAXDELIVERIES,
Reg.dwMaxPendingDeliveries, FALSE);
        SetDlgItemInt(hwnd, ED_IIS_MAX_THREAD_POOL_LIMIT,
iPoolThreadLimit, FALSE);
        SetDlgItemInt(hwnd, ED_IIS_THREAD_TIMEOUT,
iThreadTimeout, FALSE);
        SetDlgItemInt(hwnd, ED_IIS_LISTEN_BACKLOG,
iListenBackLog, FALSE);
        SetDlgItemInt(hwnd,
ED_WEB_SERVICE_BACKLOG_QUEUE_SIZE, iAcceptExOutstanding, FALSE);

        CheckDlgButton(hwnd, IDC_DBLIB, 0);
        CheckDlgButton(hwnd, IDC_ODBC, 0);
        if ( Reg.eDB_Protocol == DBLIB )
            CheckDlgButton(hwnd, IDC_DBLIB, 1);
        else
            CheckDlgButton(hwnd, IDC_ODBC, 1);

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

        CheckDlgButton(hwnd, IDC_TM_NONE, 0);
        CheckDlgButton(hwnd, IDC_TM_TUXEDO, 0);
        CheckDlgButton(hwnd, IDC_TM_MTS, 0);

```

```

        CheckDlgButton(hwnd, IDC_TM_ENCINA, 0);
        switch (Reg.eTxnMon)
        {
        case None:
            CheckDlgButton(hwnd, IDC_TM_NONE, 1);
            break;
        case TUXEDO:
            CheckDlgButton(hwnd, IDC_TM_TUXEDO, 1);
            break;
        case ENCINA:
            CheckDlgButton(hwnd, IDC_TM_ENCINA, 1);
            break;
        case COM:
            CheckDlgButton(hwnd, IDC_TM_MTS, 1);
            break;
        }

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

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

    char szFullName[256];
    char szErrTxt[128];

    // read settings from dialog

```

```

        Reg.dwNumberOfDeliveryThreads = GetDlgItemInt(hwnd, ED_THREADS,
&d, FALSE);
        Reg.dwMaxConnections = GetDlgItemInt(hwnd, ED_MAXCONNECTION, &d,
FALSE);
        Reg.dwMaxPendingDeliveries = GetDlgItemInt(hwnd, ED_MAXDELIVERIES,
&d, FALSE);

        GetDlgItemText(hwnd, ED_DB_SERVER, Reg.szDbServer,
sizeof(Reg.szDbServer));
        GetDlgItemText(hwnd, ED_DB_USER_ID, Reg.szDbUser,
sizeof(Reg.szDbUser));
        GetDlgItemText(hwnd, ED_DB_PASSWORD, Reg.szDbPassword,
sizeof(Reg.szDbPassword));
        GetDlgItemText(hwnd, ED_DB_NAME, Reg.szDbName,
sizeof(Reg.szDbName));

        if ( IsDlgButtonChecked(hwnd, IDC_DBLIB) )
        {
            Reg.eDB_Protocol = DBLIB;
            rc = 1;
        }
        else if ( IsDlgButtonChecked(hwnd, IDC_ODBC) )
        {
            Reg.eDB_Protocol = ODBC;
            rc = 2;
        }

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

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

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

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

```

```

        return;
    }

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

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

    // if using COM
    if (Reg.eTxnMon == COM)
    {
        SetDlgItemText(hDlg, IDC_STATUS, "Configuring COM.");
        SendDlgItemMessage(hDlg, IDC_PROGRESS1, PBM_STEPIT, 0,
0);
        UpdateDialog(hDlg);

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

    Sleep(100);

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

    EndDialog(hwnd, rc);
    return;
}

static void ReadRegistrySettings(void)
{
    HKEY    hKey;
    DWORD   size;
    DWORD   type;

    if ( RegOpenKeyEx(HKEY_LOCAL_MACHINE,
"SYSTEM\\CurrentControlSet\\Services\\Inetinfo\\Parameters", 0, KEY_READ,
&hKey) == ERROR_SUCCESS )
    {

```

```

        size = sizeof(iPoolThreadLimit);
        if ( RegQueryValueEx(hKey, "PoolThreadLimit", 0, &type,
(char *)&iPoolThreadLimit, &size) == ERROR_SUCCESS )
            if ( !iPoolThreadLimit )
                iPoolThreadLimit = iMaxPhysicalMemory *
2;

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

        size = sizeof(iListenBackLog);
        if ( RegQueryValueEx(hKey, "ListenBackLog", 0, &type,
(char *)&iListenBackLog, &size) == ERROR_SUCCESS )
            if ( !iListenBackLog )
                iListenBackLog = 15;

        RegCloseKey(hKey);
    }

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

        RegCloseKey(hKey);
    }
}

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

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

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

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

```

```

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

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

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

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

        RegFlushKey(hKey);
        RegCloseKey(hKey);
    }

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

        RegFlushKey(hKey);
        RegCloseKey(hKey);
    }

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

        RegFlushKey(hKey);
        RegCloseKey(hKey);
    }

    return;
}

BOOL CALLBACK CopyDlgProc(HWND hwnd, UINT uMsg, WPARAM wParam, LPARAM
lParam)
{
    if ( uMsg == WM_INITDIALOG )
    {

```

```

        SendDlgItemMessage(hwnd, IDC_PROGRESS1, PBM_SETRANGE, 0,
        MAKELPARAM(0, 16));
        SendDlgItemMessage(hwnd, IDC_PROGRESS1, PBM_SETSTEP,
        (WPARAM)1, 0);
        return TRUE;
    }
    return FALSE;
}

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

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

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

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

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

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

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

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

    CloseHandle(hFile);

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

```

```

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

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

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

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

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

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

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

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

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

```



```

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

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

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

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

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

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

        return 1;
    }

static BOOL GetInstallPath(char *szDllPath)
{
    HKEY    hKey;
    BYTE    szData[256];
    DWORD   sv;
    BOOL    bRc;
    int     len;
    char    *ptr;
    int     iRc;

    szDllPath[0] = 0;
    bRc = TRUE;

```

```

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

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

            RegCloseKey(hKey);
        }

        return bRc;
    }

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

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

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

```

```

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

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

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

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

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

    CloseServiceHandle(schService);
    return TRUE;

ServiceNotRunning:
    CloseServiceHandle(schService);
    return FALSE;
}

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

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

    if (! StartService(schService, 0, NULL) )
        goto StartWWebErr;
    //start Service pending, Check the status until the service is
running.
    if (! QueryServiceStatus(schService, &ssStatus) )
        goto StartWWebErr;
    while( ssStatus.dwCurrentState != SERVICE_RUNNING)
    {

```

```

        dwOldCheckPoint = ssStatus.dwCheckPoint;
        //Save the current checkpoint.
        Sleep(ssStatus.dwWaitHint);
        //Wait for the specified interval.
        if ( !QueryServiceStatus(schService, &ssStatus) )
            //Check the status again.
            break;
        if (dwOldCheckPoint >= ssStatus.dwCheckPoint)
            //Break if the checkpoint has not been incremented.
            break;
    }

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

    CloseServiceHandle(schService);
    return TRUE;

StartWWebErr:
    CloseServiceHandle(schService);
    return FALSE;
}

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

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

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

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

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

```

```

        CloseServiceHandle(schService);
        return TRUE;

StopWebErr:
        CloseServiceHandle(schService);
        return FALSE;
}

static void UpdateDialog(HWND hDlg)
{
    MSG msg;

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

```

install.h

```

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

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

#define BN_LOG                      1001
#define ED_KEEP                     1002
#define ED_THREADS                  1003
#define ED_THREADS2                 1004
#define IDC_PATH                   1007
#define IDC_VERSION                1009
#define IDC_RESULTS                 1010
#define IDC_PROGRESS1              1011
#define IDC_STATUS                  1012
#define IDC_BUTTON1                 1013
#define ED_MAXCONNECTION           1014
#define ED_IIS_MAX_THREAD_POOL_LIMIT 1015
#define ED_WEB_SERVICE_BACKLOG_QUEUE_SIZE 1017
#define ED_IIS_THREAD_TIMEOUT      1018
#define ED_IIS_LISTEN_BACKLOG      1019
#define IDC_DBLIB                  1021
#define IDC_ODBC                   1022
#define IDC_CONNECT_POOL           1023
#define ED_USER_CONNECT_DELAY_TIME 1024

// Next default values for new objects
//

```

install.rc

```

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

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

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

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

////////////////////////////////////
//
// Dialog
//

IDD_DIALOG1 DIALOGEX 0, 0, 219, 351
STYLE DS_MODALFRAME | DS_CENTER | WS_MINIMIZEBOX | WS_POPUP | WS_CAPTION
|
    WS_SYSMENU
CAPTION "TPC-C Web Client Installation Utility"
FONT 8, "MS Sans Serif"
BEGIN
    EDITTEXT        ED_THREADS,164,45,34,12,ES_RIGHT | ES_NUMBER,
        WS_EX_RTLREADING
    EDITTEXT        ED_MAXDELIVERIES,164,59,34,12,ES_RIGHT | ES_NUMBER,
        WS_EX_RTLREADING
    EDITTEXT        ED_MAXCONNECTION,164,73,34,12,ES_RIGHT | ES_NUMBER,
        WS_EX_RTLREADING
    CONTROL         "None",IDC_TM_NONE,"Button",BS_AUTORADIOBUTTON |
        WS_GROUP | WS_TABSTOP,43,100,33,10
    CONTROL         "COM",IDC_TM_MTS,"Button",BS_AUTORADIOBUTTON |
        WS_TABSTOP,43,113,32,10
    CONTROL         "TUXEDO",IDC_TM_TUXEDO,"Button",BS_AUTORADIOBUTTON |
        WS_DISABLED | WS_TABSTOP,106,100,46,10
    CONTROL         "ENCINA",IDC_TM_ENCINA,"Button",BS_AUTORADIOBUTTON |
        WS_DISABLED | WS_TABSTOP,106,113,43,10
    EDITTEXT        ED_DB_SERVER,131,152,67,12,ES_AUTOHSCROLL
    EDITTEXT        ED_DB_USER_ID,131,165,67,12,ES_AUTOHSCROLL
    EDITTEXT        ED_DB_PASSWORD,131,178,67,12,ES_AUTOHSCROLL
    EDITTEXT        ED_DB_NAME,131,191,67,12,ES_AUTOHSCROLL
    CONTROL         "DBLIB",IDC_DBLIB,"Button",BS_AUTORADIOBUTTON |
        WS_GROUP |
        WS_TABSTOP,45,219,39,12
    CONTROL         "ODBC",IDC_ODBC,"Button",BS_AUTORADIOBUTTON |
        WS_DISABLED | WS_TABSTOP,91,219,39,12
    EDITTEXT        ED_IIS_MAX_THREAD_POOL_LIMIT,164,263,34,12,ES_RIGHT |

```

```

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

IDD_DIALOG2 DIALOGEX 0, 0, 117, 62
STYLE DS_SETFOREGROUND | DS_3DLOOK | DS_CENTER | WS_POPUP | WS_BORDER
EXSTYLE WS_EX_STATICEDGE
FONT 12, "MS Sans Serif", 0, 0, 0x1
BEGIN
    DEFPUSHBUTTON "OK",IDOK,33,45,50,9
    CTEXT "HTML TPC-C Installation
Successfull",IDC_RESULTS,7,22,
    102,18,0,WS_EX_CLIENTEDGE
    ICON IDI_ICON2,IDC_STATIC,50,7,18,20,SS_REALSIZEIMAGE,
    WS_EX_TRANSPARENT
END

IDD_DIALOG3 DIALOG DISCARDABLE 0, 0, 91, 40
STYLE DS_SYSMODAL | DS_MODALFRAME | DS_3DLOOK | DS_CENTER | WS_CAPTION
CAPTION "Installing TPC-C Web Client"
FONT 12, "Arial Black"
BEGIN
    CONTROL
"Progress1",IDC_PROGRESS1,"msctls_progress32",WS_BORDER,
    7,20,77,13
    CTEXT "Static",IDC_STATUS,7,7,77,12,SS_SUNKEN

```

```

END

IDD_DIALOG4 DIALOG DISCARDABLE 0, 0, 291, 202
STYLE DS_MODALFRAME | DS_CENTER | WS_POPUP | WS_CAPTION | WS_SYSMENU
CAPTION "Client End User License"
FONT 8, "MS Sans Serif"
BEGIN
    EDITTEXT IDC_LICENSE,7,7,271,167,ES_MULTILINE |
ES_AUTOVSCROLL |
        ES_AUTOHSCROLL | ES_READONLY | WS_VSCROLL |
WS_HSCROLL
    DEFPUSHBUTTON "I &Agree",IDOK,87,181,50,14
    PUSHBUTTON "&Cancel",IDCANCEL,153,181,50,14
END

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

#ifdef APSTUDIO_INVOKED
GUIDELINES DESIGNINFO DISCARDABLE
BEGIN
    IDD_DIALOG1, DIALOG
    BEGIN
        LEFTMARGIN, 22
        RIGHTMARGIN, 209
        VERTGUIDE, 35
        VERTGUIDE, 198
        TOPMARGIN, 4
        BOTTOMMARGIN, 345
    END

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

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

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

#ifdef APSTUDIO_INVOKED

```

```

////////////////////////////////////
////
//
// TEXTINCLUDE
//

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

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

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

#endif    // APSTUDIO_INVOKED

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

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

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

IDR_TPCCDLL        TPCCDLL DISCARDABLE
"..\\..\\isapi_dll\\bin\\tpcc.dll"

#ifdef _MAC
////////////////////////////////////
////
//
// Version
//

VS_VERSION_INFO VERSIONINFO
FILEVERSION 0,4,20,0
PRODUCTVERSION 0,4,20,0
FILEFLAGSMASK 0x3fL
#ifdef _DEBUG
FILEFLAGS 0x1L
#else
FILEFLAGS 0x0L
#endif
FILEOS 0x40004L

```

```

FILETYPE 0x1L
FILESUBTYPE 0x0L
BEGIN
    BLOCK "StringFileInfo"
    BEGIN
        BLOCK "040904b0"
        BEGIN
            VALUE "Comments", "TPC-C Web Client Installer\0"
            VALUE "CompanyName", "Microsoft\0"
            VALUE "FileDescription", "install\0"
            VALUE "FileVersion", "0, 4, 20, 0\0"
            VALUE "InternalName", "install\0"
            VALUE "LegalCopyright", "Copyright 1999\0"
            VALUE "OriginalFilename", "install.exe\0"
            VALUE "ProductName", "Microsoft install\0"
            VALUE "ProductVersion", "0, 4, 20, 0\0"
        END
    END
    BLOCK "VarFileInfo"
    BEGIN
        VALUE "Translation", 0x409, 1200
    END
END

#endif    // !_MAC

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

IDR_LICENSE1        LICENSE DISCARDABLE    "license.txt"

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

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

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

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

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

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

```

```

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

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

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

IDR_COMTYPLIB_DLL   COM_TYPLIB DISCARDABLE
"..\\..\\tpcc_com_all\\src\\tpcc_com_all.tlb"
#ifdef   // English (U.S.) resources
////////////////////////////////////
////
//

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

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

```

install_com.cpp

```

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

#define _WIN32_WINNT 0x0500

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

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

```

```

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

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

    _bstr_t                bstrTemp, bstrTemp2, bstrTemp3,
bstrTemp4;
    _bstr_t                bstrDllPath = szDllPath;
    _variant_t             vTmp, vKey;
    long                   lActProp, lCount, lCountCo,
lCountItf, lCountMethod;
    bool                   bTmp;

    CoInitializeEx(NULL, COINIT_MULTITHREADED);

    HRESULT hr = CoCreateInstance(CLSID_COMAdminCatalog,
                                NULL,

    CLSCTX_INPROC_SERVER,
    IID_ICOMAdminCatalog,
    (void**) &pCOMAdminCat);

    if (!SUCCEEDED(hr)) goto Error;

    bstrTemp = "Applications";

    // Attempt to connect to "Applications" in the Catalog
    hr = pCOMAdminCat->GetCollection(bstrTemp,
    (IDispatch**) &pCatalogCollectionApp);
    if (!SUCCEEDED(hr)) goto Error;

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

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

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

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

        if (wcsncmp(vTmp.bstrVal, L"TPC-C"))

```

```

        {
            lCount--;
            continue;
        }
        else
        {
            hr = pCatalogCollectionApp->Remove(lCount - 1);
            if (!SUCCEEDED(hr)) goto Error;
            break;
        }
    }

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

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

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

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

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

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

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

    pCatalogObjectApp->Release();
    pCatalogObjectApp = NULL;

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

    hr = pCOMAdminCat->InstallComponent(bstrTemp,
        bstrTemp2,

```

```

        bstrTemp3,
        bstrTemp4);

    switch (hr){
    case E_INVALIDARG : break;
    case E_OUTOFMEMORY :break;
    case E_UNEXPECTED: break;
    case E_FAIL:break;
    }

    if (!SUCCEEDED(hr)) goto Error;

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

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

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

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

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

        bstrTemp = "ConstructionEnabled";
        bTmp = TRUE;
        vTmp = bTmp;
        hr = pCatalogObjectCo->put_Value(bstrTemp, vTmp);
        if (!SUCCEEDED(hr)) goto Error;

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

        bstrTemp = "JustInTimeActivation";
        bTmp = TRUE;
        vTmp = bTmp;
        hr = pCatalogObjectCo->put_Value(bstrTemp, vTmp);
        if (!SUCCEEDED(hr)) goto Error;

        bstrTemp = "MaxPoolSize";
        vTmp.Clear(); // clear variant so it isn't stored as a
bool (_variant_t feature)
        vTmp = (long)30;
        hr = pCatalogObjectCo->put_Value(bstrTemp, vTmp);
        if (!SUCCEEDED(hr)) goto Error;

```

```

        bstrTemp = "ObjectPoolingEnabled";
        bTmp = TRUE;
        vTmp = bTmp;
        hr = pCatalogObjectCo->put_Value(bstrTemp, vTmp);
        if (!SUCCEEDED(hr)) goto Error;

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

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

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

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

        // iterate through interfaces in component
        while (lCountItf > 0)
        {
            hr = pCatalogCollectionItf->get_Item(lCountItf - 1,
(IDispatch**) &pCatalogObjectItf);
            if (!SUCCEEDED(hr)) goto Error;

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

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

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

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

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

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

                pCatalogObjectMethod->Release();

```

```

        pCatalogObjectMethod = NULL;

        lCountMethod--;
    }

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

    pCatalogObjectItf->Release();
    pCatalogObjectItf = NULL;

    lCountItf--;
}

pCatalogObjectCo->Release();
pCatalogObjectCo = NULL;

lCountCo--;
}

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

pCatalogCollectionApp->Release();
pCatalogCollectionApp = NULL;

pCatalogCollectionCo->Release();
pCatalogCollectionCo = NULL;

pCatalogCollectionItf->Release();
pCatalogCollectionItf = NULL;

pCatalogCollectionMethod->Release();
pCatalogCollectionMethod = NULL;

Error:
    CoUninitialize();

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

NULL,
hr,

MAKELANGID(LANG_NEUTRAL, SUBLANG_DEFAULT),

(LPTSTR) &lpBuf,

0,

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

```



```

    }
    else
        return FALSE;
}

```

resource.h

```

//{{{NO_DEPENDENCIES}}
// Microsoft Developer Studio generated include file.
// Used by install.rc
//
#define IDD_DIALOG1 101
#define IDI_ICON1 102
#define IDR_TPCCDLL 103
#define IDD_DIALOG2 105
#define IDI_ICON2 106
#define IDR_DELIVERY 107
#define IDD_DIALOG3 108
#define IDR_LICENSE1 112
#define IDD_DIALOG4 113
#define IDR_TPCCOBJ1 117
#define IDR_TPCCSTUB1 118
#define IDR_DBLIB_DLL 122
#define IDR_ODBC_DLL 123
#define IDR_TUXEDO_APP 124
#define IDR_TUXEDO_DLL 125
#define IDR_COM_DLL 126
#define IDR_COMPS_DLL 127
#define IDR_COMALL_DLL 128
#define IDR_COMTYPLIB_DLL 129
#define BN_LOG 1001
#define ED_KEEP 1002
#define ED_THREADS 1003
#define ED_THREADS2 1004
#define IDC_PATH 1007
#define IDC_VERSION 1009
#define IDC_RESULTS 1010
#define IDC_PROGRESS1 1011
#define IDC_STATUS 1012
#define IDC_BUTTON1 1013
#define ED_MAXCONNECTION 1014
#define ED_IIS_MAX_THREAD_POOL_LIMIT 1015
#define ED_MAXDELIVERIES 1016
#define ED_WEB_SERVICE_BACKLOG_QUEUE_SIZE 1017
#define ED_IIS_THREAD_TIMEOUT 1018
#define ED_IIS_LISTEN_BACKLOG 1019
#define IDC_DBLIB 1021
#define IDC_LICENSE 1022
#define IDC_ODBC 1022
#define IDC_CONNECT_POOL 1023
#define ED_DB_SERVER 1023
#define ED_USER_CONNECT_DELAY_TIME 1024
#define ED_DB_USER_ID 1024
#define IDC_MTS 1025
#define IDC_TM_MTS 1025
#define IDC_TM_TUXEDO 1026
#define IDC_TM_NONE 1027
#define ED_DB_PASSWORD 1028
#define ED_DB_NAME 1029
#define IDC_TM_ENCINA 1030

// Next default values for new objects
//

```

```

#ifdef APSTUDIO_INVOKED
#ifdef APSTUDIO_READONLY_SYMBOLS
#define _APS_NEXT_RESOURCE_VALUE 133
#define _APS_NEXT_COMMAND_VALUE 40001
#define _APS_NEXT_CONTROL_VALUE 1031
#define _APS_NEXT_SYMED_VALUE 101
#endif
#endif

```

resource.h

```

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

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

```

tpcc.cpp

```

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

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

#include <sqltypes.h>

#ifdef ICECAP

```

```

#include <icapexp.h>
#endif

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

#include "txnlog.h"

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

// Txn monitor layer includes
#include "..\..\tm_com_dll\src\tpcc_com.h"              // COM
Services implementation on TPC-C txns

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

#define LEN_ERR_STRING 256

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

char          szMyComputerName[MAX_COMPUTERNAME_LENGTH+1];

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

// The WEBCLIENT_VERSION string specifies the version level of this web
client interface.
// The RTE must be synchronized with the interface level on login,
otherwise the login
// will fail. This is a sanity check to catch problems resulting from
mismatched versions
// of the RTE and web client.
#define WEBCLIENT_VERSION "410"

static CRITICAL_SECTION      TermCriticalSection;

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

TYPE_CTPCC_DBLIB      *pCTPCC_DBLIB_new;;
TYPE_CTPCC_COM        *pCTPCC_COM_new;

// For deferred Delivery txns:

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

HANDLE                  hWorkerSemaphore      =
INVALID_HANDLE_VALUE;

HANDLE                  hDoneEvent            =
INVALID_HANDLE_VALUE;

```

```

HANDLE                  *pDeliHandles          = NULL;

// configuration settings from registry
TPCCREGISTRYDATA      Reg;

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

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

/* FUNCTION: DllMain
 *
 * PURPOSE: This function is the entry point for the DLL. This
implementation is based on the
 * fact that DLL_PROCESS_ATTACH is only called from
the inet service once.
 *
 * ARGUMENTS: HANDLE hModule module handle
 *              DWORD ul_reason_for_call reason for
call
 *              LPVOID lpReserved
 * reserved for future use
 *
 * RETURNS:      BOOL FALSE errors
occured in initialization
 *              TRUE
 *              DLL successfully initialized
 */

BOOL APIENTRY DllMain(HANDLE hModule, DWORD ul_reason_for_call, LPVOID
lpReserved)
{
    DWORD i;
    char szEvent[LEN_ERR_STRING] = "\0";
    char szLogFile[128];
    char szDllName[128];

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

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

```

```

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

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

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

        TermInit();

        // load DLL for txn monitor
        if (Reg.eTxnMon == COM)
        {
            strcpy( szDllName, Reg.szPath );
            strcat( szDllName, "tpcc_com.dll");
            hLibInstanceTm =

LoadLibrary( szDllName );

            if (hLibInstanceTm == NULL)
                throw new
CWEBCINT_ERR( ERR_LOADDLL_FAILED, szDllName, GetLastError() );

            // get function pointer to wrapper
            pCTPCC_COM_new = (TYPE_CTPCC_COM*)
GetProcAddress(hLibInstanceTm, "CTPCC_COM_new");
            if (pCTPCC_COM_new == NULL)
                throw new
CWEBCINT_ERR( ERR_GETPROCADDR_FAILED, szDllName, GetLastError() );
        }

        // load DLL for database connection
        if ((Reg.eTxnMon == None) ||

(dwNumDeliveryThreads > 0))
        {
            if (Reg.eDB_Protocol == DBLIB)
            {
                strcpy( szDllName,
Reg.szPath );
                strcat( szDllName,
"tpcc_dblib.dll");
                LoadLibrary( szDllName );

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

                // get function pointer to
wrapper for class constructor
                pCTPCC_DBLIB_new =
(TYPE_CTPCC_DBLIB*) GetProcAddress(hLibInstanceDb, "CTPCC_DBLIB_new");
                if (pCTPCC_DBLIB_new ==
NULL)
                    throw new
CWEBCINT_ERR( ERR_GETPROCADDR_FAILED, szDllName, GetLastError() );

```

```

        }
    }
    if (dwNumDeliveryThreads)
    {
        // for deferred delivery txns:
        hDoneEvent = CreateEvent( NULL,
TRUE /* manual reset */, FALSE /* initially not signalled */, NULL );

        InitializeCriticalSection(&DelBuffCriticalSection);
        hWorkerSemaphore =
CreateSemaphore( NULL, 0, dwDelBuffSize, NULL );
        dwDelBuffFreeCount = dwDelBuffSize;

        // create unique log file name
        SYSTEMTIME Time;
        GetLocalTime( &Time );
        wsprintf( szLogFile, "%sdelivery-
%2.2d%2.2d%2.2d-%2.2d%2.2d.log",
Time.wYear % 100, Time.wMonth, Time.wDay, Time.wHour, Time.wMinute );
        Reg.szPath,
        txndelilog = new
CTxnLog(szLogFile);

        //write event into txn log for
START

        if(txndelilog-
>WriteCtrlRecToLog(TXN_EVENT_START, szMyComputerName,
sizeof(szMyComputerName)))
            throw CBaseErr("Cannot open
delivery log"); //we could not write to the delivery so stop now

        // allocate structures for delivery
        pDeliHandles = new
HANDLE[dwNumDeliveryThreads];
        pDelBuff = new
DELIVERY_TRANSACTION[dwDelBuffSize];
        // launch DeliveryWorkerThread to
perform actual delivery txns
        for(i=0; i<dwNumDeliveryThreads;
i++)
        {
            pDeliHandles[i] = (HANDLE)
_beginthread( DeliveryWorkerThread, 0, NULL );
            if (pDeliHandles[i] ==
INVALID_HANDLE_VALUE)
                throw new
CWEBCINT_ERR( ERR_DELIVERY_THREAD_FAILED );
        }
        break;

        case DLL_PROCESS_DETACH:
            if (dwNumDeliveryThreads)
            {
                if (txndelilog != NULL)
                {
                    //write event into txn log
for STOP

```

```

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

        // This will do a clean
shutdown of the delivery log file      CTxnLog *txnDelilogLocal =
txnDelilog;                          txnDelilog= NULL;
                                      delete txnDelilogLocal;
                                      }
                                      delete [] pDeliHandles;
                                      delete [] pDelBuff;
                                      CloseHandle( hWorkerSemaphore );
                                      CloseHandle( hDoneEvent );

DeleteCriticalSection(&DelBuffCriticalSection);
    }

DeleteCriticalSection(&TermCriticalSection);

    if (hLibInstanceTm != NULL)
        FreeLibrary( hLibInstanceTm );
    hLibInstanceTm = NULL;

    if (hLibInstanceDb != NULL)
        FreeLibrary( hLibInstanceDb );
    hLibInstanceDb = NULL;

    Sleep(500);
    break;

    default:
        /* nothing */;
    }
}
catch (CBaseErr *e)
{
    WriteMessageToEventLog( e->ErrorText() );
    delete e;
    TerminateExtension(0);
    return FALSE;
}
catch (...)
{
    WriteMessageToEventLog(TEXT("Unhandled exception. DLL
could not load."));
    TerminateExtension(0);
    return FALSE;
}

return TRUE;
}

/* FUNCTION: GetExtensionVersion
*
* PURPOSE: This function is called by the inet service when the DLL
is first loaded.
*

```

```

* ARGUMENTS: HSE_VERSION_INFO *pVer passed in structure in
which to place expected version number.
*
* RETURNS: TRUE inet service expected return value.
*/

BOOL WINAPI GetExtensionVersion(HSE_VERSION_INFO *pVer)
{
    pVer->dwExtensionVersion = MAKELONG(HSE_VERSION_MINOR,
HSE_VERSION_MAJOR);
    lstrcpy(pVer->lpszExtensionDesc, "TPC-C Server.",
HSE_MAX_EXT_DLL_NAME_LEN);

    return TRUE;
}

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

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

    TermDeleteAll();
    return TRUE;
}

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

DWORD WINAPI HttpExtensionProc(EXTENSION_CONTROL_BLOCK *pECB)
{

```

```

        int          iCmd, FormId, TermId, iSyncId;
        char          szBuffer[4096];

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

//DebugBreak();

#ifdef ICECAP
    StartCAP();
#endif

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

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

                throw new
CWEBCLNT_ERR( ERR_INVALID_TERMID );
            }

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

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

        switch(iCmd)
        {
        case 0:
            WelcomeForm(pECB, szBuffer);
            break;

        case 1:
            switch( FormId )
            {
                case WELCOME_FORM:
                case MAIN_MENU_FORM:
                    break;
                case NEW_ORDER_FORM:
                    ProcessNewOrderForm(pECB, TermId,
szBuffer);
                    break;
            }
        }
    }

```

```

        szBuffer);

        case PAYMENT_FORM:
            ProcessPaymentForm(pECB, TermId,
            szBuffer);
            break;
        case DELIVERY_FORM:
            ProcessDeliveryForm(pECB, TermId,
            szBuffer);
            break;
        case ORDER_STATUS_FORM:
            ProcessOrderStatusForm(pECB, TermId,
            szBuffer);
            break;
        case STOCK_LEVEL_FORM:
            ProcessStockLevelForm(pECB, TermId,
            szBuffer);
            break;
    }
    break;

case 2:
    // new-order selected from menu; display new-order
    input form
    MakeNewOrderForm(TermId, NULL, INPUT_FORM,
    szBuffer);
    break;

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

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

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

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

case 7:
    // ExitCmd
    TermDelete(TermId);
    WelcomeForm(pECB, szBuffer);
    break;

case 8:
    SubmitCmd(pECB, szBuffer);
    break;

case 9:
    // menu
    MakeMainMenuForm(TermId,
Term.pClientData[TermId].iSyncId, szBuffer);
    break;

case 10:
    // CMD=Clear

```

```

        // resets all connections; should only be used
when no other connections are active
        TermDeleteAll();
        TermInit();
        WelcomeForm(pECB, szBuffer);
        break;
    case 11: // CMD=Stats
        StatsCmd(pECB, szBuffer);
        break;
    }
}
catch (CBaseErr *e)
{
    ErrorForm( pECB, e->ErrorType(), e->ErrorNum(), TermId,
iSyncId, e->ErrorText(), szBuffer );
    delete e;
}
catch (...)
{
    ErrorForm( pECB, ERR_TYPE_WEBDLL, 0, TermId, iSyncId,
"Error: Unhandled exception in Web Client.", szBuffer );
}

#ifdef ICECAP
    StopCAP();
#endif

    lpbSize = strlen(szBuffer);
    wsprintf(szHeader1,
        "Content-Type: text/html\r\n"
        "Content-Length: %d\r\n"
        "Connection: Keep-Alive\r\n\r\n", lpbSize);
    strcat( szHeader1, szBuffer );

    (*pECB->ServerSupportFunction)(pECB->ConnId,
HSE_REQ_SEND_RESPONSE_HEADER, szHeader, (LPDWORD) &dwSize,
(LPDWORD)szHeader1);

    //finish up and keep connection
    pECB->dwHttpStatusCode = 200;
    return HSE_STATUS_SUCCESS_AND_KEEP_CONN;
}

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

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

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

    if (hEventSource != NULL)
    {
        ReportEvent(hEventSource, // handle of event source
EVENTLOG_ERROR_TYPE, // event type

```

```

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

(VOID) DeregisterEventSource(hEventSource);
}

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

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

    DELIVERY_TRANSACTION delivery;
    PDELIVERY_DATA pDeliveryData;
    TXN_RECORD_TPCC_DELIV_DEF txnDeliRec;

    DWORD index;
    HANDLE handles[2];
    __int64 trans_end; //delivery transaction
    finished time //delivery transaction start time
    __int64 trans_start;

    assert(txnDeliRec != NULL);

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

```

```

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

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

                ZeroMemory(&txnDeliRec,
                sizeof(txnDeliRec));

                // make a local copy of current entry from
                delivery buffer and increment buffer index

                EnterCriticalSection(&DelBuffCriticalSection);
                delivery = *(pDelBuff+dwDelBuffBusyIndex);
                dwDelBuffFreeCount++;
                dwDelBuffBusyIndex++;
                if (dwDelBuffBusyIndex == dwDelBuffSize)
                // wrap-around if at end of buffer
                    dwDelBuffBusyIndex = 0;

                LeaveCriticalSection(&DelBuffCriticalSection);

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

                txnDeliRec.TxnStartT0=delivery.queue;
                trans_start=GetCurrentTime();
                pTxn->Delivery();
                trans_end=GetCurrentTime();

                //log txn
                txnDeliRec.TxnStatus = ERR_SUCCESS;
                for (int i=0; i<10; i++)
                    txnDeliRec.o_id[i] = pDeliveryData-
                >o_id[i];

                txnDeliRec.TxnStartT0;
                txnDeliRec.DeltaT4 = trans_end -
                trans_start;
                txnDeliRec.DeltaTxnExec = trans_end -
                trans_start;

                if (txnDelilog != NULL)

```

```

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

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

            delete e;
        }
        catch (...)
        {
            // unhandled exception; shouldn't happen; not much
            we can do...
            WriteMessageToEventLog(TEXT("Unhandled exception
caught in DeliveryWorkerThread."));
        }
    }

ErrorExit:
    delete pTxn;
    _endthread();
}

/* FUNCTION: PostDeliveryInfo
 *
 * PURPOSE: This function enters the delivery txn into the deferred
delivery buffer.
 *
 * RETURNS: BOOL FALSE delivery information posted
successfully
TRUE error cannot post
delivery info
 */

BOOL PostDeliveryInfo(short w_id, short o_carrier_id)
{
    BOOL bError;

    EnterCriticalSection(&DelBuffCriticalSection);
    if (dwDelBuffFreeCount > 0)
    {
        bError = FALSE;
        (pDelBuff+dwDelBuffFreeIndex)->w_id =
        w_id;
        (pDelBuff+dwDelBuffFreeIndex)->o_carrier_id =
        o_carrier_id;
        (pDelBuff+dwDelBuffFreeIndex)->queue=GetCurrentTime();

        dwDelBuffFreeCount--;
        dwDelBuffFreeIndex++;
        if (dwDelBuffFreeIndex == dwDelBuffSize)
            dwDelBuffFreeIndex = 0; // wrap-
            around if at end of buffer
    }
}

```

```

        else
            // No free buffers. Return an error, which indicates
            // that the delivery buffer is full.
            // Most likely, the number of delivery worker threads
            // needs to be increased to keep up
            // with the txn rate.
            bError = TRUE;
            LeaveCriticalSection(&DelBuffCriticalSection);

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

        return bError;
    }

    /* FUNCTION: ProcessQueryString
    * PURPOSE: This function extracts the relevent information out of
    * the http command passed in from
    * the browser.
    *
    * COMMENTS: If this is the initial connection i.e. client is at
    * welcome screen then
    * there will not be a terminal id or current
    * form id. If this is the case
    * then the pTermid and pFormid return values
    * are undefined.
    */

    void ProcessQueryString(EXTENSION_CONTROL_BLOCK *pECB, int *pCmd, int
    *pFormId, int *pTermId, int *pSyncId)
    {
        char *ptr = pECB->lpszQueryString;
        char szBuffer[25];
        int i;

        //allowable client command strings i.e. CMD=command
        static char *szCmds[] =
        {
            "Process", "..NewOrder..", "..Payment..", "..Delivery..",
            "..Order-Status..", "..Stock-Level..",
            "..Exit..", "Submit", "Menu", "Clear", "Stats", ""
        };

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

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

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

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

        // see which command it matches

```

```

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

    /* FUNCTION: void WelcomeForm
    */

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

        //welcome to tpc-c html form buffer, this is first form client
        //sees.
        strcpy( szBuffer, "<HTML><HEAD><TITLE>TPC-C Web
Client</TITLE></HEAD><BODY>"
        "<B><BIG>Microsoft TPC-C
Web Client (ver 4.20)</BIG></B> <BR> <BR>"
        "<font face=\"Courier
New\"><PRE>"
        "Compiled: \"__DATE__",
        "__TIME__" <BR>"
        "Source: \"__FILE__"
        "\"</PRE></font>"
        "<FORM ACTION=\"tpcc.dll\"
METHOD=\"GET\">"
        "NAME=\"STATUSID\" VALUE=\"0\">"
        "NAME=\"ERROR\" VALUE=\"0\">"
        "NAME=\"FORMID\" VALUE=\"1\">"
        "NAME=\"TERMID\" VALUE=\"0\">"
        "NAME=\"SYNCID\" VALUE=\"0\">"
        "NAME=\"VERSION\" VALUE=\"\" WEBCLIENT_VERSION \"\">"
        );

        sprintf( szTmp, "Configuration Settings: <BR><font
face=\"Courier New\" color=\"blue\"><PRE>"
        "Txn Monitor =
<B>%s</B><BR>"
        "Database protocol =
<B>%s</B><BR>"
        "Max Connections =
<B>%d</B><BR>"
        "# of Delivery Threads =
<B>%d</B><BR>"
        "Max Pending Deliveries =
<B>%d</B><BR>"

```



```

        , szTxnMonNames[Reg.eTxnMon],
szDBNames[Reg.eDB_Protocol],
        Reg.dwMaxConnections, dwNumDeliveryThreads,
dwDelBuffSize );
    strcat( szBuffer, szTmp);

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

    if (Reg.eTxnMon == None)
        // connection options may be specified when not using a
txn monitor
        sprintf( szTmp,          "Please enter your database options
for this connection:<BR>"
                "<font face=\"Courier New\"
color=\"blue\"><PRE>"
                "DB Server      = <INPUT
NAME=\"db_server\" SIZE=20 VALUE=\"%s\"><BR>"
                "DB User ID    = <INPUT
NAME=\"db_user\" SIZE=20 VALUE=\"%s\"><BR>"
                "DB Password  = <INPUT
NAME=\"db_passwd\" SIZE=20 VALUE=\"%s\"><BR>"
                "DB Name      = <INPUT
NAME=\"db_name\" SIZE=20 VALUE=\"%s\"><BR>"
                "</PRE></font>"
                Reg.szDbServer, Reg.szDbUser,
Reg.szDbPassword, Reg.szDbName );
    else
        // if using a txn monitor, connection options are
determined from registry; can't
        // set per user. show options fyi
        sprintf( szTmp,          "Database options which will be
used by the transaction monitor:<BR>"
                "<font face=\"Courier New\"
                "DB Server      =
                "DB User ID    =
                "DB Password  =
                "DB Name      =
                "</PRE></font>"
                Reg.szDbServer, Reg.szDbUser,
Reg.szDbPassword, Reg.szDbName );
    strcat( szBuffer, szTmp);

    sprintf( szTmp,          "Please enter your Warehouse and District
for this session:<BR>"
            "<font face=\"Courier New\"
            "Warehouse ID = <INPUT NAME=\"w_id\"
SIZE=4><BR>"
            "District ID  = <INPUT
NAME=\"d_id\" SIZE=2><BR>"

```

```

        "</PRE></font><HR>"
        "<INPUT TYPE=\"submit\"
        "</FORM></BODY></HTML>");
    }

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

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

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

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

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

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

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

    iNewTerm = TermAdd();
    Term.pClientData[iNewTerm].w_id = w_id;

```

```

Term.pClientData[iNewTerm].d_id = d_id;
try
{
    if (Reg.eTxnMon == COM)
        Term.pClientData[iNewTerm].pTxn =
pCTPCC_COM_new( Reg.bCOM_SinglePool );
    else if (Reg.eDB_Protocol == DBLIB)
        Term.pClientData[iNewTerm].pTxn =
pCTPCC_DBLIB_new( szUser, szPassword, szDatabase );
}
catch (...)
{
    TermDelete(iNewTerm);
    throw;          // pass exception upward
}

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

/* FUNCTION: StatsCmd
 * PURPOSE:   This function returns to the browser the total number of
 *            active terminal ids. This routine is for development/debugging purposes.
 */

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

    EnterCriticalSection(&TermCriticalSection);

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

    LeaveCriticalSection(&TermCriticalSection);

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

char *CWEBCLNT_ERR::ErrorText()
{
    static SERRORMSG errorMsgs[] =
    {
        { ERR_COMMAND_UNDEFINED,
        "Command undefined." },
        { ERR_D_ID_INVALID,
        "Invalid District ID Must be 1 to 10." },
    },
};

```

```

{
    ERR_DELIVERY_CARRIER_ID_RANGE,
    "Delivery Carrier ID out of range must be 1 - 10."
},
{
    ERR_DELIVERY_CARRIER_INVALID,
    "Delivery Carrier ID invalid must be numeric 1 - 10."
},
{
    ERR_DELIVERY_MISSING_OCD_KEY,
    "Delivery missing Carrier ID key \"OCD*\"."
},
{
    ERR_DELIVERY_THREAD_FAILED,
    "Could not start delivery worker thread."
},
{
    ERR_GETPROCADDR_FAILED,
    "Could not map proc in DLL. GetProcAddr error. DLL="
},
{
    ERR_HTML_ILL_FORMED,
    "Required key field is missing from HTML string."
},
{
    ERR_INVALID_SYNC_CONNECTION,
    "Invalid Terminal Sync ID."
},
{
    ERR_INVALID_TERMID,
    "Invalid Terminal ID."
},
{
    ERR_LOADDLL_FAILED,
    "Load of DLL failed. DLL="
},
{
    ERR_MAX_CONNECTIONS_EXCEEDED,
    "No connections available. Max Connections is probably too low."
},
{
    ERR_MISSING_REGISTRY_ENTRIES,
    "Required registry entries are missing. Rerun INSTALL to correct."
},
{
    ERR_NEWORDER_CUSTOMER_INVALID,
    "New Order customer id invalid data type, range = 1 to 3000."
},
{
    ERR_NEWORDER_CUSTOMER_KEY,
    "New Order missing Customer key \"CID*\"."
},
{
    ERR_NEWORDER_DISTRICT_INVALID,
    "New Order District ID Invalid range 1 - 10."
},
{
    ERR_NEWORDER_FORM_MISSING_DID,
    "New Order missing District key \"DID*\"."
},
{
    ERR_NEWORDER_ITEMID_INVALID,
    "New Order Item Id is wrong data type, must be numeric."
},
{
    ERR_NEWORDER_ITEMID_RANGE,
    "New Order Item Id is out of range. Range = 1 to 999999."
},
{
    ERR_NEWORDER_ITEMID_WITHOUT_SUPPW,
    "New Order Item_Id field entered without a corresponding Supp_W."
},
{
    ERR_NEWORDER_MISSING_IID_KEY,
    "New Order missing Item Id key \"IID*\"."
},
{
    ERR_NEWORDER_MISSING_QTY_KEY,
    "New Order Missing Qty key \"Qty##*\"."
},
{
    ERR_NEWORDER_MISSING_SUPPW_KEY,
    "New Order missing Supp_W key \"SP##*\"."
},
{
    ERR_NEWORDER_NOITEMS_ENTERED,
    "New Order No order lines entered."
},
};

```

```

    {
        ERR_NEWORDER_QTY_INVALID,
        "New Order Qty invalid must be numeric range 1 - 99."
    },
    {
        ERR_NEWORDER_QTY_RANGE,
        "New Order Qty is out of range. Range = 1 to 99."
    },
    {
        ERR_NEWORDER_QTY_WITHOUT_SUPPW,
        "New Order Qty field entered without a corresponding Supp_W."
    },
    {
        ERR_NEWORDER_SUPPW_INVALID,
        "New Order Supp_W invalid data type must be numeric."
    },
    {
        ERR_NO_SERVER_SPECIFIED,
        "No Server name specified."
    },
    {
        ERR_ORDERSTATUS_CID_AND_CLT,
        "Order Status Only Customer ID or Last Name may be entered, not
both."
    },
    {
        ERR_ORDERSTATUS_CID_INVALID,
        "Order Status Customer ID invalid, range must be numeric 1 -
3000."
    },
    {
        ERR_ORDERSTATUS_CLT_RANGE,
        "Order Status Customer last name longer than 16 characters."
    },
    {
        ERR_ORDERSTATUS_DID_INVALID,
        "Order Status District invalid, value must be numeric 1 - 10."
    },
    {
        ERR_ORDERSTATUS_MISSING_CID_CLT,
        "Order Status Either Customer ID or Last Name must be entered."
    },
    {
        ERR_ORDERSTATUS_MISSING_CID_KEY,
        "Order Status missing Customer key \"CID*\"."
    },
    {
        ERR_ORDERSTATUS_MISSING_CLT_KEY,
        "Order Status missing Customer Last Name key \"CLT*\"."
    },
    {
        ERR_ORDERSTATUS_MISSING_DID_KEY,
        "Order Status missing District key \"DID*\"."
    },
    {
        ERR_PAYMENT_CDI_INVALID,
        "Payment Customer district invalid must be numeric."
    },
    {
        ERR_PAYMENT_CID_AND_CLT,
        "Payment Only Customer ID or Last Name may be entered, not both."
    },
    {
        ERR_PAYMENT_CUSTOMER_INVALID,
        "Payment Customer data type invalid, must be numeric."
    },
    {
        ERR_PAYMENT_CWI_INVALID,
        "Payment Customer Warehouse invalid, must be numeric."
    },
    {
        ERR_PAYMENT_DISTRICT_INVALID,
        "Payment District ID is invalid, must be 1 - 10."
    },
    {
        ERR_PAYMENT_HAM_INVALID,
        "Payment Amount invalid data type must be numeric."
    },
    {
        ERR_PAYMENT_HAM_RANGE,
        "Payment Amount out of range, 0 - 9999.99."
    },
    {
        ERR_PAYMENT_LAST_NAME_TOO_LONG,
        "Payment Customer last name longer than 16 characters."
    },
    },

```

```

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

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

if (m_szTextDetail)
    strcat( szTmp, m_szTextDetail );
if (m_SystemErr)

```

```

        wsprintf( szTmp+strlen(szTmp), " Error=%d",
m_SystemErr );

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

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

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

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

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

    *pQueryString = ptr;
    return;
}

```

```

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

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

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

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

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

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

```

```

        return 0;
    }

    *pQueryString = ptr;
    return atoi(ptr0);
}

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

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

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

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

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

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

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

    LeaveCriticalSection(&TermCriticalSection);
}

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

```

```

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

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

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

    LeaveCriticalSection(&TermCriticalSection);
}

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

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

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

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

```

```

        // are being attempted than were specified as "Max
Connections" at install. In this case,
        // do not bump existing connection; instead, return error
to requestor.
        if ((GetTickCount() - iTickCount) < 60000)
        {
            LeaveCriticalSection(&TermCriticalSection);
            throw new
CWEBCINT_ERR( ERR_MAX_CONNECTIONS_EXCEEDED );
        }
    }

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

    LeaveCriticalSection(&TermCriticalSection);
    return iNewTerm;
}

/* FUNCTION: TermDelete
 *
 * PURPOSE: This function makes a terminal entry in the Term array
available for reuse.
 *
 * ARGUMENTS: int id
Terminal id of client exiting
 */
void TermDelete(int id)
{
    if ( id > 0 && id < Term.iNumEntries )
    {
        delete Term.pClientData[id].pTxn;

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

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

        LeaveCriticalSection(&TermCriticalSection);
    }
}

/* FUNCTION: MakeErrorForm
 */
void ErrorForm(EXTENSION_CONTROL_BLOCK *pECB, int iType, int iErrorNum,
int iTermId, int iSyncId, char *szErrorText, char *szBuffer )
{
    wsprintf(szBuffer,
        "<HTML><HEAD><TITLE>TPC-C Error</TITLE></HEAD><BODY>"
        "<FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">"
        "<INPUT TYPE=\"hidden\" NAME=\"STATUSID\" VALUE=\"%d\">"
        "<INPUT TYPE=\"hidden\" NAME=\"ERROR\" VALUE=\"%d\">"
        "<INPUT TYPE=\"hidden\" NAME=\"FORMID\" VALUE=\"%d\">"
        "<INPUT TYPE=\"hidden\" NAME=\"TERMID\" VALUE=\"%d\">"
        "<INPUT TYPE=\"hidden\" NAME=\"SYNID\" VALUE=\"%d\">"
        "<BOLD>An Error Occurred</BOLD><BR><BR>"
        "%s"
    );
}

```

```

        "<BR><BR><HR>"
        "<INPUT TYPE=\"submit\" NAME=\"CMD\" "
VALUE=\"..NewOrder..\">"
        "<INPUT TYPE=\"submit\" NAME=\"CMD\" "
VALUE=\"..Payment..\">"
        "<INPUT TYPE=\"submit\" NAME=\"CMD\" "
VALUE=\"..Delivery..\">"
        "<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Order-
Status..\">"
        "<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Stock-
Level..\">"
        "<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Exit..\">"
        "</FORM></BODY></HTML>"
        , iType, iErrorNum, MAIN_MENU_FORM, iTermId, iSyncId,
szErrorText );
    }

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

/* FUNCTION: MakeStockLevelForm
 *
 * PURPOSE: This function constructs the Stock Level HTML page.
 *
 * COMMENTS: The internal client buffer is created when the terminal
id is assigned and should not
            be freed except when the client terminal
id is no longer needed.
 */
void MakeStockLevelForm(int iTermId, STOCK_LEVEL_DATA *pStockLevelData,
BOOL bInput, char *szForm)
{
    int c;

    if( !bInput && (pStockLevelData->exec_status_code != eOK) ) {
        WEB_ERR(pStockLevelData->exec_status_code)
    }
}

```

```

        return;
    }

    c = wsprintf(szForm,
        "<HTML><HEAD><TITLE>TPC-C Stock Level</TITLE></HEAD><FORM"
        ACTION=\"tpcc.dll\" METHOD=\"GET\">\"
        "<INPUT TYPE=\"hidden\" NAME=\"STATUSID\" VALUE=\"0\">\"
        "<INPUT TYPE=\"hidden\" NAME=\"ERROR\" VALUE=\"0\">\"
        "<INPUT TYPE=\"hidden\" NAME=\"FORMID\" VALUE=\"%d\">\"
        "<INPUT TYPE=\"hidden\" NAME=\"TERMINID\" VALUE=\"%d\">\"
        "<INPUT TYPE=\"hidden\" NAME=\"SYNCID\" VALUE=\"%d\">\"
        "<PRE><font face=\"Courier\">\"
        Stock-Level<BR>\"
        "Warehouse: %4.4d District: %2.2d<BR> <BR>\",
        STOCK_LEVEL_FORM, iTermId,
        Term.pClientData[iTermId].iSyncId,
        Term.pClientData[iTermId].w_id,
        Term.pClientData[iTermId].d_id);

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

/* FUNCTION: MakeNewOrderForm
*

```

```

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

```

```

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

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

    if (!bInput){
        if(!(pNewOrderData->exec_status_code == eOK ||
        pNewOrderData->exec_status_code == eInvalidItem )){
            //transaction error
            WEB_ERR(pNewOrderData->exec_status_code)
            return;
        }
    }

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

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

    if ( bInput )
    {
        c += wsprintf(szForm+c, "Warehouse: %4.4d ",
            Term.pClientData[iTermId].w_id );

        strcpy( szForm+c,
            "District: <INPUT NAME=\"DID*\" SIZE=1>
            Date:<BR>\"
            "Customer: <INPUT NAME=\"CID*\" SIZE=4> Name:
            Credit: %Disc:<BR>\"
            "Order Number: Number of Lines:
            W_tax: D_tax:<BR> <BR>\"
            "Supp_W Item_Id Item Name Qty
            Stock B/G Price Amount<BR>\"
            "<INPUT NAME=\"SP00*\" SIZE=4> <INPUT
            NAME=\"IID00*\" SIZE=6> <INPUT
            NAME=\"Qty00*\" SIZE=1><BR>\"
            "<INPUT NAME=\"SP01*\" SIZE=4> <INPUT
            NAME=\"IID01*\" SIZE=6> <INPUT
            NAME=\"Qty01*\" SIZE=1><BR>\"

```

```

        " <INPUT NAME=\"SP02*\" SIZE=4> <INPUT
NAME=\"IID02*\" SIZE=6>
NAME=\"Qty02*\" SIZE=1><BR>"
        " <INPUT NAME=\"SP03*\" SIZE=4> <INPUT
NAME=\"IID03*\" SIZE=6>
NAME=\"Qty03*\" SIZE=1><BR>"
        " <INPUT NAME=\"SP04*\" SIZE=4> <INPUT
NAME=\"IID04*\" SIZE=6>
NAME=\"Qty04*\" SIZE=1><BR>"
        " <INPUT NAME=\"SP05*\" SIZE=4> <INPUT
NAME=\"IID05*\" SIZE=6>
NAME=\"Qty05*\" SIZE=1><BR>"
        " <INPUT NAME=\"SP06*\" SIZE=4> <INPUT
NAME=\"IID06*\" SIZE=6>
NAME=\"Qty06*\" SIZE=1><BR>"
        " <INPUT NAME=\"SP07*\" SIZE=4> <INPUT
NAME=\"IID07*\" SIZE=6>
NAME=\"Qty07*\" SIZE=1><BR>"
        " <INPUT NAME=\"SP08*\" SIZE=4> <INPUT
NAME=\"IID08*\" SIZE=6>
NAME=\"Qty08*\" SIZE=1><BR>"
        " <INPUT NAME=\"SP09*\" SIZE=4> <INPUT
NAME=\"IID09*\" SIZE=6>
NAME=\"Qty09*\" SIZE=1><BR>"
        " <INPUT NAME=\"SP10*\" SIZE=4> <INPUT
NAME=\"IID10*\" SIZE=6>
NAME=\"Qty10*\" SIZE=1><BR>"
        " <INPUT NAME=\"SP11*\" SIZE=4> <INPUT
NAME=\"IID11*\" SIZE=6>
NAME=\"Qty11*\" SIZE=1><BR>"
        " <INPUT NAME=\"SP12*\" SIZE=4> <INPUT
NAME=\"IID12*\" SIZE=6>
NAME=\"Qty12*\" SIZE=1><BR>"
        " <INPUT NAME=\"SP13*\" SIZE=4> <INPUT
NAME=\"IID13*\" SIZE=6>
NAME=\"Qty13*\" SIZE=1><BR>"
        " <INPUT NAME=\"SP14*\" SIZE=4> <INPUT
NAME=\"IID14*\" SIZE=6>
NAME=\"Qty14*\" SIZE=1><BR>"
        "Execution Status:
Total:<BR>"
        "</font></PRE><HR>"
        "<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"Process\">"
        "<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"Menu\">"
        "</FORM></HTML>"
    );
    }
    else
    {
        c += wsprintf(szForm+c, "Warehouse: %4.4d
District: %2.2d
Date: ",
pNewOrderData->w_id,
pNewOrderData->d_id);

        if ( bValid )
        {
            c += wsprintf(szForm+c, "%2.2d-%2.2d-
%4.4d %2.2d:%2.2d:%2.2d",
pNewOrderData->o_entry_d.day,
pNewOrderData->o_entry_d.month,
pNewOrderData->o_entry_d.year,

```

```

pNewOrderData->o_entry_d.hour,
pNewOrderData->o_entry_d.minute,
pNewOrderData->o_entry_d.second);
    }

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

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

        for(i=0; i<pNewOrderData->o_ol_cnt; i++)
        {
            c += sprintf(szForm+c,
" %4.4d %6.6d %-24s %2.2d %3.3d %1.1s $%6.2f $%7.2f
<BR>",
pNewOrderData->OL[i].ol_supply_w_id,
pNewOrderData->OL[i].ol_i_id,
pNewOrderData->OL[i].ol_i_name,
pNewOrderData->OL[i].ol_quantity,
pNewOrderData->OL[i].ol_stock,
pNewOrderData->OL[i].ol_brand_generic,
pNewOrderData->OL[i].ol_i_price,
pNewOrderData->OL[i].ol_amount );

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

        i = 0;
    }

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

    if ( bValid )
    {
        c += sprintf(szForm+c, "Execution Status:
Transaction committed. Total: $%8.2f ",
pNewOrderData->total_amount);
    }
    else

```



```

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

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

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

void MakePaymentForm(int iTermId, PAYMENT_DATA *pPaymentData, BOOL
bInput, char *szForm)
{
    int c;

    if(!bInput && (pPaymentData->exec_status_code != eOK ) ) {
        WEB_ERR(pPaymentData->exec_status_code)
        return;
    }
    c = sprintf(szForm,
        "<HTML><HEAD><TITLE>TPC-C Payment</TITLE></HEAD><BODY>"
        "<FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">"
        "<INPUT TYPE=\"hidden\" NAME=\"STATUSID\" VALUE=\"0\">"
        "<INPUT TYPE=\"hidden\" NAME=\"ERROR\" VALUE=\"0\">"
        "<INPUT TYPE=\"hidden\" NAME=\"FORMID\" VALUE=\"%d\">"
        "<INPUT TYPE=\"hidden\" NAME=\"TERMID\" VALUE=\"%d\">"
        "<INPUT TYPE=\"hidden\" NAME=\"SYNCID\" VALUE=\"%d\">"
        "<PRE><font face=\"Courier\">"
        "Payment<BR>"
        "Date: "
        , PAYMENT_FORM, iTermId,
        Term.pClientData[iTermId].iSyncId);

    if ( !bInput )
    {
        c += sprintf(szForm+c, "%2.2d-%2.2d-
%4.4d %2.2d:%2.2d:%2.2d",
            pPaymentData->h_date.day,
            pPaymentData->h_date.month,
            pPaymentData->h_date.year,
            pPaymentData->h_date.hour,
            pPaymentData->h_date.minute,
            pPaymentData->h_date.second);
    }
}

```

```

    }
    if ( bInput )
    {
        c += sprintf(szForm+c,
            "<BR> <BR>Warehouse: %4.4d
District: <INPUT
NAME=\"DID*\" SIZE=1><BR> <BR> <BR> <BR> <BR>"
            "Customer: <INPUT NAME=\"CID*\" SIZE=4>"
            "Cust-Warehouse: <INPUT NAME=\"CWI*\" SIZE=4> "
            "Cust-District: <INPUT NAME=\"CDI*\" SIZE=1><BR>"
            "Name: <INPUT NAME=\"CLT*\"
SIZE=16>
Since:<BR>"
            "
Credit:<BR>"
            "
Disc:<BR>"
            "
Phone:<BR> <BR>"
            "Amount Paid: $<INPUT NAME=\"HAM*\"
SIZE=7>
New Cust-Balance:<BR>"
            "Credit Limit:<BR> <BR>Cust-Data: <BR> <BR> <BR>
<BR> <BR></font></PRE><HR>"
            "<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"Process\"><INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"Menu\">"
            "</BODY></FORM></HTML>"
            , Term.pClientData[iTermId].w_id);
    }
    else
    {
        if(pPaymentData->c_id==0)
            c+=sprintf(szForm+c, "<BR> <BR> INVALID CUSTOMER
<BR><BR>");
        else if(pPaymentData->exec_status_code==eSqlError)
            c+=sprintf(szForm+c, "<BR> <BR> INVALID C_ID=%d OR
D_ID=%d OR W_ID=%d<BR><BR>", pPaymentData->c_id, pPaymentData-
>c_w_id, pPaymentData->c_d_id);
        else
        {
            c += sprintf(szForm+c,
                "<BR> <BR>Warehouse: %4.4d
District: %2.2d<BR>"
                "%-20s %-20s<BR>"
                "%-20s %-20s<BR>"
                "%-20s %-2s %5.5s-%4.4s %-20s %-2s %5.5s-
%4.4s<BR> <BR>"
                "Customer: %4.4d Cust-Warehouse: %4.4d Cust-
District: %2.2d<BR>"
                "Name: %-16s %-2s %-16s Since: %2.2d-%2.2d-
%4.4d<BR>"
                " %-20s Credit: %-
2s<BR>"
                , Term.pClientData[iTermId].w_id, pPaymentData-
>d_id, pPaymentData->w_street_1, pPaymentData-
>d_street_1, pPaymentData->w_street_2, pPaymentData-
>d_street_2, pPaymentData->w_city, pPaymentData->w_state,
pPaymentData->w_zip, pPaymentData->w_zip+5
            );
        }
    }
}

```

```

        , pPaymentData->d_city, pPaymentData->d_state,
pPaymentData->d_zip, pPaymentData->d_zip+5
        , pPaymentData->c_id, pPaymentData->c_w_id,
        pPaymentData->c_d_id
        , pPaymentData->c_first, pPaymentData->c_middle,
pPaymentData->c_last
        , pPaymentData->c_since.day, pPaymentData->
>c_since.month, pPaymentData->c_since.year
        , pPaymentData->c_street_1, pPaymentData->c_credit
        );

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

        c += wsprintf(szForm+c,
        "
        %-20s %-2s %5.5s-%4.4s
Phone: %6.6s-%3.3s-%3.3s-%4.4s<BR> <BR>",
        pPaymentData->c_city, pPaymentData->c_state,
pPaymentData->c_zip, pPaymentData->c_zip+5,
        pPaymentData->c_phone, pPaymentData->c_phone+6,
pPaymentData->c_phone+9, pPaymentData->c_phone+12 );

        c += sprintf(szForm+c,
        "Amount Paid:          $%7.2f      New Cust-
Balance: $%14.2f<BR>"
        , pPaymentData->h_amount, pPaymentData->c_balance
        , pPaymentData->c_credit_lim
        );

        if ( pPaymentData->c_credit[0] == 'B' && pPaymentData->
>c_credit[1] == 'C' )
        c += wsprintf(szForm+c,
        "Cust-Data: %-
50.50s<BR>          %-50.50s<BR>          %-
50.50s<BR>",
        pPaymentData->c_data, pPaymentData->
>c_data+50, pPaymentData->c_data+100, pPaymentData->c_data+150 );
        else
        strcpy(szForm+c, "Cust-Data: <BR> <BR> <BR>
<BR>");
        }
        strcat(szForm, " <BR></font></PRE><HR>"
        " <INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..NewOrder..\">"
        " <INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..Payment..\">"
        " <INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..Delivery..\">"
        " <INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..Order-Status..\">"
        " <INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..Stock-Level..\">"
        " <INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..Exit..\">"
        " </BODY></FORM></HTML>");
    }
}

```

```

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

void MakeOrderStatusForm(int iTermId, ORDER_STATUS_DATA
*pOrderStatusData, BOOL bInput, char *szForm)
{
    int i, c;
    static char szBR[] = " <BR> <BR> <BR> <BR> <BR> <BR> <BR> <BR>
<BR> <BR> <BR> <BR> <BR> <BR> <BR>";

    if(!bInput && (pOrderStatusData->exec_status_code != eOK ) ) {
        WEB_ERR(pOrderStatusData->exec_status_code)
        return;
    }

    c = wsprintf(szForm,
    "<HTML><HEAD><TITLE>TPC-C Order-
Status</TITLE></HEAD><BODY>"
    "<FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">"
    "<INPUT TYPE=\"hidden\" NAME=\"STATUSID\" VALUE=\"0\">"
    "<INPUT TYPE=\"hidden\" NAME=\"ERROR\" VALUE=\"0\">"
    "<INPUT TYPE=\"hidden\" NAME=\"FORMID\" VALUE=\"%d\">"
    "<INPUT TYPE=\"hidden\" NAME=\"TERMID\" VALUE=\"%d\">"
    "<INPUT TYPE=\"hidden\" NAME=\"SYNCD\" VALUE=\"%d\">"
    "<PRE><font face=\"Courier\">"
    Order-Status<BR>"
    "Warehouse: %4.4d ",
    ORDER_STATUS_FORM, iTermId,
    Term.pClientData[iTermId].iSyncId, Term.pClientData[iTermId].w_id);

    if ( bInput )
    {
        strcpy(szForm+c,
        "District: <INPUT NAME=\"DID*\" SIZE=1><BR>"
        "Customer: <INPUT NAME=\"CID*\" SIZE=4> Name:
<INPUT NAME=\"CLT*\" SIZE=23><BR>"
        "Cust-Balance:<BR> <BR>"
        "Order-Number:          Entry-Date:
Carrier-Number:<BR>"
        "Supply-W      Item-Id      Qty      Amount
Delivery-Date<BR> <BR> <BR> <BR> <BR> <BR> <BR> <BR> <BR> <BR> <BR>
<BR> <BR></font></PRE>"
        "<HR><INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"Process\"><INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"Menu\">"
        "</BODY></FORM></HTML>");
    }
    else
    {
        if (pOrderStatusData->o_ol_cnt == 0)
            c+=wsprintf(szForm+c, "<BR> <BR> NO SUCH ORDER
<BR><BR>");
        else if (pOrderStatusData->c_id == 0 && pOrderStatusData->
>c_last[0] == 0)
            c+=wsprintf(szForm+c, "<BR> <BR> INVALID CUSTOMER
<BR><BR>");
        else

```

```

{
    c += sprintf(szForm+c,
        "District: %2.2d<BR>"
        "Customer: %4.4d   Name: %-16s %-2s %-16s<BR>",
        pOrderStatusData->d_id, pOrderStatusData->c_id,
        pOrderStatusData->c_first, pOrderStatusData-
>c_middle, pOrderStatusData->c_last);

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

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

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

        strncpy( szForm+c, szBR, (15-i)*5 );
        c += (15-i)*5;
        strcpy(szForm+c,
            "</font></PRE><HR><INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..NewOrder..\">"
            "<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"..Payment..\">"

```

```

        "<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"..Delivery..\">"
        "<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"..Order-Status..\">"
        "<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"..Stock-Level..\">"
        "<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"..Exit..\">"
        "</BODY></FORM></HTML>" );
    }

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

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

    c = sprintf(szForm,
        "<HTML><HEAD><TITLE>TPC-C Delivery</TITLE></HEAD><BODY>"
        "<FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">"
        "<INPUT TYPE=\"hidden\" NAME=\"STATUSID\" VALUE=\"%d\">"
        "<INPUT TYPE=\"hidden\" NAME=\"ERROR\" VALUE=\"%0\">"
        "<INPUT TYPE=\"hidden\" NAME=\"FORMID\" VALUE=\"%d\">"
        "<INPUT TYPE=\"hidden\" NAME=\"TERMID\" VALUE=\"%d\">"
        "<INPUT TYPE=\"hidden\" NAME=\"SYNCD\" VALUE=\"%d\">"
        "<PRE><font face=\"Courier\">"
Delivery<BR>"
        "Warehouse: %4.4d<BR> <BR>",
        (!bInput && (pDeliveryData->exec_status_code != eOK)) ?
ERR_TYPE_DELIVERY_POST : 0,
        DELIVERY_FORM, iTermId, Term.pClientData[iTermId].iSyncId,
        Term.pClientData[iTermId].w_id);

    if ( bInput )
    {
        strcpy( szForm+c,
            "Carrier Number: <INPUT NAME=\"OCD*\" SIZE=1><BR>"
<BR>"
            "Execution Status: <BR> <BR> <BR> <BR> <BR> <BR>
<BR> <BR>"
            " <BR> <BR> <BR> <BR> <BR> <BR> <BR>
</font></PRE><HR>"
            "<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"Process\">"
            "<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"Menu\">"
            "</BODY></FORM></HTML>" );
    }
    else
    {
        sprintf( szForm+c,
            "Carrier Number: %2.2d<BR> <BR>"
            "Execution Status: %s <BR> <BR> <BR> <BR> <BR>
<BR> <BR> <BR>"

```

```

                " <BR> <BR> <BR> <BR> <BR> <BR> <BR> <BR>
</font></PRE>"
                "<HR><INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"..NewOrder..\">"
                "<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"..Payment..\">"
                "<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"..Delivery..\">"
                "<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"..Order-Status..\">"
                "<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"..Stock-Level..\">"
                "<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"..Exit..\">"
                "</BODY></FORM></HTML>"
                pDeliveryData->o_carrier_id,
                (pDeliveryData->exec_status_code == eOK) ?
"Delivery has been queued." : "Delivery Post Failed "
            );
        }
}

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

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

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

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

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

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

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

```

```

 *           int
iTermId client browser terminal id
 */

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

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

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

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

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

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

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

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

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

/* FUNCTION: ProcessDeliveryForm
 *
 * PURPOSE: This function gets and validates the input data from the
delivery form
 *           filling in the required input variables. It then
calls the PostDeliveryInfo

```

```

*           Api, The client is then informed that the
transaction has been posted.
*
* ARGUMENTS: EXTENSION_CONTROL_BLOCK      *pECB  passed in structure
pointer from inetsrv.
*           int
iTermId client browser terminal id
*
*/

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

    PDELIVERY_DATA pDelivery;

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

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

    if (dwNumDeliveryThreads)
    {
        //post delivery info
        if ( PostDeliveryInfo(pDelivery->w_id, pDelivery-
>o_carrier_id) )
            pDelivery->exec_status_code = eDeliveryFailed;
        else
            pDelivery->exec_status_code = eOK;
    }
    else // delivery is done synchronously if no delivery threads
configured
        Term.pClientData[iTermId].pTxn->Delivery();

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

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

void ProcessStockLevelForm(EXTENSION_CONTROL_BLOCK *pECB, int iTermId,
char *szBuffer)

```

```

{
    char      *ptr = pECB->lpszQueryString;

    PSTOCK_LEVEL_DATA pStockLevel;

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

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

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

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

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

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

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

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

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

```

```

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

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

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

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

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

        pNewOrderData->o_ol_cnt = items;
    }

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

```

```

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

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

    GetKeyValue(&ptr, "CID*", szTmp, sizeof(szTmp),
ERR_PAYMENT_MISSING_CID_KEY);
    if ( szTmp[0] == 0 )
    {
        bCustIdBlank = TRUE;
        pPaymentData->c_id = 0;
    }
    else
    {
        // parse customer id and verify that last name was NOT
entered
        bCustIdBlank = FALSE;
        if ( !IsNumeric(szTmp) )
            throw new
CWEBCLNT_ERR( ERR_PAYMENT_CUSTOMER_INVALID );
        pPaymentData->c_id = atoi(szTmp);
    }

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

    if( ( pPaymentData->c_w_id == 0 ) || ( pPaymentData->c_d_id
==0 ) )
        throw new CWEBCLNT_ERR( ERR_PAYMENT_CUSTOMER_INVALID );

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

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

    GetKeyValue(&ptr, "HAM*", szTmp, sizeof(szTmp),
ERR_PAYMENT_MISSING_HAM_KEY);
    if (!IsDecimal(szTmp))
        throw new CWEBCLNT_ERR( ERR_PAYMENT_HAM_INVALID );
    pPaymentData->h_amount = atof(szTmp);
}

```

```

    if ( pPaymentData->h_amount >= 10000.00 || pPaymentData->h_amount
< 0 )
        throw new CWEBCLNT_ERR( ERR_PAYMENT_HAM_RANGE );
}

/* FUNCTION: GetOrderStatusData
 *
 * PURPOSE: This function extracts and validates the payment form
data from an http command string.
 */
void GetOrderStatusData(LPSTR lpszQueryString, ORDER_STATUS_DATA
*pOrderStatusData)
{
    char    szTmp[26];
    char    *ptr = lpszQueryString;

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

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

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

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

```

```

 *
 *           TRUE    if string contains
only numeric characters i.e. '0' - '9'
 */

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

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

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

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

    if ( *ptr == 0 )
        return FALSE;

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

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

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

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

```

tpcc.def

LIBRARY TPCC.DLL

EXPORTS

GetExtensionVersion @1
HttpExtensionProc @2
TerminateExtension @3

tpcc.h

```
/* FILE: TPCC.H
 * Microsoft TPC-C Kit Ver. 4.20.000
 * Copyright Microsoft, 1999
 * All Rights Reserved
 *
 * Version 4.10.000 audited by Richard Gimarc,
 * Performance Metrics, 3/17/99
 *
 * PURPOSE: Header file for ISAPI TPCC.DLL, defines structures
 * and functions used in the isapi tpcc.dll.
 */

//VERSION RESOURCE DEFINES

#define _APS_NEXT_RESOURCE_VALUE 101
#define _APS_NEXT_COMMAND_VALUE 40001
#define _APS_NEXT_CONTROL_VALUE 1000
#define _APS_NEXT_SYMED_VALUE 101

#define TP_MAX_RETRIES 50

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

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

//This structure defines the data necessary to keep distinct for each
//terminal or client connection.
typedef struct _CLIENTDATA
{
    int iNextFree;
    //index of next free element or -1 if this entry in use.
```

```
int w_id;
//warehouse id assigned at welcome form
int d_id;
//district id assigned at welcome form

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

CTPCC_BASE *pTxn;
} CLIENTDATA, *PCLIENTDATA;

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

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

enum WEBERROR
{
    NO_ERR,
    ERR_COMMAND_UNDEFINED,
    ERR_D_ID_INVALID,
    ERR_DELIVERY_CARRIER_ID_RANGE,
    ERR_DELIVERY_CARRIER_INVALID,
    ERR_DELIVERY_MISSING_OCD_KEY,
    ERR_DELIVERY_THREAD_FAILED,
    ERR_GETPROCADDR_FAILED,
    ERR_HTML_ILL_FORMED,
    ERR_INVALID_SYNC_CONNECTION,
    ERR_INVALID_TERMID,
    ERR_LOADDLL_FAILED,
    ERR_MAX_CONNECTIONS_EXCEEDED,
    ERR_MEM_ALLOC_FAILED,
    ERR_MISSING_REGISTRY_ENTRIES,
    ERR_NEWORDER_CUSTOMER_INVALID,
    ERR_NEWORDER_CUSTOMER_KEY,
    ERR_NEWORDER_DISTRICT_INVALID,
    ERR_NEWORDER_FORM_MISSING_DID,
    ERR_NEWORDER_ITEMID_INVALID,
    ERR_NEWORDER_ITEMID_RANGE,
    ERR_NEWORDER_ITEMID_WITHOUT_SUPPW,
    ERR_NEWORDER_MISSING_IID_KEY,
    ERR_NEWORDER_MISSING_QTY_KEY,
    ERR_NEWORDER_MISSING_SUPPW_KEY,
    ERR_NEWORDER_NOITEMS_ENTERED,
    ERR_NEWORDER_QTY_INVALID,
    ERR_NEWORDER_QTY_RANGE,
```



```

ERR_NEWORDER_QTY_WITHOUT_SUPPW,
ERR_NEWORDER_SUPPW_INVALID,
ERR_NO_SERVER_SPECIFIED,
ERR_ORDERSTATUS_CID_AND_CLT,
ERR_ORDERSTATUS_CID_INVALID,
ERR_ORDERSTATUS_CLT_RANGE,
ERR_ORDERSTATUS_DID_INVALID,
ERR_ORDERSTATUS_MISSING_CID_CLT,
ERR_ORDERSTATUS_MISSING_CID_KEY,
ERR_ORDERSTATUS_MISSING_CLT_KEY,
ERR_ORDERSTATUS_MISSING_DID_KEY,
ERR_PAYMENT_CDI_INVALID,
ERR_PAYMENT_CID_AND_CLT,
ERR_PAYMENT_CUSTOMER_INVALID,
ERR_PAYMENT_CWI_INVALID,
ERR_PAYMENT_DISTRICT_INVALID,
ERR_PAYMENT_HAM_INVALID,
ERR_PAYMENT_HAM_RANGE,
ERR_PAYMENT_LAST_NAME_TOO_LONG,
ERR_PAYMENT_MISSING_CDI_KEY,
ERR_PAYMENT_MISSING_CID_CLT,
ERR_PAYMENT_MISSING_CID_KEY,
ERR_PAYMENT_MISSING_CLT,
ERR_PAYMENT_MISSING_CLT_KEY,
ERR_PAYMENT_MISSING_CWI_KEY,
ERR_PAYMENT_MISSING_DID_KEY,
ERR_PAYMENT_MISSING_HAM_KEY,
ERR_STOCKLEVEL_MISSING_THRESHOLD_KEY,
ERR_STOCKLEVEL_THRESHOLD_INVALID,
ERR_STOCKLEVEL_THRESHOLD_RANGE,
ERR_VERSION_MISMATCH,
ERR_W_ID_INVALID
};

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

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

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

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

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

};

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

//function prototypes

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

```

```

void ProcessDeliveryForm(EXTENSION_CONTROL_BLOCK *pECB, int iTermId,
char *szBuffer);
void ProcessStockLevelForm(EXTENSION_CONTROL_BLOCK *pECB, int iTermId,
char *szBuffer);
void GetNewOrderData(LPSTR lpszQueryString, NEW_ORDER_DATA
*pNewOrderData);
void GetPaymentData(LPSTR lpszQueryString, PAYMENT_DATA *pPaymentData);
void GetOrderStatusData(LPSTR lpszQueryString, ORDER_STATUS_DATA
*pOrderStatusData);
BOOL PostDeliveryInfo(short w_id, short o_carrier_id);
BOOL IsNumeric(char *ptr);
BOOL IsDecimal(char *ptr);
void DeliveryWorkerThread(void *ptr);

```

tpcc.rc

```

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

```

```

#define APSTUDIO_READONLY_SYMBOLS
////////////////////////////////////
//
// Generated from the TEXTINCLUDE 2 resource.
//
#include "afxres.h"
////////////////////////////////////
//
#undef APSTUDIO_READONLY_SYMBOLS
////////////////////////////////////
//
// English (U.S.) resources
//
#if !defined(AFX_RESOURCE_DLL) || defined(AFX_TARG_ENU)
#ifdef _WIN32
LANGUAGE LANG_ENGLISH, SUBLANG_ENGLISH_US
#pragma code_page(1252)
#endif // _WIN32

#ifdef _MAC
////////////////////////////////////
//
// Version
//

```

```

VS_VERSION_INFO VERSIONINFO
FILEVERSION 0,4,0,0
PRODUCTVERSION 0,4,0,0
FILEFLAGSMASK 0x3fL
#ifdef _DEBUG
FILEFLAGS 0x1L
#else
FILEFLAGS 0x0L
#endif
FILEOS 0x40004L
FILETYPE 0x2L
FILESUBTYPE 0x0L
BEGIN
    BLOCK "StringFileInfo"

```

```

BEGIN
    BLOCK "040904b0"
    BEGIN
        VALUE "Comments", "TPC-C HTML DLL Server (DBLIB)\0"
        VALUE "CompanyName", "Microsoft\0"
        VALUE "FileDescription", "TPC-C HTML DLL Server (DBLIB)\0"
        VALUE "FileVersion", "0, 4, 0, 0\0"
        VALUE "InternalName", "tpcc\0"
        VALUE "LegalCopyright", "Copyright 1997\0"
        VALUE "OriginalFilename", "tpcc.dll\0"
        VALUE "ProductName", "Microsoft tpcc\0"
        VALUE "ProductVersion", "0, 4, 0, 0\0"
    END
END
BLOCK "VarFileInfo"
BEGIN
    VALUE "Translation", 0x409, 1200
END
#endif // !_MAC

#ifdef APSTUDIO_INVOKED
////////////////////////////////////
//
// TEXTINCLUDE
//
1 TEXTINCLUDE DISCARDABLE
BEGIN
    "resource.h\0"
END

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

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

#endif // APSTUDIO_INVOKED

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

```

```

END

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

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

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

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

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

```

txnlog.h

```

#include <fstream.h>

#define LOG_START "Start of Delivery Log"
#define LOG_END "End of Delivery Log"

#define TXN_EVENT_START 2
#define TXN_EVENT_STOP 4

typedef struct _TXN_RECORD_TPCC_DELIV_DEF
{
    __int64 TxnStartT0; // start of txn
    int DeltaT4; // response
    time (ms) DeltaTxnExec; // execution time
    (ms)
    WORD w_id; // warehouse ID
    BYTE TxnStatus; // completion status
    for txn to indicate errors
    short o_carrier_id; // carrier id
}

```

```

    long o_id[10]; // returned delivery
    transaction ids
} TXN_RECORD_TPCC_DELIV_DEF, *PTXN_RECORD_TPCC_DELIV_DEF;

typedef struct _TXN_LOG_HEADER
{
    char Start[sizeof(LOG_START)]; //
    begin string
    char Host[MAX_COMPUTERNAME_LENGTH+1]; // which
    computer this log is from
    __int64 BeginLog;
    __int64 // timestamp of log start
    __int64 BeginCTLog;
    __int64 // clock ticks in ms
    __int64 iRecCount;
    __int64 // number of records in log file
    __int64 iFileSize;
    // file size in bytes
} TXN_LOG_HEADER;

class CTxnLog
{
private:
    __int64 BeginLog; // timestamp of log
    start __int64 BeginCTLog; // clock ticks in ms
    int iRecCount; // number of
    records in log file
    int iFileSize; // size of
    the log so far
    TXN_LOG_HEADER lh; // log header
    ofstream log; // log file
    CRITICAL_SECTION LogCriticalSection;

public:
    CTxnLog(LPCTSTR szFileName);
    ~CTxnLog(void);

    int WriteToLog(PTXN_RECORD_TPCC_DELIV_DEF pTxnRcd);
    int WriteCtrlRecToLog(BYTE SubType, LPTSTR lpStr, DWORD dwLen);
};

CTxnLog::CTxnLog(LPCTSTR szFileName){
    log.open(szFileName,ios::out|ios::binary);
    memset(&lh, sizeof(TXN_LOG_HEADER),0);
    memcpy(&lh.Start,LOG_START,sizeof(LOG_START));
    InitializeCriticalSection(&LogCriticalSection);
}

CTxnLog::WriteCtrlRecToLog(BYTE type, LPTSTR lpStr, DWORD dwLen){
    if (type == TXN_EVENT_START ){
        strcpy(lh.Host,lpStr);
        iFileSize=sizeof(TXN_LOG_HEADER);
        BeginLog=time(NULL); //get start date time
        BeginCTLog=GetCurrentTime(); //get start clock ticks
        iRecCount=0;
        log.write((char*)&lh,sizeof(TXN_LOG_HEADER));
        return 0;
    }else if( type == TXN_EVENT_STOP) {

```

```

        lh.BeginLog=BeginLog;
        lh.BeginCTLog=BeginCTLog;
        lh.iFileSize=iFileSize;
        lh.iRecCount=iRecCount;
        log.seekp(0); //go to the front of the file
        log.write((char*)&lh,sizeof(TXN_LOG_HEADER));
        log.flush(); //flush any outstanding ios
        return 0;
    }
    return 1;
}

CTxnLog::~CTxnLog(){
    log.close();
}

CTxnLog::WriteToLog(PTXN_RECORD_TPCC_DELIV_DEF pTxnRcrd){
    EnterCriticalSection(&LogCriticalSection);

    log.write((char *)pTxnRcrd,sizeof(TXN_RECORD_TPCC_DELIV_DEF));

    //log accounting
    iFileSize+=sizeof(TXN_RECORD_TPCC_DELIV_DEF);
    ++iRecCount;

    LeaveCriticalSection(&LogCriticalSection);
    return 0;
}

```

tpcc_com.cpp

```

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

// needed for CoInitializeEx
#define _WIN32_WINNT 0x0400

#include <windows.h>

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

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

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

// wrapper routine for class constructor

```

```

__declspec( dllexport ) CTPCC_COM* CTPCC_COM_new(BOOL bSinglePool)
{
    return new CTPCC_COM(bSinglePool);
}

CTPCC_COM::CTPCC_COM(BOOL bSinglePool)
{
    HRESULT hr = NULL;
    long lRet = 0;
    ULONG ulTmpSize = 0;

    m_pTxn = NULL;
    m_pNewOrder = NULL;
    m_pPayment = NULL;
    m_pStockLevel = NULL;
    m_pOrderStatus = NULL;

    m_bSinglePool = bSinglePool;

    ulTmpSize = (ULONG) sizeof(COM_DATA);
    VariantInit(&m_vTxn);
    m_vTxn.vt = VT_SAFEARRAY;

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

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

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

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

        // all txns will use same component
        m_pPayment = m_pNewOrder;
        m_pStockLevel = m_pNewOrder;
        m_pOrderStatus = m_pNewOrder;
    }
    else
    {
        // use different components for each txn

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

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

```

```

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

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

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

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

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

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

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

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

void CTPCC_COM::NewOrder()
{
    VARIANT vTxn_out;

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

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

```

```

void CTPCC_COM::Payment()
{
    VARIANT vTxn_out;

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

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

void CTPCC_COM::StockLevel()
{
    VARIANT vTxn_out;

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

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

void CTPCC_COM::OrderStatus()
{
    VARIANT vTxn_out;

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

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

```

tpcc_com.h

```

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

#pragma once
#include <stdio.h>

```

```

#include "..\..\tpcc_com_ps\src\tpcc_com_ps.h"

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

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

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

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

    int m_hr;
    int m_iErrorType;
    int m_iError;

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

    int ErrorNum() {return m_hr;}

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

class DllDecl CTPCC_COM : public CTPCC_BASE
{

```

```

private:
    BOOL m_bSinglePool;

    // COM Interface pointers
    ITPCC* m_pNewOrder;
    ITPCC* m_pPayment;
    ITPCC* m_pStockLevel;
    ITPCC* m_pOrderStatus;

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

    VARIANT m_vTxn;

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

    inline PNEW_ORDER_DATA BuffAddr_NewOrder()
    { return &m_pTxn->u.NewOrder; };
    inline PPAYMENT_DATA BuffAddr_Payment()
    { return &m_pTxn->u.Payment; };
    inline PDELIVERY_DATA BuffAddr_Delivery()
    { return &m_pTxn->u.Delivery; };
    inline PSTOCK_LEVEL_DATA BuffAddr_StockLevel()
    { return &m_pTxn->u.StockLevel; };
    inline PORDER_STATUS_DATA BuffAddr_OrderStatus()
    { return &m_pTxn->u.OrderStatus; };

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

inline void ReleaseInterface(IUnknown *pUnk)
{
    if (pUnk)
    {
        pUnk->Release();
        pUnk = NULL;
    }
}

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

typedef CTPCC_COM* (TYPE_CTPCC_COM)(BOOL);

```

Methods.h

```

/* FILE: METHODS.H
* Microsoft TPC-C Kit Ver. 4.20.000
* Copyright Microsoft, 1999
* All Rights Reserved
*
* not yet audited
*
* PURPOSE: Header file for COM components.
*
* Change history:
* 4.20.000 - first version
*/
enum COMPONENT_ERROR
{
    ERR_MISSING_REGISTRY_ENTRIES = 1,
    ERR_LOADDLL_FAILED,
    ERR_GETPROCADDR_FAILED,
    ERR_UNKNOWN_DB_PROTOCOL
};
class CCOMPONENT_ERR : public CBaseErr
{
public:
    CCOMPONENT_ERR(COMPONENT_ERROR Err)
    {
        m_Error = Err;
        m_szTextDetail = NULL;
        m_SystemErr = 0;
        m_szErrorText = NULL;
    };
    CCOMPONENT_ERR(COMPONENT_ERROR Err, char
        *szTextDetail, DWORD dwSystemErr)
    {
        m_Error = Err;
        m_szTextDetail = new
            char[strlen(szTextDetail)+1];
        strcpy( m_szTextDetail, szTextDetail );
        m_SystemErr = dwSystemErr;
        m_szErrorText = NULL;
    };
    ~CCOMPONENT_ERR()
    {
        if (m_szTextDetail != NULL)
            delete [] m_szTextDetail;
        if (m_szErrorText != NULL)
            delete [] m_szErrorText;
    };
    COMPONENT_ERROR m_Error;
    char *m_szTextDetail;
    char *m_szErrorText;
    DWORD m_SystemErr;
    int ErrorType() {return ERR_TYPE_COMPONENT;};
    int ErrorNum() {return m_Error;};
    char *ErrorText();
};
static void WriteMessageToEventLog(LPTSTR lpszMsg);
// CTPCC_Common
class CTPCC_Common :
public ITPCC,

```

```

public IObjectControl,
public IObjectConstruct,
public CComObjectRootEx<CComSingleThreadModel>
{
public:
    BEGIN_COM_MAP(CTPCC_Common)
        COM_INTERFACE_ENTRY(ITPCC)
        COM_INTERFACE_ENTRY(IObjectControl)
        COM_INTERFACE_ENTRY(IObjectConstruct)
    END_COM_MAP()
    CTPCC_Common();
    ~CTPCC_Common();
    // ITPCC
public:
    HRESULT __stdcall NewOrder( VARIANT txn_in,
        VARIANT* txn_out);
    HRESULT __stdcall Payment( VARIANT txn_in,
        VARIANT* txn_out);
    HRESULT __stdcall Delivery( VARIANT txn_in,
        VARIANT* txn_out) {return E_NOTIMPL;};
    HRESULT __stdcall StockLevel( VARIANT txn_in, VARIANT*
        txn_out);
    HRESULT __stdcall OrderStatus( VARIANT txn_in,
        VARIANT* txn_out);
    HRESULT __stdcall CallSetComplete();
    // IObjectControl
    STDMETHODCALLTYPE CanBePooled() { return m_bCanBePooled;
    };
    STDMETHODCALLTYPE Activate() { return S_OK; } // we don't support COM
Services transactions (no enlistment)
    STDMETHODCALLTYPE Deactivate() { /* nothing to do */ }
    // IObjectConstruct
    STDMETHODCALLTYPE Construct(IDispatch * pUnk);
    // helper methods
private:
    BOOL m_bCanBePooled;
    CTPCC_BASE *m_pTxn;
    struct COM_DATA
    {
        int retval;
        int error;
        union
        {
            NEW_ORDER_DATA NewOrder;
            PAYMENT_DATA Payment;
            DELIVERY_DATA Delivery;
            STOCK_LEVEL_DATA StockLevel;
            ORDER_STATUS_DATA OrderStatus;
        } u;
    };
};
// CTPCC
class CTPCC :
public CTPCC_Common,
public CComCoClass<CTPCC, &CLSID_TPCC>
{
public:
    DECLARE_REGISTRY_RESOURCEID(IDR_TPCC)
    BEGIN_COM_MAP(CTPCC)
        COM_INTERFACE_ENTRY2(IUnknown, CComObjectRootEx)
        COM_INTERFACE_ENTRY_CHAIN(CTPCC_Common)
    };
};

```

```

        END_COM_MAP()
};
// CNewOrder
class CNewOrder :
public CTPCC_Common,
public CComCoClass<CNewOrder, &CLSID_NewOrder>
{
public:
    DECLARE_REGISTRY_RESOURCEID(IDR_NEWORDER)
    BEGIN_COM_MAP(CNewOrder)
    COM_INTERFACE_ENTRY2(IUnknown, CComObjectRootEx)
    COM_INTERFACE_ENTRY_CHAIN(CTPCC_Common)
    END_COM_MAP()
    // ITPCC

public:
    // HRESULT __stdcall NewOrder( VARIANT txn_in,VARIANT* txn_out)
    {return E_NOTIMPL;}
    HRESULT __stdcall Payment( VARIANT txn_in,
        VARIANT* txn_out) {return E_NOTIMPL;}
    HRESULT __stdcall StockLevel( VARIANT txn_in, VARIANT*
        txn_out) {return E_NOTIMPL;}
    HRESULT __stdcall OrderStatus( VARIANT txn_in,
        VARIANT* txn_out) {return E_NOTIMPL;}
};
// COrderStatus
class COrderStatus :
public CTPCC_Common,
public CComCoClass<COrderStatus, &CLSID_OrderStatus>
{
public:
    DECLARE_REGISTRY_RESOURCEID(IDR_ORDERSTATUS)
    BEGIN_COM_MAP(COrderStatus)
    COM_INTERFACE_ENTRY2(IUnknown, CComObjectRootEx)
    COM_INTERFACE_ENTRY_CHAIN(CTPCC_Common)
    END_COM_MAP()
    // ITPCC

public:
    HRESULT __stdcall NewOrder( VARIANT txn_in,
        VARIANT* txn_out) {return E_NOTIMPL;}
    HRESULT __stdcall Payment( VARIANT txn_in,
        VARIANT* txn_out) {return E_NOTIMPL;}
    HRESULT __stdcall StockLevel( VARIANT txn_in, VARIANT*
        txn_out) {return E_NOTIMPL;}
    // HRESULT __stdcall OrderStatus( VARIANT txn_in,VARIANT*
        txn_out) {return E_NOTIMPL;}
};
// CPayment
class CPayment :
public CTPCC_Common,
public CComCoClass<CPayment, &CLSID_Payment>
{
public:
    DECLARE_REGISTRY_RESOURCEID(IDR_PAYMENT)
    BEGIN_COM_MAP(CPayment)
    COM_INTERFACE_ENTRY2(IUnknown, CComObjectRootEx)
    COM_INTERFACE_ENTRY_CHAIN(CTPCC_Common)
    END_COM_MAP()

```

```

        // ITPCC
public:
    HRESULT __stdcall NewOrder( VARIANT txn_in,
        VARIANT* txn_out) {return E_NOTIMPL;}
    // HRESULT __stdcall Payment( VARIANT txn_in,VARIANT* txn_out)
    {return E_NOTIMPL;}
    HRESULT __stdcall StockLevel( VARIANT txn_in, VARIANT*
        txn_out) {return E_NOTIMPL;}
    HRESULT __stdcall OrderStatus( VARIANT txn_in,
        VARIANT* txn_out) {return E_NOTIMPL;}
};
// CStockLevel
class CStockLevel :
public CTPCC_Common,
public CComCoClass<CStockLevel, &CLSID_StockLevel>
{
public:
    DECLARE_REGISTRY_RESOURCEID(IDR_STOCKLEVEL)
    BEGIN_COM_MAP(CStockLevel)
    COM_INTERFACE_ENTRY2(IUnknown, CComObjectRootEx)
    COM_INTERFACE_ENTRY_CHAIN(CTPCC_Common)
    END_COM_MAP()
    // ITPCC

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

```

resource.h

```

//{{NO_DEPENDENCIES}}
// Microsoft Developer Studio generated include file.
// Used by tpcc_com_all.rc
//
#define PROJNAME 1
#define IDS_PROJNAME 1
#define IDR_NEWORDER 102
#define IDR_ORDERSTATUS 103
#define IDR_PAYMENT 104
#define IDR_STOCKLEVEL 105
#define IDR_TPCC 109

// Next default values for new objects
//
#ifdef APSTUDIO_INVOKED
#ifdef APSTUDIO_READONLY_SYMBOLS
#define _APS_NEXT_RESOURCE_VALUE 115
#define _APS_NEXT_COMMAND_VALUE 40001
#define _APS_NEXT_CONTROL_VALUE 1000
#define _APS_NEXT_SYMED_VALUE 101
#endif
#endif

```

tpcc_com_all.cpp


```

/* FILE: TPCC_COM_ALL.CPP
* Microsoft TPC-C Kit Ver. 4.20.000
* Copyright Microsoft, 1999
* All Rights Reserved
*
* Version 4.10.000 audited by Richard Gimarc,
Performance Metrics, 3/17/99
*
* PURPOSE: Implementation for TPC-C Tuxedo class.
* Contact: Charles Levine (clevine@microsoft.com)
*
* Change history:
* 4.20.000 - updated rev number to match kit
*/
#define STRICT
#define _WIN32_WINNT 0x0400
#define _ATL_APARTMENT_THREADED
#include <stdio.h>
#include <atlbase.h>
//You may derive a class from CComModule and use it if you want to
override
//something, but do not change the name of _Module
extern CComModule _Module;
#include <atlcom.h>
#include <initguid.h>
#include <transact.h>
#include <atlimpl.cpp>
#include <comsvcs.h>
#include <sqltypes.h>
#include <sql.h>
#include <sqlext.h>
#include "tpcc_com_ps.h"
#include "..\..\common\src\trans.h" //tpckit //transaction header
contains definitions of structures specific to TPC-C
#include "..\..\common\src\txn_base.h"
#include "..\..\common\src\error.h"
#include "..\..\common\src\ReadRegistry.h"
#include "..\..\db_dblib_dll\src\tpcc_dblib.h" // DBLIB implementation
of TPC-C txns
#include "resource.h"
#include "tpcc_com_all.h"
#include "tpcc_com_all_i.c"
#include "Methods.h"
#include "..\..\tpcc_com_ps\src\tpcc_com_ps_i.c"
#include "..\..\common\src\ReadRegistry.cpp"
CComModule _Module;
//CAtlComModule _Module;
BEGIN_OBJECT_MAP(ObjectMap)
OBJECT_ENTRY(CLSID_TPCC, CTPCC)
OBJECT_ENTRY(CLSID_NewOrder, CNewOrder)
OBJECT_ENTRY(CLSID_OrderStatus, COrderStatus)
OBJECT_ENTRY(CLSID_Payment, CPayment)
OBJECT_ENTRY(CLSID_StockLevel, CStockLevel)
END_OBJECT_MAP()
// configuration settings from registry
TPCCREGISTRYDATA Reg;
char szMyComputerName[MAX_COMPUTERNAME_LENGTH+1];
static HINSTANCE hLibInstanceDb = NULL;
TYPE_CTPCC_DBLIB *pCTPCC_DBLIB_new;
////////////////////////////////////
//
// DLL Entry Point
extern "C"

```

```

BOOL WINAPI DllMain(HINSTANCE hInstance, DWORD dwReason, LPVOID
/*lpReserved*/)
{
    char szDllName[128];
    //DebugBreak();
    try
    {
        if (dwReason == DLL_PROCESS_ATTACH)
        {
            _Module.Init(ObjectMap, hInstance);
            DisableThreadLibraryCalls(hInstance);
            DWORD dwSize = MAX_COMPUTERNAME_LENGTH+1;
            GetComputerName(szMyComputerName, &dwSize);
            szMyComputerName[dwSize] = 0;
            if ( ReadTPCCRegistrySettings( &Reg ) )
                throw new CCOMPONENT_ERR(
                    ERR_MISSING_REGISTRY_ENTRIES );
            if (Reg.eDB_Protocol == DBLIB)
            {
                strcpy( szDllName, Reg.szPath );
                strcat( szDllName, "tpcc_dblib.dll");
                hLibInstanceDb = LoadLibrary( szDllName );
                if (hLibInstanceDb == NULL)
                    throw new CCOMPONENT_ERR(
                        ERR_LOADDLL_FAILED, szDllName,
                        GetLastError() );
                // get function pointer to wrapper for
                class constructor
                pCTPCC_DBLIB_new = (TYPE_CTPCC_DBLIB*)
                    GetProcAddress(hLibInstanceDb, "CTPCC_DBLIB_new");
                if (pCTPCC_DBLIB_new == NULL)
                    throw new CCOMPONENT_ERR(
                        ERR_GETPROCADDR_FAILED, szDllName,
                        GetLastError() );
            }
            else
                throw new
                CCOMPONENT_ERR( ERR_UNKNOWN_DB_PROTOCOL
                );
            else if (dwReason == DLL_PROCESS_DETACH)
                _Module.Term();
        }
        catch (CBaseErr *e)
        {
            WriteMessageToEventLog(e->ErrorText());
            delete e;
            return FALSE;
        }
        catch (...)
        {
            WriteMessageToEventLog(TEXT("Unhandled exception in
            object DllMain"));
            return FALSE;
        }
        return TRUE; // OK
    }
    //////////////////////////////////////
    //
    // Used to determine whether the DLL can be unloaded by OLE
    STDAPI DllCanUnloadNow(void)
    {

```

```

        return (_Module.GetLockCount()==0) ? S_OK : S_FALSE;
    }
    //////////////////////////////////////////
    // Returns a class factory to create an object of the requested type
    STDAPI DllGetClassObject(REFCLSID rclsid, REFIID riid, LPVOID* ppv)
    {
        return _Module.GetClassObject(rclsid, riid, ppv);
    }
    //////////////////////////////////////////
    // DllRegisterServer - Adds entries to the system registry
    STDAPI DllRegisterServer(void)
    {
        // registers object, typelib and all interfaces in typelib
        return _Module.RegisterServer(TRUE);
    }
    //////////////////////////////////////////
    // DllUnregisterServer - Removes entries from the system registry
    STDAPI DllUnregisterServer(void)
    {
        _Module.UnregisterServer();
        return S_OK;
    }

    static void WriteMessageToEventLog(LPTSTR lpszMsg)
    {
        TCHAR szMsg[256];
        HANDLE hEventSource;
        LPTSTR lpszStrings[2];
        // Use event logging to log the error.
        //
        hEventSource = RegisterEventSource(NULL,
        TEXT("tpcc_com_all.dll"));
        _stprintf(szMsg, TEXT("Error in COM+ TPC-C Component: "));
        lpszStrings[0] = szMsg;
        lpszStrings[1] = lpszMsg;
        if (hEventSource != NULL)
        {
            ReportEvent(hEventSource, // handle of event source
            EVENTLOG_ERROR_TYPE, // event type
            0, // event category
            0, // event ID
            NULL, // current user's SID
            2, // strings in lpszStrings
            0, // no bytes of raw data
            (LPCTSTR *)lpszStrings, // array of error strings
            NULL); // no raw data
            (VOID) DeregisterEventSource(hEventSource);
        }
    }

    inline void ReleaseInterface(IUnknown *pUnk)
    {
        if (pUnk)
        {
            pUnk->Release();
            pUnk = NULL;
        }
    }

    /* FUNCTION: CCOMPONENT_ERR::ErrorText
    *

```

```

    */
    char* CCOMPONENT_ERR::ErrorText(void)
    {
        static SERRORMSG errorMsgs[] =
        {
            { ERR_MISSING_REGISTRY_ENTRIES, "Required entries missing
            from registry." },
            { ERR_LOADDLL_FAILED, "Load of DLL failed. DLL=" },
            { ERR_GETPROCADDR_FAILED, "Could not map proc in DLL.
            GetProcAddress error. DLL=" },
            { ERR_UNKNOWN_DB_PROTOCOL, "Unknown database protocol
            specified in registry." },
            { 0, "" }
        };
        char szTmp[256];
        int i = 0;
        while (TRUE)
        {
            if (errorMsgs[i].szMsg[0] == 0)
            {
                strcpy( szTmp, "Unknown error number." );
                break;
            }
            if (m_Error == errorMsgs[i].iError)
            {
                strcpy( szTmp, errorMsgs[i].szMsg );
                break;
            }
            i++;
        }
        if (m_szTextDetail)
            strcat( szTmp, m_szTextDetail );
        if (m_SystemErr)
            wsprintf( szTmp+strlen(szTmp), " Error=%d",
            m_SystemErr );
        m_szErrorText = new char[strlen(szTmp)+1];
        strcpy( m_szErrorText, szTmp );
        return m_szErrorText;
    }

    CTPCC_Common::CTPCC_Common()
    {
        m_pTxn = NULL;
        m_bCanBePooled = TRUE;
    }

    CTPCC_Common::~CTPCC_Common()
    {
        if (m_pTxn)
            delete m_pTxn;
    }

    HRESULT CTPCC_Common::CallSetComplete()
    {
        IObjectContext* pObjectContext = NULL;
        // get our object context
        HRESULT hr = CoGetObjectContext( IID_IObjectContext, (void
        **)&pObjectContext
        );
        pObjectContext->SetComplete();
        ReleaseInterface(pObjectContext);
        return hr;
    }

    //
    // called by the ctor activator
    //

```

```

STDMETHODIMP CTPCC_Common::Construct(IDispatch * pUnk)
{
    // Code to access construction string, if needed later...
    // if (!pUnk)
    // return E_UNEXPECTED;
    // IOObjectConstructString * pString = NULL;
    // HRESULT hr = pUnk->QueryInterface(IID_IOObjectConstructString,
    (void **)&pString);
    // pString->Release();
    // DebugBreak();
    try
    {
        if (Reg.eDB_Protocol == DBLIB)
            m_pTxn = pCTPCC_DBLIB_new( Reg.szDbUser,
Reg.szDbPassword, Reg.szDbName );
    }
    catch (CBaseErr *e)
    {
        WriteMessageToEventLog(e->ErrorText());
        delete e;
        return E_FAIL;
    }
    catch (...)
    {
        WriteMessageToEventLog(TEXT("Unhandled exception in
object ::Construct"));
        return E_FAIL;
    }
    return S_OK;
}
HRESULT CTPCC_Common::NewOrder(VARIANT txn_in, VARIANT* txn_out)
{
    PNEW_ORDER_DATA pNewOrder;
    COM_DATA *pData;
    try
    {
        pData = (COM_DATA*)txn_in.parray->pvData;
        pNewOrder = m_pTxn->BuffAddr_NewOrder();
        memcpy(pNewOrder, &pData->u.NewOrder,
sizeof(NEW_ORDER_DATA));
        m_pTxn->NewOrder(); // do the actual txn
        VariantInit(txn_out);
        txn_out->vt = VT_SAFEARRAY;
        txn_out->parray = SafeArrayCreateVector(VT_UI1,
            txn_in.parray->rgsabound->cElements,
            txn_in.parray->rgsabound->cElements);
        pData = (COM_DATA*) txn_out->parray->pvData;
        memcpy( &pData->u.NewOrder, pNewOrder,
sizeof(NEW_ORDER_DATA));
        pData->retval = ERR_SUCCESS;
        pData->error = 0;
        return S_OK;
    }
    catch (CDBLIBERR *e)
    {
        // check for lost database connection; if yes, component
is toast
        if ( ((e->ErrorType() == ERR_TYPE_DBLIB) &&
            ((e->ErrorNum() == ERR_ATTACH) || (e->ErrorNum()
== ERR_DETACH)) )
            m_bCanBePooled = FALSE;
        pData->retval = e->ErrorType();
        pData->error = e->ErrorNum();
    }
}

```

```

        delete e;
        return E_FAIL;
    }
    catch (...)
    {
        WriteMessageToEventLog(TEXT("Unhandled exception."));
        pData->retval = ERR_TYPE_LOGIC;
        pData->error = 0;
        m_bCanBePooled = FALSE;
        return E_FAIL;
    }
}
HRESULT CTPCC_Common::Payment(VARIANT txn_in, VARIANT* txn_out)
{
    PPAYMENT_DATA pPayment;
    COM_DATA *pData;
    try
    {
        pData = (COM_DATA*)txn_in.parray->pvData;
        pPayment = m_pTxn->BuffAddr_Payment();
        memcpy(pPayment, &pData->u.Payment,
sizeof(PAYMENT_DATA));
        m_pTxn->Payment(); // do the actual txn
        VariantInit(txn_out);
        txn_out->vt = VT_SAFEARRAY;
        txn_out->parray = SafeArrayCreateVector( VT_UI1,
            txn_in.parray->rgsabound->cElements,
            txn_in.parray->rgsabound->cElements);
        pData = (COM_DATA*) txn_out->parray->pvData;
        memcpy( &pData->u.Payment, pPayment,
sizeof(PAYMENT_DATA));
        pData->retval = ERR_SUCCESS;
        pData->error = 0;
        return S_OK;
    }
    catch (CDBLIBERR *e)
    {
        // check for lost database connection; if yes, component
is toast
        if ( ((e->ErrorType() == ERR_TYPE_DBLIB) &&
            ((e->ErrorNum() == ERR_ATTACH) || (e->ErrorNum()
== ERR_DETACH)) )
            m_bCanBePooled = FALSE;
        pData->retval = e->ErrorType();
        pData->error = e->ErrorNum();
        delete e;
        return E_FAIL;
    }
    catch (...)
    {
        WriteMessageToEventLog(TEXT("Unhandled exception."));
        pData->retval = ERR_TYPE_LOGIC;
        pData->error = 0;
        m_bCanBePooled = FALSE;
        return E_FAIL;
    }
}
HRESULT CTPCC_Common::StockLevel(VARIANT txn_in, VARIANT* txn_out)
{
    PSTOCK_LEVEL_DATA pStockLevel;
    COM_DATA *pData;
    try
    {

```

```

        pData = (COM_DATA*)txn_in.parray->pvData;
        pStockLevel = m_pTxn->BuffAddr_StockLevel();
        memcpy(pStockLevel, &pData->u.StockLevel,
sizeof(STOCK_LEVEL_DATA));
        m_pTxn->StockLevel();
        VariantInit(txn_out);
        txn_out->vt = VT_SAFEARRAY;
        txn_out->parray = SafeArrayCreateVector( VT_UI1,
        txn_in.parray->rgsabound->cElements,
        txn_in.parray->rgsabound->cElements);
        pData = (COM_DATA*)txn_out->parray->pvData;
        memcpy( &pData->u.StockLevel, pStockLevel,
sizeof(STOCK_LEVEL_DATA));
        pData->retval = ERR_SUCCESS;
        pData->error = 0;
        return S_OK;
    }
    //catch (CBaseErr *e)
    catch (CDBLIBERR *e)
    {
        // check for lost database connection; if yes, component
is toast
        if ( ((e->ErrorType() == ERR_TYPE_DBLIB) &&
        ((e->ErrorNum() == ERR_ATTACH)) || (e->ErrorNum()
== ERR_DETACH)) )
        {
            m_bCanBePooled = FALSE;
            pData->retval = e->ErrorType();
            pData->error = e->ErrorNum();
            delete e;
            return E_FAIL;
        }
    }
    catch (...)
    {
        WriteMessageToEventLog(TEXT("Unhandled exception.));
        pData->retval = ERR_TYPE_LOGIC;
        pData->error = 0;
        m_bCanBePooled = FALSE;
        return E_FAIL;
    }
}
HRESULT CTPCC_Common::OrderStatus(VARIANT txn_in, VARIANT* txn_out)
{
    PORDER_STATUS_DATA pOrderStatus;
    COM_DATA *pData;
    try
    {
        pData = (COM_DATA*)txn_in.parray->pvData;
        pOrderStatus = m_pTxn->BuffAddr_OrderStatus();
        memcpy(pOrderStatus, &pData->u.OrderStatus,
        sizeof(ORDER_STATUS_DATA));
        m_pTxn->OrderStatus();
        VariantInit(txn_out);
        txn_out->vt = VT_SAFEARRAY;
        txn_out->parray = SafeArrayCreateVector( VT_UI1,
        txn_in.parray->rgsabound->cElements,
        txn_in.parray->rgsabound->cElements);
        pData = (COM_DATA*)txn_out->parray->pvData;
        memcpy( &pData->u.OrderStatus, pOrderStatus,
        sizeof(ORDER_STATUS_DATA));
        pData->retval = ERR_SUCCESS;
        pData->error = 0;
        return S_OK;
    }
}

```

```

        catch (CDBLIBERR *e)
        {
            // check for lost database connection; if yes, component
is toast
            if ( ((e->ErrorType() == ERR_TYPE_DBLIB) &&
            ((e->ErrorNum() == ERR_ATTACH)) || (e->ErrorNum()
== ERR_DETACH)) )
            {
                m_bCanBePooled = FALSE;
                pData->retval = e->ErrorType();
                pData->error = e->ErrorNum();
                delete e;
                return E_FAIL;
            }
        }
        catch (...)
        {
            WriteMessageToEventLog(TEXT("Unhandled exception.));
            pData->retval = ERR_TYPE_LOGIC;
            pData->error = 0;
            m_bCanBePooled = FALSE;
            return E_FAIL;
        }
    }
}

```

tpcc_com_all.def

```

; tpcc_com_all.def : Declares the module parameters.
LIBRARY "tpcc_com_all.dll"
EXPORTS
DllCanUnloadNow @1 PRIVATE
DllGetClassObject @2 PRIVATE
DllRegisterServer @3 PRIVATE
DllUnregisterServer @4 PRIVATE

```

tpcc_com_all.h

```

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

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

/* File created by MIDL compiler version 6.00.0347 */
/* at Mon Nov 15 21:38:43 2004
*/
/* Compiler settings for .\src\tpcc_com_all.idl:
    Oicf, W1, Zp8, env=Win32 (32b run)
    protocol : dce , ms_ext, c_ext
    error checks: allocation ref bounds_check enum stub_data
    VC __declspec() decoration level:
        __declspec(uuid()), __declspec(selectany), __declspec(novtable)
        DECLSPEC_UUID(), MIDL_INTERFACE()
*/
//@@MIDL_FILE_HEADING( )

/* verify that the <rpcndr.h> version is high enough to compile this
file*/
#ifndef __REQUIRED_RPCNDR_H_VERSION__
#define __REQUIRED_RPCNDR_H_VERSION__ 440
#endif

#include "rpc.h"

```

```

#include "rpcndr.h"

#ifndef __tpcc_com_all_h__
#define __tpcc_com_all_h__

#if defined(_MSC_VER) && (_MSC_VER >= 1020)
#pragma once
#endif

/* Forward Declarations */

#ifndef __TPCC_FWD_DEFINED__
#define __TPCC_FWD_DEFINED__

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

#endif /* __TPCC_FWD_DEFINED__ */

#ifndef __NewOrder_FWD_DEFINED__
#define __NewOrder_FWD_DEFINED__

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

#endif /* __NewOrder_FWD_DEFINED__ */

#ifndef __OrderStatus_FWD_DEFINED__
#define __OrderStatus_FWD_DEFINED__

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

#endif /* __OrderStatus_FWD_DEFINED__ */

#ifndef __Payment_FWD_DEFINED__
#define __Payment_FWD_DEFINED__

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

#endif /* __Payment_FWD_DEFINED__ */

#ifndef __StockLevel_FWD_DEFINED__
#define __StockLevel_FWD_DEFINED__

#ifdef __cplusplus

```

```

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

#endif /* __StockLevel_FWD_DEFINED__ */

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

#ifdef __cplusplus
extern "C"{
#endif

void * __RPC_USER MIDL_user_allocate(size_t);
void __RPC_USER MIDL_user_free( void * );

/* interface __MIDL_itf_tpcc_com_all_0000 */
/* [local] */

extern RPC_IF_HANDLE __MIDL_itf_tpcc_com_all_0000_v0_0_c_ifspec;
extern RPC_IF_HANDLE __MIDL_itf_tpcc_com_all_0000_v0_0_s_ifspec;

#ifndef __TPCCLib_LIBRARY_DEFINED__
#define __TPCCLib_LIBRARY_DEFINED__

/* library TPCCLib */
/* [helpstring][version][uuid] */

EXTERN_C const IID LIBID_TPCCLib;

EXTERN_C const CLSID CLSID_TPCC;

#ifdef __cplusplus
class DECLSPEC_UUID("122A3128-2520-11D3-BA71-00C04FBFE08B")
TPCC;
#endif

EXTERN_C const CLSID CLSID_NewOrder;

#ifdef __cplusplus
class DECLSPEC_UUID("975BAABF-84A7-11D2-BA47-00C04FBFE08B")
NewOrder;
#endif

EXTERN_C const CLSID CLSID_OrderStatus;

#ifdef __cplusplus

```

```

class DECLSPEC_UUID("266836AD-A50D-11D2-BA4E-00C04FBFE08B")
OrderStatus;
#endif

EXTERN_C const CLSID CLSID_Payment;

#ifdef __cplusplus

class DECLSPEC_UUID("CD02F7EF-A4FA-11D2-BA4E-00C04FBFE08B")
Payment;
#endif

EXTERN_C const CLSID CLSID_StockLevel;

#ifdef __cplusplus

class DECLSPEC_UUID("2668369E-A50D-11D2-BA4E-00C04FBFE08B")
StockLevel;
#endif
#endif /* __TPCCLib_LIBRARY_DEFINED__ */

/* Additional Prototypes for ALL interfaces */

/* end of Additional Prototypes */

#ifdef __cplusplus
}
#endif

#endif

```

tpcc_com_all.idl

```

/* FILE: TPCC.IDL
* Microsoft TPC-C Kit Ver. 4.20.000
* Copyright Microsoft, 1999
* All Rights Reserved
*
* not yet audited
*
* PURPOSE: IDL source for TPCC.dll. This file is processed by the MIDL
tool to
* produce the type library (TPCC.tlb) and
marshalling code.
*
* Change history:
* 4.20.000 - first version
*/
interface TPCC;
interface NewOrder;
interface OrderStatus;
interface Payment;
interface StockLevel;
import "oidl.idl";
import "ocidl.idl";
import "..\tpcc_com_ps\src\tpcc_com_ps.idl";
[
  uuid(122A3117-2520-11D3-BA71-00C04FBFE08B),
  version(1.0),
  helpstring("TPC-C 1.0 Type Library")
]
library TPCCLib
{

```

```

importlib("stdole32.tlb");
importlib("stdole2.tlb");
[
  uuid(122A3128-2520-11D3-BA71-00C04FBFE08B),

  helpstring("All Txns Class")
]
coclass TPCC
{
  [default] interface ITPCC;
};
[
  uuid(975BAABF-84A7-11D2-BA47-00C04FBFE08B),
  helpstring("NewOrder Class")
]
coclass NewOrder
{
  [default] interface ITPCC;
};
[
  uuid(266836AD-A50D-11D2-BA4E-00C04FBFE08B),
  helpstring("OrderStatus Class")
]
coclass OrderStatus
{
  [default] interface ITPCC;
};
[
  uuid(CD02F7EF-A4FA-11D2-BA4E-00C04FBFE08B),
  helpstring("Payment Class")
]
coclass Payment
{
  [default] interface ITPCC;
};
[
  uuid(2668369E-A50D-11D2-BA4E-00C04FBFE08B),
  helpstring("StockLevel Class")
]
coclass StockLevel
{
  [default] interface ITPCC;
};
};

```

tpcc_com_all.rc

```

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

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

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

```

```

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

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

#ifdef _MAC
////////////////////////////////////
////
//
// Version
//

VS_VERSION_INFO VERSIONINFO
FILEVERSION 1,0,0,1
PRODUCTVERSION 1,0,0,1
FILEFLAGSMASK 0x3fL
#ifdef _DEBUG
FILEFLAGS 0x1L
#else
FILEFLAGS 0x0L
#endif
FILESOS 0x40004L
FILETYPE 0x2L
FILESUBTYPE 0x0L
BEGIN
    BLOCK "StringFileInfo"
    BEGIN
        BLOCK "040904b0"
        BEGIN
            VALUE "Comments", "\0"
            VALUE "CompanyName", "\0"
            VALUE "FileDescription", "tpcc_dblib\0"
            VALUE "FileVersion", "1, 0, 0, 1\0"
            VALUE "InternalName", "tpcc_dblib\0"
            VALUE "LegalCopyright", "Copyright 2004\0"
            VALUE "LegalTrademarks", "\0"
            VALUE "OriginalFilename", "tpcc_dblib.dll\0"
            VALUE "PrivateBuild", "\0"
            VALUE "ProductName", "tpcc_dblib\0"
            VALUE "ProductVersion", "1, 0, 0, 1\0"
            VALUE "SpecialBuild", "\0"
        END
    END
    BLOCK "VarFileInfo"
    BEGIN
        VALUE "Translation", 0x409, 1200
    END
END

#endif // !_MAC

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

```

```

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

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

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

#endif // APSTUDIO_INVOKED

////////////////////////////////////
////
// REGISTRY
//

IDR_TPCC                REGISTRY DISCARDABLE "tpcc_com_all.rgs"
IDR_PAYMENT             REGISTRY DISCARDABLE "tpcc_com_pay.rgs"
IDR_STOCKLEVEL          REGISTRY DISCARDABLE "tpcc_com_sl.rgs"
IDR_ORDERSTATUS         REGISTRY DISCARDABLE "tpcc_com_os.rgs"
IDR_NEWORDER            REGISTRY DISCARDABLE "tpcc_com_no.rgs"

////////////////////////////////////
////
// String Table

STRINGTABLE DISCARDABLE
BEGIN
    IDS_PROJNAME          "tpcc_com_all"
END

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

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

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

```

tpcc_com_all.rgs

```
HKCR
{
    TPCC.AllTxns.1 = s 'All Txns Class'
    {
        CLSID = s '{122A3128-2520-11D3-BA71-00C04FBFE08B}'
    }
    TPCC.AllTxns = s 'TPCC Class'
    {
        CurVer = s 'TPCC.AllTxns.1'
    }
    NoRemove CLSID
    {
        ForceRemove {122A3128-2520-11D3-BA71-00C04FBFE08B} = s
        'TPCC Class'
        {
            ProgID = s 'TPCC.AllTxns.1'
            VersionIndependentProgID = s 'TPCC.AllTxns'
            InprocServer32 = s '%MODULE%'
            {
                val ThreadingModel = s 'Both'
            }
        }
    }
}
```

tpcc_com_all.i.c

```
#pragma warning( disable: 4049 ) /* more than 64k source lines */
/* this ALWAYS GENERATED file contains the IIDs and CLSIDs */
/* link this file in with the server and any clients */

/* File created by MIDL compiler version 6.00.0347 */
/* at Mon Nov 15 21:38:43 2004
*/
/* Compiler settings for .\src\tpcc_com_all.idl:
    Oicf, W1, Zp8, env=win32 (32b run)
    protocol : dce , ms_ext, c_ext
    error checks: allocation ref bounds_check enum stub_data
    VC __declspec() decoration level:
        __declspec(uuid()), __declspec(selectany), __declspec(novtable)
        DECLSPEC_UUID(), MIDL_INTERFACE()
*/
//@@MIDL_FILE_HEADING(  )

#if !defined(_M_IA64) && !defined(_M_AMD64)

#ifdef __cplusplus
extern "C"{
#endif

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

#ifdef _MIDL_USE_GUIDDEF_
```

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

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

#else // !_MIDL_USE_GUIDDEF_

#ifndef __IID_DEFINED__
#define __IID_DEFINED__

typedef struct _IID
{
    unsigned long x;
    unsigned short s1;
    unsigned short s2;
    unsigned char c[8];
} IID;

#endif // __IID_DEFINED__

#ifndef CLSID_DEFINED
#define CLSID_DEFINED
typedef IID CLSID;
#endif // CLSID_DEFINED

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

#endif !_MIDL_USE_GUIDDEF_

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

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

MIDL_DEFINE_GUID(CLSID,
CLSID_NewOrder,0x975BAABF,0x84A7,0x11D2,0xBA,0x47,0x00,0xC0,0x4F,0xBF,0xE0,0x8B);

MIDL_DEFINE_GUID(CLSID,
CLSID_OrderStatus,0x266836AD,0xA50D,0x11D2,0xBA,0x4E,0x00,0xC0,0x4F,0xBF,0xE0,0x8B);

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



```

MIDL_DEFINE_GUID(CLSID,
CLSID_StockLevel, 0x2668369E, 0xA50D, 0x11D2, 0xBA, 0x4E, 0x00, 0xC0, 0x4F, 0xBF,
0xE0, 0x8B);

#undef MIDL_DEFINE_GUID

#ifdef __cplusplus
}
#endif

#endif /* !defined(_M_IA64) && !defined(_M_AMD64) */

#pragma warning( disable: 4049 ) /* more than 64k source lines */
/* this ALWAYS GENERATED file contains the IIDs and CLSIDs */
/* link this file in with the server and any clients */

/* File created by MIDL compiler version 6.00.0347 */
/* at Mon Nov 15 21:38:43 2004
*/
/* Compiler settings for .\src\tpcc_com_all.idl:
    Oicf, W1, Zp8, env=Win64 (32b run, appending)
    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)

#ifdef __cplusplus
extern "C"{
#endif

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

#ifdef _MIDL_USE_GUIDDEF_

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

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

#else // !_MIDL_USE_GUIDDEF_

#ifndef __IID_DEFINED__
#define __IID_DEFINED__

```

```

typedef struct _IID
{
    unsigned long x;
    unsigned short s1;
    unsigned short s2;
    unsigned char c[8];
} IID;

#endif // __IID_DEFINED__

#ifndef CLSID_DEFINED
#define CLSID_DEFINED
typedef IID CLSID;
#endif // CLSID_DEFINED

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

#endif !_MIDL_USE_GUIDDEF_

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

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

MIDL_DEFINE_GUID(CLSID,
CLSID_NewOrder, 0x975BAABF, 0x84A7, 0x11D2, 0xBA, 0x47, 0x00, 0xC0, 0x4F, 0xBF, 0x
E0, 0x8B);

MIDL_DEFINE_GUID(CLSID,
CLSID_OrderStatus, 0x266836AD, 0xA50D, 0x11D2, 0xBA, 0x4E, 0x00, 0xC0, 0x4F, 0xBF
, 0xE0, 0x8B);

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

MIDL_DEFINE_GUID(CLSID,
CLSID_StockLevel, 0x2668369E, 0xA50D, 0x11D2, 0xBA, 0x4E, 0x00, 0xC0, 0x4F, 0xBF,
0xE0, 0x8B);

#undef MIDL_DEFINE_GUID

#ifdef __cplusplus
}
#endif

#endif /* defined(_M_IA64) || defined(_M_AMD64) */

```

tpcc_com_no.rgs

```

HKCR
{

```

```

        TPCC.NewOrder.1 = s 'NewOrder Class'
    {
        CLSID = s '{975BAABF-84A7-11D2-BA47-00C04FBFE08B}'
    }
    TPCC.NewOrder = s 'NewOrder Class'
    {
        CurVer = s 'TPCC.NewOrder.1'
    }
    NoRemove CLSID
    {
        ForceRemove {975BAABF-84A7-11D2-BA47-00C04FBFE08B} = s 'NewOrder
        Class'
    }
    {
        ProgID = s 'TPCC.NewOrder.1'
        VersionIndependentProgID = s 'TPCC.NewOrder'
        InprocServer32 = s '%MODULE%'
    }
    {
        val ThreadingModel = s 'Both'
    }
}
}
}

```

tpcc_com_os.rgs

```

HKCR
{
    TPCC.OrderStatus.1 = s 'OrderStatus Class'
    {
        CLSID = s '{266836AD-A50D-11D2-BA4E-00C04FBFE08B}'
    }
    TPCC.OrderStatus = s 'OrderStatus Class'
    {
        CurVer = s 'TPCC.OrderStatus.1'
    }
    NoRemove CLSID
    {
        ForceRemove {266836AD-A50D-11D2-BA4E-00C04FBFE08B} = s 'OrderStatus
        Class'
    }
    {
        ProgID = s 'TPCC.OrderStatus.1'
        VersionIndependentProgID = s 'TPCC.OrderStatus'
        InprocServer32 = s '%MODULE%'
    }
    {
        val ThreadingModel = s 'Both'
    }
}
}
}

```

tpcc_com_pay.rgs

```

HKCR
{
    TPCC.Payment.1 = s 'Payment Class'
    {
        CLSID = s '{CD02F7EF-A4FA-11D2-BA4E-00C04FBFE08B}'
    }
    TPCC.Payment = s 'Payment Class'
    {
        CurVer = s 'TPCC.Payment.1'
    }
    NoRemove CLSID
}

```

```

{
    ForceRemove {CD02F7EF-A4FA-11D2-BA4E-00C04FBFE08B} = s 'Payment
    Class'
    {
        ProgID = s 'TPCC.Payment.1'
        VersionIndependentProgID = s 'TPCC.Payment'
        InprocServer32 = s '%MODULE%'
    }
    {
        val ThreadingModel = s 'Both'
    }
}
}
}

```

tpcc_com_ps.h

```

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

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

/* File created by MIDL compiler version 6.00.0347 */
/* at Mon Nov 15 21:38:37 2004
*/
/*
Compiler settings for .\src\tpcc_com_ps.idl:
    Oicf, W1, Zp8, env=win32 (32b run)
    protocol: dce , ms_ext, c_ext
    error checks: allocation ref bounds_check enum stub_data
    VC __declspec() decoration level:
        __declspec(uuid()), __declspec(selectany), __declspec(novtable)
        DECLSPEC_UUID(), MIDL_INTERFACE()
*/
//@@MIDL_FILE_HEADING(  )

/* verify that the <rpcndr.h> version is high enough to compile this
file*/
#ifndef __REQUIRED_RPCNDR_H_VERSION__
#define __REQUIRED_RPCNDR_H_VERSION__ 440
#endif

#include "rpc.h"
#include "rpcndr.h"

#ifndef __RPCNDR_H_VERSION__
#error this stub requires an updated version of <rpcndr.h>
#endif // __RPCNDR_H_VERSION__

#ifndef COM_NO_WINDOWS_H
#include "windows.h"
#include "ole2.h"
#endif /*COM_NO_WINDOWS_H*/

#ifndef __tpcc_com_ps_h__
#define __tpcc_com_ps_h__

#if defined(_MSC_VER) && (_MSC_VER >= 1020)
#pragma once
#endif

/* Forward Declarations */

```

```

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

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

#ifdef __cplusplus
extern "C"{
#endif

void * __RPC_USER MIDL_user_allocate(size_t);
void __RPC_USER MIDL_user_free( void * );

/* interface __MIDL_itf_tpcc_com_ps_0000 */
/* [local] */

extern RPC_IF_HANDLE __MIDL_itf_tpcc_com_ps_0000_v0_0_c_ifspec;
extern RPC_IF_HANDLE __MIDL_itf_tpcc_com_ps_0000_v0_0_s_ifspec;

#ifndef __ITPCC_INTERFACE_DEFINED__
#define __ITPCC_INTERFACE_DEFINED__

/* interface ITPCC */
/* [unique][helpstring][uuid][oleautomation][object] */

EXTERN_C const IID IID_ITPCC;

#if defined(__cplusplus) && !defined(CINTERFACE)

    MIDL_INTERFACE("FEEE6AA2-84B1-11d2-BA47-00C04FBFE08B")
    ITPCC : public IUnknown
    {
    public:
        virtual HRESULT __stdcall NewOrder(
            /* [in] */ VARIANT txn_in,
            /* [out] */ VARIANT *txn_out) = 0;

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

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

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

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

        virtual HRESULT __stdcall CallSetComplete( void) = 0;

```

```

};

#else /* C style interface */

typedef struct ITPCCVtbl
{
    BEGIN_INTERFACE

    HRESULT ( STDMETHODCALLTYPE *QueryInterface )(
        ITPCC * This,
        /* [in] */ REFIID riid,
        /* [iid_is][out] */ void **ppvObject);

    ULONG ( STDMETHODCALLTYPE *AddRef )(
        ITPCC * This);

    ULONG ( STDMETHODCALLTYPE *Release )(
        ITPCC * This);

    HRESULT ( STDMETHODCALLTYPE *NewOrder )(
        ITPCC * This,
        /* [in] */ VARIANT txn_in,
        /* [out] */ VARIANT *txn_out);

    HRESULT ( STDMETHODCALLTYPE *Payment )(
        ITPCC * This,
        /* [in] */ VARIANT txn_in,
        /* [out] */ VARIANT *txn_out);

    HRESULT ( STDMETHODCALLTYPE *Delivery )(
        ITPCC * This,
        /* [in] */ VARIANT txn_in,
        /* [out] */ VARIANT *txn_out);

    HRESULT ( STDMETHODCALLTYPE *StockLevel )(
        ITPCC * This,
        /* [in] */ VARIANT txn_in,
        /* [out] */ VARIANT *txn_out);

    HRESULT ( STDMETHODCALLTYPE *OrderStatus )(
        ITPCC * This,
        /* [in] */ VARIANT txn_in,
        /* [out] */ VARIANT *txn_out);

    HRESULT ( STDMETHODCALLTYPE *CallSetComplete )(
        ITPCC * This);

    END_INTERFACE
} ITPCCVtbl;

interface ITPCC
{
    CONST_VTBL struct ITPCCVtbl *lpVtbl;
};

#ifdef COBJMACROS

#define ITPCC_QueryInterface(This,riid,ppvObject) \
    (This->lpVtbl->QueryInterface(This,riid,ppvObject)

```

```

#define ITPCC_AddRef(This) \
    (This)->lpVtbl -> AddRef(This)

#define ITPCC_Release(This) \
    (This)->lpVtbl -> Release(This)

#define ITPCC_NewOrder(This,txn_in,txn_out) \
    (This)->lpVtbl -> NewOrder(This,txn_in,txn_out)

#define ITPCC_Payment(This,txn_in,txn_out) \
    (This)->lpVtbl -> Payment(This,txn_in,txn_out)

#define ITPCC_Delivery(This,txn_in,txn_out) \
    (This)->lpVtbl -> Delivery(This,txn_in,txn_out)

#define ITPCC_StockLevel(This,txn_in,txn_out) \
    (This)->lpVtbl -> StockLevel(This,txn_in,txn_out)

#define ITPCC_OrderStatus(This,txn_in,txn_out) \
    (This)->lpVtbl -> OrderStatus(This,txn_in,txn_out)

#define ITPCC_CallSetComplete(This) \
    (This)->lpVtbl -> CallSetComplete(This)

```

```

#endif /* COBJMACROS */

```

```

#endif /* C style interface */

```

```

HRESULT __stdcall ITPCC_NewOrder_Proxy(
    ITPCC * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT *txn_out);

```

```

void __RPC_STUB ITPCC_NewOrder_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *_pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,
    DWORD *_pdwStubPhase);

```

```

HRESULT __stdcall ITPCC_Payment_Proxy(
    ITPCC * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT *txn_out);

```

```

void __RPC_STUB ITPCC_Payment_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *_pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,
    DWORD *_pdwStubPhase);

```

```

HRESULT __stdcall ITPCC_Delivery_Proxy(
    ITPCC * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT *txn_out);

```

```

void __RPC_STUB ITPCC_Delivery_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *_pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,
    DWORD *_pdwStubPhase);

```

```

HRESULT __stdcall ITPCC_StockLevel_Proxy(
    ITPCC * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT *txn_out);

```

```

void __RPC_STUB ITPCC_StockLevel_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *_pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,
    DWORD *_pdwStubPhase);

```

```

HRESULT __stdcall ITPCC_OrderStatus_Proxy(
    ITPCC * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT *txn_out);

```

```

void __RPC_STUB ITPCC_OrderStatus_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *_pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,
    DWORD *_pdwStubPhase);

```

```

HRESULT __stdcall ITPCC_CallSetComplete_Proxy(
    ITPCC * This);

```

```

void __RPC_STUB ITPCC_CallSetComplete_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *_pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,
    DWORD *_pdwStubPhase);

```

```

#endif /* __ITPCC_INTERFACE_DEFINED__ */

```

```

/* Additional Prototypes for ALL interfaces */

```

```

unsigned long __RPC_USER VARIANT_UserSize( unsigned
long *, unsigned long , VARIANT * );
unsigned char * __RPC_USER VARIANT_UserMarshal( unsigned long *,
unsigned char *, VARIANT * );
unsigned char * __RPC_USER VARIANT_UserUnmarshal(unsigned long *,
unsigned char *, VARIANT * );
void __RPC_USER VARIANT_UserFree( unsigned
long *, VARIANT * );

```

```

/* end of Additional Prototypes */

```

```

#ifdef __cplusplus
}
#endif

#endif

```

tpcc_com_sl.rgs

```

HKCR
{
  TPCC.StockLevel.1 = s 'StockLevel Class'
  {
    CLSID = s '{2668369E-A50D-11D2-BA4E-00C04FBFE08B}'
  }
  TPCC.StockLevel = s 'StockLevel Class'
  {
    CurVer = s 'TPCC.StockLevel.1'
  }
  NoRemove CLSID
  {
    ForceRemove {2668369E-A50D-11D2-BA4E-00C04FBFE08B} = s 'StockLevel Class'
  }
  {
    ProgID = s 'TPCC.StockLevel.1'
    VersionIndependentProgID = s 'TPCC.StockLevel'
    InprocServer32 = s '%MODULE%'
  }
  {
    val ThreadingModel = s 'Both'
  }
}

```

dlldata.c

```

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

DO NOT ALTER THIS FILE

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

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

*****/

#include <rpcproxy.h>

#ifdef __cplusplus
extern "C" {
#endif

EXTERN_PROXY_FILE( tpcc_com_ps )

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

```

```

DLLDATA_ROUTINES( aProxyFileList, GET_DLL_CLSID )

```

```

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

```

```

/* end of generated dlldata file */

```

tpcc_com_ps.def

```

LIBRARY      "tpcc_com_ps"

DESCRIPTION  'Proxy/Stub DLL'

EXPORTS
  DllGetClassObject      @1      PRIVATE
  DllCanUnloadNow        @2      PRIVATE
  GetProxyDllInfo        @3      PRIVATE
  DllRegisterServer      @4      PRIVATE
  DllUnregisterServer     @5      PRIVATE

```

tpcc_com_ps.h

```

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

```

```

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

```

```

/* File created by MIDL compiler version 6.00.0347 */
/* at Mon Nov 15 21:38:37 2004
*/
/* Compiler settings for .\src\tpcc_com_ps.idl:
   Oicf, W1, Zp8, env=Win32 (32b run)
   protocol : dce , ms_ext, c_ext
   error checks: allocation ref bounds_check enum stub_data
   VC __declspec() decoration level:
       __declspec(uuid()), __declspec(selectany), __declspec(novtable)
       DECLSPEC_UUID(), MIDL_INTERFACE()
*/
//@@MIDL_FILE_HEADING(  )

```

```

/* verify that the <rpcndr.h> version is high enough to compile this
file*/

```

```

#ifndef __REQUIRED_RPCNDR_H_VERSION__
#define __REQUIRED_RPCNDR_H_VERSION__ 440
#endif

```

```

#include "rpc.h"
#include "rpcndr.h"

```

```

#ifndef __RPCNDR_H_VERSION__
#error this stub requires an updated version of <rpcndr.h>
#endif // __RPCNDR_H_VERSION__

```

```

#ifndef COM_NO_WINDOWS_H
#include "windows.h"
#include "ole2.h"

```

```

#endif /*COM_NO_WINDOWS_H*/

#ifndef __tpcc_com_ps_h__
#define __tpcc_com_ps_h__

#if defined(_MSC_VER) && (_MSC_VER >= 1020)
#pragma once
#endif

/* Forward Declarations */

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

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

#ifdef __cplusplus
extern "C"{
#endif

void * __RPC_USER MIDL_user_allocate(size_t);
void __RPC_USER MIDL_user_free( void * );

/* interface __MIDL_itf_tpcc_com_ps_0000 */
/* [local] */

extern RPC_IF_HANDLE __MIDL_itf_tpcc_com_ps_0000_v0_0_c_ifspec;
extern RPC_IF_HANDLE __MIDL_itf_tpcc_com_ps_0000_v0_0_s_ifspec;

#ifndef __ITPCC_INTERFACE_DEFINED__
#define __ITPCC_INTERFACE_DEFINED__

/* interface ITPCC */
/* [unique][helpstring][uuid][oleautomation][object] */

EXTERN_C const IID IID_ITPCC;

#if defined(__cplusplus) && !defined(CINTERFACE)
    MIDL_INTERFACE("FEEE6AA2-84B1-11d2-BA47-00C04FBFE08B")
    ITPCC : public IUnknown
    {
    public:
        virtual HRESULT __stdcall NewOrder(
            /* [in] */ VARIANT txn_in,
            /* [out] */ VARIANT *txn_out) = 0;

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

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

```

```

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

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

        virtual HRESULT __stdcall CallSetComplete( void) = 0;
    };

#else /* C style interface */
    typedef struct ITPCCVtbl
    {
        BEGIN_INTERFACE

        HRESULT ( STDMETHODCALLTYPE *QueryInterface )(
            ITPCC * This,
            /* [in] */ REFIID riid,
            /* [iid_is][out] */ void **ppvObject);

        ULONG ( STDMETHODCALLTYPE *AddRef )(
            ITPCC * This);

        ULONG ( STDMETHODCALLTYPE *Release )(
            ITPCC * This);

        HRESULT ( STDMETHODCALLTYPE *NewOrder )(
            ITPCC * This,
            /* [in] */ VARIANT txn_in,
            /* [out] */ VARIANT *txn_out);

        HRESULT ( STDMETHODCALLTYPE *Payment )(
            ITPCC * This,
            /* [in] */ VARIANT txn_in,
            /* [out] */ VARIANT *txn_out);

        HRESULT ( STDMETHODCALLTYPE *Delivery )(
            ITPCC * This,
            /* [in] */ VARIANT txn_in,
            /* [out] */ VARIANT *txn_out);

        HRESULT ( STDMETHODCALLTYPE *StockLevel )(
            ITPCC * This,
            /* [in] */ VARIANT txn_in,
            /* [out] */ VARIANT *txn_out);

        HRESULT ( STDMETHODCALLTYPE *OrderStatus )(
            ITPCC * This,
            /* [in] */ VARIANT txn_in,
            /* [out] */ VARIANT *txn_out);

        HRESULT ( STDMETHODCALLTYPE *CallSetComplete )(
            ITPCC * This);

        END_INTERFACE
    } ITPCCVtbl;

    interface ITPCC
    {

```

```

        CONST_VTBL struct ITPCCVtbl *lpVtbl;
    };

#ifdef COBJMACROS

#define ITPCC_QueryInterface(This, riid, ppvObject) \
    (This)->lpVtbl -> QueryInterface(This, riid, ppvObject)

#define ITPCC_AddRef(This) \
    (This)->lpVtbl -> AddRef(This)

#define ITPCC_Release(This) \
    (This)->lpVtbl -> Release(This)

#define ITPCC_NewOrder(This, txn_in, txn_out) \
    (This)->lpVtbl -> NewOrder(This, txn_in, txn_out)

#define ITPCC_Payment(This, txn_in, txn_out) \
    (This)->lpVtbl -> Payment(This, txn_in, txn_out)

#define ITPCC_Delivery(This, txn_in, txn_out) \
    (This)->lpVtbl -> Delivery(This, txn_in, txn_out)

#define ITPCC_StockLevel(This, txn_in, txn_out) \
    (This)->lpVtbl -> StockLevel(This, txn_in, txn_out)

#define ITPCC_OrderStatus(This, txn_in, txn_out) \
    (This)->lpVtbl -> OrderStatus(This, txn_in, txn_out)

#define ITPCC_CallSetComplete(This) \
    (This)->lpVtbl -> CallSetComplete(This)

#endif /* COBJMACROS */

#endif /* C style interface */

HRESULT __stdcall ITPCC_NewOrder_Proxy(
    ITPCC * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT *txn_out);

void __RPC_STUB ITPCC_NewOrder_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *_pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,
    DWORD *_pdwStubPhase);

HRESULT __stdcall ITPCC_Payment_Proxy(
    ITPCC * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT *txn_out);

void __RPC_STUB ITPCC_Payment_Stub(

```

```

    IRpcStubBuffer *This,
    IRpcChannelBuffer *_pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,
    DWORD *_pdwStubPhase);

HRESULT __stdcall ITPCC_Delivery_Proxy(
    ITPCC * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT *txn_out);

void __RPC_STUB ITPCC_Delivery_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *_pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,
    DWORD *_pdwStubPhase);

HRESULT __stdcall ITPCC_StockLevel_Proxy(
    ITPCC * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT *txn_out);

void __RPC_STUB ITPCC_StockLevel_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *_pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,
    DWORD *_pdwStubPhase);

HRESULT __stdcall ITPCC_OrderStatus_Proxy(
    ITPCC * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT *txn_out);

void __RPC_STUB ITPCC_OrderStatus_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *_pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,
    DWORD *_pdwStubPhase);

HRESULT __stdcall ITPCC_CallSetComplete_Proxy(
    ITPCC * This);

void __RPC_STUB ITPCC_CallSetComplete_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *_pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,
    DWORD *_pdwStubPhase);

#endif /* __ITPCC_INTERFACE_DEFINED__ */

/* Additional Prototypes for ALL interfaces */

```

```

unsigned long      __RPC_USER VARIANT_UserSize(      unsigned
long *, unsigned long      , VARIANT * );
unsigned char * __RPC_USER VARIANT_UserMarshal( unsigned long *,
unsigned char *, VARIANT * );
unsigned char * __RPC_USER VARIANT_UserUnmarshal(unsigned long *,
unsigned char *, VARIANT * );
void      __RPC_USER VARIANT_UserFree(      unsigned
long *, VARIANT * );

```

```
/* end of Additional Prototypes */
```

```

#ifdef __cplusplus
}
#endif

```

```
#endif
```

tpcc_com_ps.idl

```

/*      FILE:      ITPCC.IDL
*      Microsoft TPC-C Kit Ver. 4.20.000
*      Copyright Microsoft, 1999
*      All Rights Reserved
*
*      not yet audited
*
*      PURPOSE:      Defines the interface used by TPCC. This interface
can be implemented by C++ components.
*
*      Change history:
*      4.20.000 - first version
*/

```

```
// Forward declare all types defined
```

```

interface ITPCC;
import "oaidl.idl";
import "ocidl.idl";

[
    object,
    oleautomation,
    uuid(FEEE6AA2-84B1-11d2-BA47-00C04FBFE08B),
    helpstring("ITPCC Interface"),
    pointer_default(unique)
]
interface ITPCC : IUnknown
{
    HRESULT STDMETHODCALLTYPE NewOrder(
        [in] VARIANT txn_in,
        [out] VARIANT *txn_out
    );

    HRESULT STDMETHODCALLTYPE Payment(
        [in] VARIANT txn_in,
        [out] VARIANT *txn_out
    );

    HRESULT STDMETHODCALLTYPE Delivery(
        [in] VARIANT txn_in,

```

```

[out] VARIANT *txn_out
    );

    HRESULT STDMETHODCALLTYPE StockLevel(
        [in] VARIANT txn_in,
        [out] VARIANT *txn_out
    );

    HRESULT STDMETHODCALLTYPE OrderStatus(
        [in] VARIANT txn_in,
        [out] VARIANT *txn_out
    );

    HRESULT STDMETHODCALLTYPE CallSetComplete(
        [in] VARIANT txn_in,
        [out] VARIANT *txn_out
    );

```

```
}; // interface ITPCC
```

tpcc_com_ps.i.c

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

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

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

```

/* File created by MIDL compiler version 6.00.0347 */
/* at Mon Nov 15 21:38:37 2004
*/

```

```

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

```

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

```
#if !defined(_M_IA64) && !defined(_M_AMD64)
```

```

#ifdef __cplusplus
extern "C"{
#endif

```

```

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

```

```
#ifdef _MIDL_USE_GUIDDEF_
```

```

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

```



```

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

#else // !_MIDL_USE_GUIDDEF_

#ifndef __IID_DEFINED__
#define __IID_DEFINED__

typedef struct _IID
{
    unsigned long x;
    unsigned short s1;
    unsigned short s2;
    unsigned char c[8];
} IID;

#endif // __IID_DEFINED__

#ifndef CLSID_DEFINED
#define CLSID_DEFINED
typedef IID CLSID;
#endif // CLSID_DEFINED

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

#endif !_MIDL_USE_GUIDDEF_

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

#undef MIDL_DEFINE_GUID

#ifdef __cplusplus
}
#endif

#endif /* !defined(_M_IA64) && !defined(_M_AMD64) */

#pragma warning( disable: 4049 ) /* more than 64k source lines */
/* this ALWAYS GENERATED file contains the IIDs and CLSIDs */
/* link this file in with the server and any clients */

/* File created by MIDL compiler version 6.00.0347 */
/* at Mon Nov 15 21:38:37 2004 */
/* Compiler settings for .\src\tpcc_com_ps.idl:
    0icf, w1, Zp8, env=win64 (32b run, appending)
    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(  )

```

```

#if defined(_M_IA64) || defined(_M_AMD64)

#ifdef __cplusplus
extern "C"{
#endif

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

#ifdef _MIDL_USE_GUIDDEF_

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

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

#else // !_MIDL_USE_GUIDDEF_

#ifndef __IID_DEFINED__
#define __IID_DEFINED__

typedef struct _IID
{
    unsigned long x;
    unsigned short s1;
    unsigned short s2;
    unsigned char c[8];
} IID;

#endif // __IID_DEFINED__

#ifndef CLSID_DEFINED
#define CLSID_DEFINED
typedef IID CLSID;
#endif // CLSID_DEFINED

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

#endif !_MIDL_USE_GUIDDEF_

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

#undef MIDL_DEFINE_GUID

#ifdef __cplusplus
}
#endif

#endif /* defined(_M_IA64) || defined(_M_AMD64) */

```

tpcc_com_ps_p.c

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

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

/* File created by MIDL compiler version 6.00.0347 */
/* at Mon Nov 15 21:38:37 2004
*/
/* Compiler settings for .\src\tpcc_com_ps.idl:
    Oicf, W1, Zp8, env=Win32 (32b run)
    protocol : dce , ms_ext, c_ext
    error checks: allocation ref bounds_check enum stub_data
    VC __declspec() decoration level:
        __declspec(uuid()), __declspec(selectany), __declspec(novtable)
    DECLSPEC_UUID(), MIDL_INTERFACE()
*/
//@@MIDL_FILE_HEADING( )

#if !defined(_M_IA64) && !defined(_M_AMD64)
#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__ 440
#endif

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

#include "tpcc_com_ps.h"

#define TYPE_FORMAT_STRING_SIZE 1023
#define PROC_FORMAT_STRING_SIZE 193
#define TRANSMIT_AS_TABLE_SIZE 0
#define WIRE_MARSHAL_TABLE_SIZE 1

typedef struct _MIDL_TYPE_FORMAT_STRING
{
    short Pad;
    unsigned char Format[ TYPE_FORMAT_STRING_SIZE ];
} MIDL_TYPE_FORMAT_STRING;

typedef struct _MIDL_PROC_FORMAT_STRING
{
    short Pad;
    unsigned char Format[ PROC_FORMAT_STRING_SIZE ];
} MIDL_PROC_FORMAT_STRING;

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 ITPCC_ServerInfo;
extern const MIDL_STUBLESS_PROXY_INFO ITPCC_ProxyInfo;
```

```
extern const USER_MARSHAL_ROUTINE_QUADRUPLE
UserMarshalRoutines[ WIRE_MARSHAL_TABLE_SIZE ];
```

```
#if !defined(__RPC_WIN32__)
#error Invalid build platform for this stub.
#endif

#if !(TARGET_IS_NT40_OR_LATER)
#error You need a Windows NT 4.0 or later to run this stub because it
uses these features:
#error -Oicf or -Oicf, [wire_marshal] or [user_marshal] attribute.
#error However, your C/C++ compilation flags indicate you intend to run
this app on earlier systems.
#error This app will die there with the RPC_X_WRONG_STUB_VERSION error.
#endif
```

```
static const MIDL_PROC_FORMAT_STRING __MIDL_ProcFormatString =
{
    0,
```

```
/* Procedure NewOrder */
```

```
0x33, /* FC_AUTO_HANDLE */
0x6c, /* Old Flags: object, 0i2 */
/* 2 */ NdrFcLong( 0x0 ), /* 0 */
/* 6 */ NdrFcShort( 0x3 ), /* 3 */
/* 8 */ NdrFcShort( 0x1c ), /* x86 Stack size/offset = 28 */
/* 10 */ NdrFcShort( 0x0 ), /* 0 */
/* 12 */ NdrFcShort( 0x8 ), /* 8 */
/* 14 */ 0x7, /* 0i2 Flags: srv must size, clt must
size, has return, */
0x3, /* 3 */

/* Parameter txn_in */
/* 16 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in,
by val, */
/* 18 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
/* 20 */ NdrFcShort( 0x3e2 ), /* Type Offset=994 */
```

```
/* Parameter txn_out */
```

```
/* 22 */ NdrFcShort( 0x4113 ), /* Flags: must size, must free,
out, simple ref, srv alloc size=16 */
/* 24 */ NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
/* 26 */ NdrFcShort( 0x3f4 ), /* Type Offset=1012 */
```

```
/* Return value */
```

```

/* 28 */      NdrFcShort( 0x70 ), /* Flags: out, return, base type,
*/
/* 30 */      NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
/* 32 */      0x8, /* FC_LONG */
/* 0 */
/* Procedure Payment */

/* 34 */      0x33, /* FC_AUTO_HANDLE */
/* 0x6c, /* Old Flags: object, 0i2 */
/* 36 */      NdrFcLong( 0x0 ), /* 0 */
/* 40 */      NdrFcShort( 0x4 ), /* 4 */
/* 42 */      NdrFcShort( 0x1c ), /* x86 Stack size/offset = 28 */
/* 44 */      NdrFcShort( 0x0 ), /* 0 */
/* 46 */      NdrFcShort( 0x8 ), /* 8 */
/* 48 */      0x7, /* 0i2 Flags: srv must size, clt must
size, has return, */
/* 0x3, /* 3 */

/* Parameter txn_in */

/* 50 */      NdrFcShort( 0x8b ), /* Flags: must size, must free, in,
by val, */
/* 52 */      NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
/* 54 */      NdrFcShort( 0x3e2 ), /* Type Offset=994 */

/* Parameter txn_out */

/* 56 */      NdrFcShort( 0x4113 ), /* Flags: must size, must free,
out, simple ref, srv alloc size=16 */
/* 58 */      NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
/* 60 */      NdrFcShort( 0x3f4 ), /* Type Offset=1012 */

/* Return value */

/* 62 */      NdrFcShort( 0x70 ), /* Flags: out, return, base type,
*/
/* 64 */      NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
/* 66 */      0x8, /* FC_LONG */
/* 0 */

/* Procedure Delivery */

/* 68 */      0x33, /* FC_AUTO_HANDLE */
/* 0x6c, /* Old Flags: object, 0i2 */
/* 70 */      NdrFcLong( 0x0 ), /* 0 */
/* 74 */      NdrFcShort( 0x5 ), /* 5 */
/* 76 */      NdrFcShort( 0x1c ), /* x86 Stack size/offset = 28 */
/* 78 */      NdrFcShort( 0x0 ), /* 0 */
/* 80 */      NdrFcShort( 0x8 ), /* 8 */
/* 82 */      0x7, /* 0i2 Flags: srv must size, clt must
size, has return, */
/* 0x3, /* 3 */

/* Parameter txn_in */

/* 84 */      NdrFcShort( 0x8b ), /* Flags: must size, must free, in,
by val, */
/* 86 */      NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
/* 88 */      NdrFcShort( 0x3e2 ), /* Type Offset=994 */

/* Parameter txn_out */

```

```

/* 90 */      NdrFcShort( 0x4113 ), /* Flags: must size, must free,
out, simple ref, srv alloc size=16 */
/* 92 */      NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
/* 94 */      NdrFcShort( 0x3f4 ), /* Type Offset=1012 */

/* Return value */

/* 96 */      NdrFcShort( 0x70 ), /* Flags: out, return, base type,
*/
/* 98 */      NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
/* 100 */      0x8, /* FC_LONG */
/* 0 */

/* Procedure StockLevel */

/* 102 */      0x33, /* FC_AUTO_HANDLE */
/* 0x6c, /* Old Flags: object, 0i2 */
/* 104 */      NdrFcLong( 0x0 ), /* 0 */
/* 108 */      NdrFcShort( 0x6 ), /* 6 */
/* 110 */      NdrFcShort( 0x1c ), /* x86 Stack size/offset = 28 */
/* 112 */      NdrFcShort( 0x0 ), /* 0 */
/* 114 */      NdrFcShort( 0x8 ), /* 8 */
/* 116 */      0x7, /* 0i2 Flags: srv must size, clt must
size, has return, */
/* 0x3, /* 3 */

/* Parameter txn_in */

/* 118 */      NdrFcShort( 0x8b ), /* Flags: must size, must free, in,
by val, */
/* 120 */      NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
/* 122 */      NdrFcShort( 0x3e2 ), /* Type Offset=994 */

/* Parameter txn_out */

/* 124 */      NdrFcShort( 0x4113 ), /* Flags: must size, must free,
out, simple ref, srv alloc size=16 */
/* 126 */      NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
/* 128 */      NdrFcShort( 0x3f4 ), /* Type Offset=1012 */

/* Return value */

/* 130 */      NdrFcShort( 0x70 ), /* Flags: out, return, base type,
*/
/* 132 */      NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
/* 134 */      0x8, /* FC_LONG */
/* 0 */

/* Procedure OrderStatus */

/* 136 */      0x33, /* FC_AUTO_HANDLE */
/* 0x6c, /* Old Flags: object, 0i2 */
/* 138 */      NdrFcLong( 0x0 ), /* 0 */
/* 142 */      NdrFcShort( 0x7 ), /* 7 */
/* 144 */      NdrFcShort( 0x1c ), /* x86 Stack size/offset = 28 */
/* 146 */      NdrFcShort( 0x0 ), /* 0 */
/* 148 */      NdrFcShort( 0x8 ), /* 8 */
/* 150 */      0x7, /* 0i2 Flags: srv must size, clt must
size, has return, */
/* 0x3, /* 3 */

/* Parameter txn_in */

```

```

/* 152 */      NdrFcShort( 0x8b ), /* Flags: must size, must free, in,
by val, */
/* 154 */      NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
/* 156 */      NdrFcShort( 0x3e2 ), /* Type Offset=994 */

/* Parameter txn_out */

/* 158 */      NdrFcShort( 0x4113 ), /* Flags: must size, must free,
out, simple ref, srv alloc size=16 */
/* 160 */      NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
/* 162 */      NdrFcShort( 0x3f4 ), /* Type Offset=1012 */

/* Return value */

/* 164 */      NdrFcShort( 0x70 ), /* Flags: out, return, base type,
*/
/* 166 */      NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
/* 168 */      0x8, /* FC_LONG */
/* 168 */      0x0, /* 0 */

/* Procedure CallSetComplete */

/* 170 */      0x33, /* FC_AUTO_HANDLE */
/* 170 */      0x6c, /* Old Flags: object, 0i2 */
/* 172 */      NdrFcLong( 0x0 ), /* 0 */
/* 176 */      NdrFcShort( 0x8 ), /* 8 */
/* 178 */      NdrFcShort( 0x8 ), /* x86 Stack size/offset = 8 */
/* 180 */      NdrFcShort( 0x0 ), /* 0 */
/* 182 */      NdrFcShort( 0x8 ), /* 8 */
/* 184 */      0x4, /* 0i2 Flags: has return, */
/* 184 */      0x1, /* 1 */

/* Return value */

/* 186 */      NdrFcShort( 0x70 ), /* Flags: out, return, base type,
*/
/* 188 */      NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
/* 190 */      0x8, /* FC_LONG */
/* 190 */      0x0, /* 0 */

/* 190 */      0x0

};

static const MIDL_TYPE_FORMAT_STRING __MIDL_TypeFormatString =
{
    0,
    {
        NdrFcShort( 0x0 ), /* 0 */

        0x12, 0x0, /* FC_UP */
        NdrFcShort( 0x3ca ), /* Offset= 970 (974) */

        0x2b, /* FC_NON_ENCAPSULATED_UNION */
        0x9, /* FC_ULONG */

        0x7, /* Corr desc: FC_USHORT */
        0x0, /* 0 */

        NdrFcShort( 0xffff8 ), /* -8 */
        NdrFcShort( 0x2 ), /* Offset= 2 (14) */
        NdrFcShort( 0x10 ), /* 16 */
        NdrFcShort( 0x2f ), /* 47 */
        NdrFcLong( 0x14 ), /* 20 */
        NdrFcShort( 0x800b ), /* Simple arm type: FC_HYPER */

```

```

/* 24 */      NdrFcLong( 0x3 ), /* 3 */
/* 28 */      NdrFcShort( 0x8008 ), /* Simple arm type: FC_LONG */
/* 30 */      NdrFcLong( 0x11 ), /* 17 */
/* 34 */      NdrFcShort( 0x8001 ), /* Simple arm type: FC_BYTE */
/* 36 */      NdrFcLong( 0x2 ), /* 2 */
/* 40 */      NdrFcShort( 0x8006 ), /* Simple arm type: FC_SHORT */
/* 42 */      NdrFcLong( 0x4 ), /* 4 */
/* 46 */      NdrFcShort( 0x800a ), /* Simple arm type: FC_FLOAT */
/* 48 */      NdrFcLong( 0x5 ), /* 5 */
/* 52 */      NdrFcShort( 0x800c ), /* Simple arm type: FC_DOUBLE */
/* 54 */      NdrFcLong( 0xb ), /* 11 */
/* 58 */      NdrFcShort( 0x8006 ), /* Simple arm type: FC_SHORT */
/* 60 */      NdrFcLong( 0xa ), /* 10 */
/* 64 */      NdrFcShort( 0x8008 ), /* Simple arm type: FC_LONG */
/* 66 */      NdrFcLong( 0x6 ), /* 6 */
/* 70 */      NdrFcShort( 0xe8 ), /* Offset= 232 (302) */
/* 72 */      NdrFcLong( 0x7 ), /* 7 */
/* 76 */      NdrFcShort( 0x800c ), /* Simple arm type: FC_DOUBLE */
/* 78 */      NdrFcLong( 0x8 ), /* 8 */
/* 82 */      NdrFcShort( 0xe2 ), /* Offset= 226 (308) */
/* 84 */      NdrFcLong( 0xd ), /* 13 */
/* 88 */      NdrFcShort( 0xf4 ), /* Offset= 244 (332) */
/* 90 */      NdrFcLong( 0x9 ), /* 9 */
/* 94 */      NdrFcShort( 0x100 ), /* Offset= 256 (350) */
/* 96 */      NdrFcLong( 0x2000 ), /* 8192 */
/* 100 */      NdrFcShort( 0x10c ), /* Offset= 268 (368) */
/* 102 */      NdrFcLong( 0x24 ), /* 36 */
/* 106 */      NdrFcShort( 0x31a ), /* Offset= 794 (900) */
/* 108 */      NdrFcLong( 0x4024 ), /* 16420 */
/* 112 */      NdrFcShort( 0x314 ), /* Offset= 788 (900) */
/* 114 */      NdrFcLong( 0x4011 ), /* 16401 */
/* 118 */      NdrFcShort( 0x312 ), /* Offset= 786 (904) */
/* 120 */      NdrFcLong( 0x4002 ), /* 16386 */
/* 124 */      NdrFcShort( 0x310 ), /* Offset= 784 (908) */
/* 126 */      NdrFcLong( 0x4003 ), /* 16387 */
/* 130 */      NdrFcShort( 0x30e ), /* Offset= 782 (912) */
/* 132 */      NdrFcLong( 0x4014 ), /* 16404 */
/* 136 */      NdrFcShort( 0x30c ), /* Offset= 780 (916) */
/* 138 */      NdrFcLong( 0x4004 ), /* 16388 */
/* 142 */      NdrFcShort( 0x30a ), /* Offset= 778 (920) */
/* 144 */      NdrFcLong( 0x4005 ), /* 16389 */
/* 148 */      NdrFcShort( 0x308 ), /* Offset= 776 (924) */
/* 150 */      NdrFcLong( 0x400b ), /* 16395 */
/* 154 */      NdrFcShort( 0x2f2 ), /* Offset= 754 (908) */
/* 156 */      NdrFcLong( 0x400a ), /* 16394 */
/* 160 */      NdrFcShort( 0x2f0 ), /* Offset= 752 (912) */
/* 162 */      NdrFcLong( 0x4006 ), /* 16390 */
/* 166 */      NdrFcShort( 0x2fa ), /* Offset= 762 (928) */
/* 168 */      NdrFcLong( 0x4007 ), /* 16391 */
/* 172 */      NdrFcShort( 0x2f0 ), /* Offset= 752 (924) */
/* 174 */      NdrFcLong( 0x4008 ), /* 16392 */
/* 178 */      NdrFcShort( 0x2f2 ), /* Offset= 754 (932) */
/* 180 */      NdrFcLong( 0x400d ), /* 16397 */
/* 184 */      NdrFcShort( 0x2f0 ), /* Offset= 752 (936) */
/* 186 */      NdrFcLong( 0x4009 ), /* 16393 */
/* 190 */      NdrFcShort( 0x2ee ), /* Offset= 750 (940) */
/* 192 */      NdrFcLong( 0x6000 ), /* 24576 */
/* 196 */      NdrFcShort( 0x2ec ), /* Offset= 748 (944) */
/* 198 */      NdrFcLong( 0x400c ), /* 16396 */
/* 202 */      NdrFcShort( 0x2ea ), /* Offset= 746 (948) */
/* 204 */      NdrFcLong( 0x10 ), /* 16 */
/* 208 */      NdrFcShort( 0x8002 ), /* Simple arm type: FC_CHAR */
/* 210 */      NdrFcLong( 0x12 ), /* 18 */

```

```

/* 214 */ NdrFcShort( 0x8006 ), /* Simple arm type: FC_SHORT */
/* 216 */ NdrFcLong( 0x13 ), /* 19 */
/* 220 */ NdrFcShort( 0x8008 ), /* Simple arm type: FC_LONG */
/* 222 */ NdrFcLong( 0x15 ), /* 21 */
/* 226 */ NdrFcShort( 0x800b ), /* Simple arm type: FC_HYPER */
/* 228 */ NdrFcLong( 0x16 ), /* 22 */
/* 232 */ NdrFcShort( 0x8008 ), /* Simple arm type: FC_LONG */
/* 234 */ NdrFcLong( 0x17 ), /* 23 */
/* 238 */ NdrFcShort( 0x8008 ), /* Simple arm type: FC_LONG */
/* 240 */ NdrFcLong( 0xe ), /* 14 */
/* 244 */ NdrFcShort( 0x2c8 ), /* Offset= 712 (956) */
/* 246 */ NdrFcLong( 0x400e ), /* 16398 */
/* 250 */ NdrFcShort( 0x2cc ), /* Offset= 716 (966) */
/* 252 */ NdrFcLong( 0x4010 ), /* 16400 */
/* 256 */ NdrFcShort( 0x2ca ), /* Offset= 714 (970) */
/* 258 */ NdrFcLong( 0x4012 ), /* 16402 */
/* 262 */ NdrFcShort( 0x286 ), /* Offset= 646 (908) */
/* 264 */ NdrFcLong( 0x4013 ), /* 16403 */
/* 268 */ NdrFcShort( 0x284 ), /* Offset= 644 (912) */
/* 270 */ NdrFcLong( 0x4015 ), /* 16405 */
/* 274 */ NdrFcShort( 0x282 ), /* Offset= 642 (916) */
/* 276 */ NdrFcLong( 0x4016 ), /* 16406 */
/* 280 */ NdrFcShort( 0x278 ), /* Offset= 632 (912) */
/* 282 */ NdrFcLong( 0x4017 ), /* 16407 */
/* 286 */ NdrFcShort( 0x272 ), /* Offset= 626 (912) */
/* 288 */ NdrFcLong( 0x0 ), /* 0 */
/* 292 */ NdrFcShort( 0x0 ), /* Offset= 0 (292) */
/* 294 */ NdrFcLong( 0x1 ), /* 1 */
/* 298 */ NdrFcShort( 0x0 ), /* Offset= 0 (298) */
/* 300 */ NdrFcShort( 0xffffffff ), /* Offset= -1 (299) */
/* 302 */

    0x15, /* FC_STRUCT */
    0x7, /* 7 */
/* 304 */ NdrFcShort( 0x8 ), /* 8 */
/* 306 */ 0xb, /* FC_HYPER */
    0x5b, /* FC_END */

/* 308 */

    0x12, 0x0, /* FC_UP */
/* 310 */ NdrFcShort( 0xc ), /* Offset= 12 (322) */
/* 312 */

    0x1b, /* FC_CARRAY */
    0x1, /* 1 */
/* 314 */ NdrFcShort( 0x2 ), /* 2 */
/* 316 */ 0x9, /* Corr desc: FC_ULONG */
    0x0, /* */
/* 318 */ NdrFcShort( 0xfffc ), /* -4 */
/* 320 */ 0x6, /* FC_SHORT */
    0x5b, /* FC_END */

/* 322 */

    0x17, /* FC_CSTRUCT */
    0x3, /* 3 */
/* 324 */ NdrFcShort( 0x8 ), /* 8 */
/* 326 */ NdrFcShort( 0xffffffff2 ), /* Offset= -14 (312) */
/* 328 */ 0x8, /* FC_LONG */
    0x8, /* FC_LONG */
/* 330 */ 0x5c, /* FC_PAD */
    0x5b, /* FC_END */

/* 332 */

    0x2f, /* FC_IP */
    0x5a, /* FC_CONSTANT_IID */
/* 334 */ NdrFcLong( 0x0 ), /* 0 */
/* 338 */ NdrFcShort( 0x0 ), /* 0 */
/* 340 */ NdrFcShort( 0x0 ), /* 0 */

```

```

/* 342 */
/* 344 */
/* 346 */
/* 348 */
/* 350 */
/* 352 */
/* 356 */
/* 358 */
/* 360 */
/* 362 */
/* 364 */
/* 366 */
/* 368 */
/* 370 */
/* 372 */
/* 374 */
/* 376 */
/* 378 */
/* 380 */
/* 382 */
/* 386 */
/* 388 */
/* 392 */
/* 394 */
/* 398 */
/* 400 */
/* 404 */
/* 406 */
/* 410 */
/* 412 */
/* 416 */
/* 418 */
/* 422 */
/* 424 */
/* 428 */
/* 430 */
/* 434 */
/* 436 */
/* 440 */
/* 442 */
/* 444 */
/* 446 */
/* 448 */
/* 450 */
/* 452 */

```

```

0xc0, /* 192 */
    0x0, /* 0 */
    0x0, /* 0 */
    0x0, /* 0 */
    0x0, /* 0 */
    0x46, /* 70 */
    0x2f, /* FC_IP */
    0x5a, /* FC_CONSTANT_IID */
NdrFcLong( 0x20400 ), /* 132096 */
NdrFcShort( 0x0 ), /* 0 */
NdrFcShort( 0x0 ), /* 0 */
0xc0, /* 192 */
    0x0, /* 0 */
    0x0, /* 0 */
    0x0, /* 0 */
    0x0, /* 0 */
    0x46, /* 70 */
    0x12, 0x10, /* FC_UP [pointer_deref] */
NdrFcShort( 0x2 ), /* Offset= 2 (372) */
    0x12, 0x0, /* FC_UP */
NdrFcShort( 0x1fc ), /* Offset= 508 (882) */
    0x2a, /* FC_ENCAPSULATED_UNION */
    0x49, /* 73 */
NdrFcShort( 0x18 ), /* 24 */
NdrFcShort( 0xa ), /* 10 */
NdrFcLong( 0x8 ), /* 8 */
NdrFcShort( 0x58 ), /* Offset= 88 (474) */
NdrFcLong( 0xd ), /* 13 */
NdrFcShort( 0x78 ), /* Offset= 120 (512) */
NdrFcLong( 0x9 ), /* 9 */
NdrFcShort( 0x94 ), /* Offset= 148 (546) */
NdrFcLong( 0xc ), /* 12 */
NdrFcShort( 0xbc ), /* Offset= 188 (592) */
NdrFcLong( 0x24 ), /* 36 */
NdrFcShort( 0x114 ), /* Offset= 276 (686) */
NdrFcLong( 0x800d ), /* 32781 */
NdrFcShort( 0x130 ), /* Offset= 304 (720) */
NdrFcLong( 0x10 ), /* 16 */
NdrFcShort( 0x148 ), /* Offset= 328 (750) */
NdrFcLong( 0x2 ), /* 2 */
NdrFcShort( 0x160 ), /* Offset= 352 (780) */
NdrFcLong( 0x3 ), /* 3 */
NdrFcShort( 0x178 ), /* Offset= 376 (810) */
NdrFcLong( 0x14 ), /* 20 */
NdrFcShort( 0x190 ), /* Offset= 400 (840) */
NdrFcShort( 0xffffffff ), /* Offset= -1 (441) */
    0x1b, /* FC_CARRAY */
    0x3, /* 3 */
NdrFcShort( 0x4 ), /* 4 */
0x19, /* Corr desc: field pointer, FC_ULONG */
    0x0, /* */
NdrFcShort( 0x0 ), /* 0 */
    0x4b, /* FC_PP */

```

```

/* 454 */          0x5c,          /* FC_PAD */          /* 528 */
/* 456 */          0x48,          /* FC_VARIABLE_REPEAT */
/* 458 */          0x49,          /* FC_FIXED_OFFSET */          /* 530 */
/* 460 */          NdrFcShort( 0x4 ), /* 4 */          /* 532 */
/* 462 */          NdrFcShort( 0x0 ), /* 0 */
/* 464 */          NdrFcShort( 0x1 ), /* 1 */
/* 466 */          NdrFcShort( 0x0 ), /* 0 */
/* 468 */          NdrFcShort( 0x0 ), /* 0 */
/* 470 */          0x12, 0x0, /* FC_UP */
/* 472 */          NdrFcShort( 0xfffff6e ), /* Offset= -146 (322) */          /* 542 */
/* 474 */          0x5b,          /* FC_END */          /* 544 */
/* 476 */          0x8,          /* FC_LONG */          /* 546 */
/* 478 */          0x5c,          /* FC_PAD */
/* 480 */          0x5b,          /* FC_END */          /* 548 */
/* 482 */          0x16,          /* FC_PSTRUCT */          /* 550 */
/* 484 */          0x3,          /* 3 */          /* 552 */
/* 486 */          NdrFcShort( 0x8 ), /* 8 */          /* 554 */
/* 488 */          0x4b,          /* FC_PP */          /* 556 */
/* 490 */          0x5c,          /* FC_PAD */          /* 558 */
/* 492 */          0x46,          /* FC_NO_REPEAT */          /* 560 */
/* 494 */          0x5c,          /* FC_PAD */          /* 562 */
/* 496 */          NdrFcShort( 0x4 ), /* 4 */
/* 498 */          NdrFcShort( 0x4 ), /* 4 */
/* 500 */          0x11, 0x0, /* FC_RP */
/* 502 */          NdrFcShort( 0xfffffd4 ), /* Offset= -44 (444) */
/* 504 */          0x5b,          /* FC_END */
/* 506 */          0x8,          /* FC_LONG */
/* 508 */          0x5b,          /* FC_END */          /* 572 */
/* 510 */          0x21,          /* FC_BOGUS_ARRAY */
/* 512 */          0x3,          /* 3 */
/* 514 */          NdrFcShort( 0x0 ), /* 0 */
/* 516 */          0x19, /* Corr desc: field pointer, FC_ULONG */
/* 518 */          0x0,          /* 0 */
/* 520 */          NdrFcShort( 0x0 ), /* 0 */
/* 522 */          NdrFcLong( 0xffffffff ), /* -1 */
/* 524 */          0x4c,          /* FC_EMBEDDED_COMPLEX */
/* 526 */          0x0,          /* 0 */
/* 528 */          NdrFcShort( 0xfffff50 ), /* Offset= -176 (332) */
/* 530 */          0x5c,          /* FC_PAD */
/* 532 */          0x5b,          /* FC_END */
/* 534 */          0x1a,          /* FC_BOGUS_STRUCT */
/* 536 */          0x3,          /* 3 */
/* 538 */          NdrFcShort( 0x8 ), /* 8 */
/* 540 */          NdrFcShort( 0x0 ), /* 0 */
/* 542 */          NdrFcShort( 0x6 ), /* Offset= 6 (524) */
/* 544 */          0x8,          /* FC_LONG */
/* 546 */          0x36,          /* FC_POINTER */
/* 548 */          0x5c,          /* FC_PAD */
/* 550 */          0x5b,          /* FC_END */
/* 552 */          0x11, 0x0, /* FC_RP */
/* 554 */          NdrFcShort( 0xffffffe0 ), /* Offset= -32 (494) */
/* 556 */          0x21,          /* FC_BOGUS_ARRAY */
/* 558 */          0x3,          /* 3 */
/* 560 */          NdrFcShort( 0x0 ), /* 0 */
/* 562 */          0x19, /* Corr desc: field pointer, FC_ULONG */
/* 564 */          0x0,          /* 0 */
/* 566 */          NdrFcShort( 0x0 ), /* 0 */
/* 568 */          0x4b,          /* FC_PP */
/* 570 */          0x5c,          /* FC_PAD */
/* 572 */          0x48,          /* FC_VARIABLE_REPEAT */
/* 574 */          0x49,          /* FC_FIXED_OFFSET */
/* 576 */          NdrFcShort( 0x4 ), /* 4 */
/* 578 */          NdrFcShort( 0x0 ), /* 0 */
/* 580 */          NdrFcShort( 0x1 ), /* 1 */
/* 582 */          NdrFcShort( 0x0 ), /* 0 */
/* 584 */          NdrFcShort( 0x0 ), /* 0 */
/* 586 */          0x12, 0x0, /* FC_UP */
/* 588 */          NdrFcShort( 0x184 ), /* Offset= 388 (974) */
/* 590 */          0x5b,          /* FC_END */
/* 592 */          0x8,          /* FC_LONG */
/* 594 */          0x5c,          /* FC_PAD */
/* 596 */          0x5b,          /* FC_END */
/* 598 */          0x1a,          /* FC_BOGUS_STRUCT */
/* 600 */          0x3,          /* 3 */
/* 602 */          NdrFcShort( 0x8 ), /* 8 */
/* 604 */          NdrFcShort( 0x0 ), /* 0 */
/* 606 */          NdrFcShort( 0x6 ), /* Offset= 6 (604) */
/* 608 */          0x8,          /* FC_LONG */
/* 610 */          0x36,          /* FC_POINTER */
/* 612 */          0x5c,          /* FC_PAD */
/* 614 */          0x5b,          /* FC_END */

```

```

/* 606 */      0x11, 0x0, /* FC_RP */
/* 608 */      NdrFcShort( 0xffffffffd4 ), /* Offset= -44 (562) */

/* 610 */      0x2f, /* FC_IP */
/* 614 */      0x5a, /* FC_CONSTANT_IID */
/* 616 */      NdrFcLong( 0x2f ), /* 47 */
/* 618 */      NdrFcShort( 0x0 ), /* 0 */
/* 620 */      0xc0, /* 192 */
/* 622 */      0x0, /* 0 */
/* 624 */      0x0, /* 0 */
/* 626 */      0x0, /* 0 */
/* 628 */      0x0, /* 0 */
/* 630 */      0x0, /* 0 */
/* 632 */      0x0, /* 0 */
/* 634 */      0x0, /* 0 */
/* 636 */      0x46, /* 70 */

/* 638 */      0x1b, /* FC_CARRAY */
/* 640 */      0x0, /* 0 */
/* 642 */      NdrFcShort( 0x1 ), /* 1 */
/* 644 */      0x19, /* Corr desc: field pointer, FC_ULONG */
/* 646 */      0x0, /* 0 */
/* 648 */      NdrFcShort( 0x4 ), /* 4 */
/* 650 */      0x1, /* FC_BYTE */
/* 652 */      0x5b, /* FC_END */

/* 654 */      0x1a, /* FC_BOGUS_STRUCT */
/* 656 */      0x3, /* 3 */
/* 658 */      NdrFcShort( 0x10 ), /* 16 */
/* 660 */      NdrFcShort( 0x0 ), /* 0 */
/* 662 */      NdrFcShort( 0xa ), /* Offset= 10 (652) */
/* 664 */      0x8, /* FC_LONG */
/* 666 */      0x8, /* FC_LONG */
/* 668 */      0x4c, /* FC_EMBEDDED_COMPLEX */
/* 670 */      0x0, /* 0 */
/* 672 */      NdrFcShort( 0xffffffffd8 ), /* Offset= -40 (608) */
/* 674 */      0x36, /* FC_POINTER */
/* 676 */      0x5b, /* FC_END */

/* 678 */      0x12, 0x0, /* FC_UP */
/* 680 */      NdrFcShort( 0xffffffffe4 ), /* Offset= -28 (626) */
/* 682 */      0x1b, /* FC_CARRAY */
/* 684 */      0x3, /* 3 */
/* 686 */      NdrFcShort( 0x4 ), /* 4 */
/* 688 */      0x19, /* Corr desc: field pointer, FC_ULONG */
/* 690 */      0x0, /* 0 */
/* 692 */      NdrFcShort( 0x0 ), /* 0 */
/* 694 */      0x4b, /* FC_PP */
/* 696 */      0x5c, /* FC_PAD */

/* 698 */      0x48, /* FC_VARIABLE_REPEAT */
/* 700 */      0x49, /* FC_FIXED_OFFSET */
/* 702 */      NdrFcShort( 0x4 ), /* 4 */
/* 704 */      NdrFcShort( 0x0 ), /* 0 */
/* 706 */      NdrFcShort( 0x1 ), /* 1 */
/* 708 */      NdrFcShort( 0x0 ), /* 0 */
/* 710 */      NdrFcShort( 0x0 ), /* 0 */
/* 712 */      0x12, 0x0, /* FC_UP */
/* 714 */      NdrFcShort( 0xffffffffd4 ), /* Offset= -44 (636) */
/* 716 */      0x5b, /* FC_END */

/* 718 */      0x8, /* FC_LONG */
/* 720 */      0x5c, /* FC_PAD */
/* 722 */      0x5b, /* FC_END */

/* 724 */      0x1a, /* FC_BOGUS_STRUCT */
/* 726 */      0x3, /* 3 */
/* 728 */      NdrFcShort( 0x10 ), /* 16 */
/* 730 */      0x8, /* FC_LONG */
/* 732 */      0x6, /* FC_SHORT */
/* 734 */      0x4c, /* FC_EMBEDDED_COMPLEX */
/* 736 */      0x0, /* 0 */
/* 738 */      NdrFcShort( 0xfffffffff1 ), /* Offset= -15 (702) */
/* 740 */      0x5b, /* FC_END */

/* 742 */      0x1a, /* FC_BOGUS_STRUCT */
/* 744 */      0x3, /* 3 */
/* 746 */      NdrFcShort( 0x18 ), /* 24 */
/* 748 */      NdrFcShort( 0x0 ), /* 0 */
/* 750 */      NdrFcShort( 0xa ), /* Offset= 10 (736) */
/* 752 */      0x8, /* FC_LONG */
/* 754 */      0x36, /* FC_POINTER */
/* 756 */      0x4c, /* FC_EMBEDDED_COMPLEX */
/* 758 */      0x0, /* 0 */
/* 760 */      NdrFcShort( 0xffffffffe8 ), /* Offset= -24 (708) */
/* 762 */      0x5c, /* FC_PAD */
/* 764 */      0x5b, /* FC_END */

/* 766 */      0x11, 0x0, /* FC_RP */
/* 768 */      NdrFcShort( 0xfffffffff0c ), /* Offset= -244 (494) */
/* 770 */      0x1b, /* FC_CARRAY */
/* 772 */      0x0, /* 0 */
/* 774 */      NdrFcShort( 0x1 ), /* 1 */
/* 776 */      0x19, /* Corr desc: field pointer, FC_ULONG */
/* 778 */      0x0, /* 0 */
/* 780 */      NdrFcShort( 0x0 ), /* 0 */
/* 782 */      0x1, /* FC_BYTE */
/* 784 */      0x5b, /* FC_END */

/* 786 */      0x16, /* FC_PSTRUCT */
/* 788 */      0x3, /* 3 */

```

/* 752 */	NdrFcShort(0x8),	/* 8 */		0x46,	/* FC_NO_REPEAT */
/* 754 */				0x5c,	/* FC_PAD */
	0x4b,	/* FC_PP */	/* 818 */	NdrFcShort(0x4),	/* 4 */
/* 756 */	0x5c,	/* FC_PAD */	/* 820 */	NdrFcShort(0x4),	/* 4 */
			/* 822 */	0x12, 0x0,	/* FC_UP */
	0x46,	/* FC_NO_REPEAT */	/* 824 */	NdrFcShort(0xffffffe8),	/* Offset= -24 (800) */
	0x5c,	/* FC_PAD */	/* 826 */		
/* 758 */	NdrFcShort(0x4),	/* 4 */		0x5b,	/* FC_END */
/* 760 */	NdrFcShort(0x4),	/* 4 */			
/* 762 */	0x12, 0x0,	/* FC_UP */		0x8,	/* FC_LONG */
/* 764 */	NdrFcShort(0xffffffe8),	/* Offset= -24 (740) */	/* 828 */	0x8,	/* FC_LONG */
/* 766 */			/* 830 */	0x5b,	/* FC_END */
	0x5b,	/* FC_END */			
	0x8,	/* FC_LONG */		0x1b,	/* FC_CARRAY */
/* 768 */	0x8,	/* FC_LONG */	/* 832 */	0x7,	/* 7 */
/* 770 */	0x5b,	/* FC_END */	/* 834 */	NdrFcShort(0x8),	/* 8 */
				0x19,	/* Corr desc: field pointer, FC_ULONG */
	0x1b,	/* FC_CARRAY */	/* 836 */	0x0,	/* 0 */
/* 772 */	NdrFcShort(0x2),	/* 2 */	/* 838 */	NdrFcShort(0x0),	/* 0 */
/* 774 */	0x19,	/* Corr desc: field pointer, FC_ULONG */	/* 840 */	0xb,	/* FC_HYPER */
	0x0,	/* 0 */		0x5b,	/* FC_END */
/* 776 */	NdrFcShort(0x0),	/* 0 */			
/* 778 */	0x6,	/* FC_SHORT */	/* 842 */	0x16,	/* FC_PSTRUCT */
	0x5b,	/* FC_END */	/* 844 */	0x3,	/* 3 */
/* 780 */				NdrFcShort(0x8),	/* 8 */
	0x16,	/* FC_PSTRUCT */			
/* 782 */	0x3,	/* 3 */	/* 846 */	0x4b,	/* FC_PP */
/* 784 */	NdrFcShort(0x8),	/* 8 */		0x5c,	/* FC_PAD */
/* 786 */	0x4b,	/* FC_PP */	/* 848 */	0x46,	/* FC_NO_REPEAT */
	0x5c,	/* FC_PAD */	/* 850 */	0x5c,	/* FC_PAD */
			/* 852 */	NdrFcShort(0x4),	/* 4 */
	0x46,	/* FC_NO_REPEAT */	/* 854 */	NdrFcShort(0x4),	/* 4 */
/* 788 */	NdrFcShort(0x4),	/* 4 */	/* 856 */	0x12, 0x0,	/* FC_UP */
/* 790 */	NdrFcShort(0x4),	/* 4 */		NdrFcShort(0xffffffe8),	/* Offset= -24 (830) */
/* 792 */	0x12, 0x0,	/* FC_UP */			
/* 794 */	NdrFcShort(0xffffffe8),	/* Offset= -24 (770) */	/* 858 */	0x5b,	/* FC_END */
/* 796 */			/* 860 */		
	0x5b,	/* FC_END */		0x8,	/* FC_LONG */
	0x8,	/* FC_LONG */	/* 862 */	0x8,	/* FC_LONG */
/* 798 */	0x8,	/* FC_LONG */	/* 864 */	0x5b,	/* FC_END */
/* 800 */	0x5b,	/* FC_END */			
	0x1b,	/* FC_CARRAY */	/* 866 */	0x15,	/* FC_STRUCT */
/* 802 */	NdrFcShort(0x4),	/* 4 */	/* 868 */	0x3,	/* 3 */
/* 804 */	0x19,	/* Corr desc: field pointer, FC_ULONG */		NdrFcShort(0x8),	/* 8 */
	0x0,	/* 0 */		0x8,	/* FC_LONG */
/* 806 */	NdrFcShort(0x0),	/* 0 */	/* 870 */	0x5c,	/* FC_PAD */
/* 808 */	0x8,	/* FC_LONG */	/* 872 */	0x5b,	/* FC_END */
/* 810 */	0x5b,	/* FC_END */			
	0x16,	/* FC_PSTRUCT */	/* 874 */	0x1b,	/* FC_CARRAY */
/* 812 */	0x3,	/* 3 */	/* 876 */	0x3,	/* 3 */
/* 814 */	NdrFcShort(0x8),	/* 8 */	/* 878 */	NdrFcShort(0x8),	/* 8 */
			/* 880 */	0x7,	/* Corr desc: FC_USHORT */
	0x4b,	/* FC_PP */		0x0,	/* 0 */
/* 816 */	0x5c,	/* FC_PAD */	/* 882 */	NdrFcShort(0xfffd8),	/* -40 */
				0x4c,	/* FC_EMBEDDED_COMPLEX */
				0x0,	/* 0 */
				NdrFcShort(0xfffffee),	/* Offset= -18 (860) */
				0x5c,	/* FC_PAD */
				0x5b,	/* FC_END */
				0x1a,	/* FC_BOGUS_STRUCT */


```

/* 884 */      0x3,          /* 3 */
/* 886 */      NdrFcShort( 0x28 ), /* 40 */
/* 888 */      NdrFcShort( 0xfffffee ), /* Offset= -18 (868) */
/* 890 */      0x6,          /* FC_SHORT */
/* 892 */      0x6,          /* FC_SHORT */
/* 894 */      0x8,          /* FC_LONG */
/* 896 */      0x8,          /* FC_LONG */
/* 898 */      0x4c,         /* FC_EMBEDDED_COMPLEX */
/* 896 */      0x0,          /* 0 */
/* 898 */      NdrFcShort( 0xfffffdf8 ), /* Offset= -520 (376) */
/* 900 */      0x5c,         /* FC_PAD */
/* 900 */      0x5b,         /* FC_END */
/* 902 */      0x12, 0x0,     /* FC_UP */
/* 904 */      NdrFcShort( 0xfffffef6 ), /* Offset= -266 (636) */
/* 906 */      0x1,          /* FC_UP [simple_pointer] */
/* 908 */      0x5c,         /* FC_BYTE */
/* 910 */      0x12, 0x8,     /* FC_UP [simple_pointer] */
/* 912 */      0x6,          /* FC_SHORT */
/* 914 */      0x8,          /* FC_LONG */
/* 916 */      0x5c,         /* FC_PAD */
/* 918 */      0xb,          /* FC_UP [simple_pointer] */
/* 920 */      0x5c,         /* FC_HYPER */
/* 922 */      0xa,          /* FC_UP [simple_pointer] */
/* 924 */      0x5c,         /* FC_FLOAT */
/* 926 */      0xc,          /* FC_UP [simple_pointer] */
/* 928 */      0x5c,         /* FC_DOUBLE */
/* 930 */      0x12, 0x0,     /* FC_UP */
/* 932 */      NdrFcShort( 0xfffffd8c ), /* Offset= -628 (302) */
/* 934 */      0x12, 0x10,     /* FC_UP [pointer_deref] */
/* 936 */      NdrFcShort( 0xfffffd8e ), /* Offset= -626 (308) */
/* 938 */      0x12, 0x10,     /* FC_UP [pointer_deref] */
/* 940 */      NdrFcShort( 0xfffffda2 ), /* Offset= -606 (332) */
/* 942 */      0x12, 0x10,     /* FC_UP [pointer_deref] */
/* 944 */      NdrFcShort( 0xfffffdb0 ), /* Offset= -592 (350) */
/* 946 */      0x12, 0x10,     /* FC_UP [pointer_deref] */
/* 948 */      NdrFcShort( 0xfffffdb6 ), /* Offset= -578 (368) */
/* 950 */      0x12, 0x10,     /* FC_UP [pointer_deref] */
/* 952 */      NdrFcShort( 0x2 ), /* Offset= 2 (952) */
/* 954 */      0x12, 0x0,     /* FC_UP */
/* 956 */      NdrFcShort( 0x14 ), /* Offset= 20 (974) */
/* 956 */      0x15,         /* FC_STRUCT */

```

```

/* 958 */      0x7,          /* 7 */
/* 960 */      NdrFcShort( 0x10 ), /* 16 */
/* 962 */      0x6,          /* FC_SHORT */
/* 964 */      0x1,          /* FC_BYTE */
/* 966 */      0x1,          /* FC_BYTE */
/* 968 */      0xb,          /* FC_LONG */
/* 970 */      0x5b,         /* FC_HYPER */
/* 972 */      0x12, 0x0,     /* FC_UP */
/* 974 */      NdrFcShort( 0xfffffff4 ), /* Offset= -12 (956) */
/* 976 */      0x2,          /* FC_UP [simple_pointer] */
/* 978 */      0x5c,         /* FC_CHAR */
/* 980 */      0x1a,         /* FC_BOGUS_STRUCT */
/* 982 */      0x7,          /* 7 */
/* 984 */      NdrFcShort( 0x20 ), /* 32 */
/* 986 */      NdrFcShort( 0x0 ), /* 0 */
/* 988 */      NdrFcShort( 0x0 ), /* Offset= 0 (980) */
/* 990 */      0x8,          /* FC_LONG */
/* 992 */      0x6,          /* FC_LONG */
/* 994 */      0x6,          /* FC_SHORT */
/* 996 */      0x6,          /* FC_SHORT */
/* 998 */      0x4c,         /* FC_SHORT */
/* 1000 */      0x0,          /* FC_EMBEDDED_COMPLEX */
/* 1002 */      NdrFcShort( 0xfffffc28 ), /* Offset= -984 (6) */
/* 1004 */      0x5c,         /* FC_PAD */
/* 1006 */      0xb4,         /* FC_END */
/* 1008 */      0x83,         /* FC_USER_MARSHAL */
/* 1010 */      NdrFcShort( 0x0 ), /* 131 */
/* 1012 */      NdrFcShort( 0x10 ), /* 0 */
/* 1014 */      NdrFcShort( 0x0 ), /* 16 */
/* 1016 */      NdrFcShort( 0x0 ), /* 0 */
/* 1018 */      NdrFcShort( 0xfffffc18 ), /* Offset= -1000 (2) */
/* 1020 */      0x11, 0x4,     /* FC_UP [allocated_on_stack] */
/* 1022 */      NdrFcShort( 0x6 ), /* Offset= 6 (1012) */
/* 1024 */      0x13, 0x0,     /* FC_OP */
/* 1026 */      NdrFcShort( 0xffffffdc ), /* Offset= -36 (974) */
/* 1028 */      0xb4,         /* FC_USER_MARSHAL */
/* 1030 */      0x83,         /* 131 */
/* 1032 */      NdrFcShort( 0x0 ), /* 0 */
/* 1034 */      NdrFcShort( 0x10 ), /* 16 */
/* 1036 */      NdrFcShort( 0x0 ), /* 0 */
/* 1038 */      NdrFcShort( 0xfffffff4 ), /* Offset= -12 (1008) */
/* 1040 */      0x0
}
};

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

```

```

        ,VARIANT_UserFree
    };

/* Standard interface: __MIDL_itf_tpcc_com_ps_0000, ver. 0.0,
GUID={0x00000000,0x0000,0x0000,{0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00}}
*/

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

/* Object interface: ITPCC, ver. 0.0,
GUID={0xFEEE6AA2,0x84B1,0x11d2,{0xBA,0x47,0x00,0xC0,0x4F,0xBF,0xE0,0x8B}}
*/

#pragma code_seg(".orpc")
static const unsigned short ITPCC_FormatStringOffsetTable[] =
{
    0,
    34,
    68,
    102,
    136,
    170
};

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

static const MIDL_SERVER_INFO ITPCC_ServerInfo =
{
    &Object_StubDesc,
    0,
    __MIDL_ProcFormatString.Format,
    &ITPCC_FormatStringOffsetTable[-3],
    0,
    0,
    0,
    0,
    0
};
CINTERFACE_PROXY_VTABLE(9) _ITPCCProxyVtbl =
{
    &ITPCC_ProxyInfo,
    &IID_ITPCC,
    IUnknown_QueryInterface_Proxy,
    IUnknown_AddRef_Proxy,
    IUnknown_Release_Proxy ,

```

```

    (void *) (INT_PTR) -1 /* ITPCC::NewOrder */ ,
    (void *) (INT_PTR) -1 /* ITPCC::Payment */ ,
    (void *) (INT_PTR) -1 /* ITPCC::Delivery */ ,
    (void *) (INT_PTR) -1 /* ITPCC::StockLevel */ ,
    (void *) (INT_PTR) -1 /* ITPCC::OrderStatus */ ,
    (void *) (INT_PTR) -1 /* ITPCC::CallSetComplete */
};

const CInterfaceStubVtbl _ITPCCStubVtbl =
{
    &IID_ITPCC,
    &ITPCC_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 */
    0x20000, /* Ndr library version */
    0,
    0x600015b, /* MIDL Version 6.0.347 */
    0,
    UserMarshalRoutines,
    0, /* notify & notify_flag routine table */
    0x1, /* MIDL flag */
    0, /* cs routines */
    0, /* proxy/server info */
    0 /* Reserved5 */
};

const CInterfaceProxyVtbl * _tpcc_com_ps_ProxyVtblList[] =
{
    ( CInterfaceProxyVtbl *) &_ITPCCProxyVtbl,
    0
};

const CInterfaceStubVtbl * _tpcc_com_ps_StubVtblList[] =
{
    ( CInterfaceStubVtbl *) &_ITPCCStubVtbl,
    0
};

PCInterfaceName const _tpcc_com_ps_InterfaceNamesList[] =
{
    "ITPCC",
    0
};

#define _tpcc_com_ps_CHECK_IID(n)    IID_GENERIC_CHECK_IID( _tpcc_com_ps,
pIID, n)

```

```

int __stdcall _tpcc_com_ps_IID_Lookup( const IID * pIID, int * pIndex )
{
    if(!_tpcc_com_ps_CHECK_IID(0))
    {
        *pIndex = 0;
        return 1;
    }

    return 0;
}

const ExtendedProxyFileInfo tpcc_com_ps_ProxyFileInfo =
{
    (PCInterfaceProxyVtblList *) & _tpcc_com_ps_ProxyVtblList,
    (PCInterfaceStubVtblList *) & _tpcc_com_ps_StubVtblList,
    (const PCInterfaceName * ) & _tpcc_com_ps_InterfaceNamesList,
    0, // no delegation
    & _tpcc_com_ps_IID_Lookup,
    1,
    2,
    0, /* table of [async_uuid] interfaces */
    0, /* Filler1 */
    0, /* Filler2 */
    0 /* Filler3 */
};

#endif /* !defined(_M_IA64) && !defined(_M_AMD64)*/

#pragma warning( disable: 4049 ) /* more than 64k source lines */
/* this ALWAYS GENERATED file contains the proxy stub code */

/* File created by MIDL compiler version 6.00.0347 */
/* at Mon Nov 15 21:38:37 2004
*/
/* Compiler settings for .\src\tpcc_com_ps.idl:
    Oicf, W1, Zp8, env=win64 (32b run,appending)
    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(  )

#if defined(_M_IA64) || defined(_M_AMD64)
#define USE_STUBLESS_PROXY

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

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

```

```

#endif // __RPCPROXY_H_VERSION__

#include "tpcc_com_ps.h"

#define TYPE_FORMAT_STRING_SIZE 1003
#define PROC_FORMAT_STRING_SIZE 253
#define TRANSMIT_AS_TABLE_SIZE 0
#define WIRE_MARSHAL_TABLE_SIZE 1

typedef struct _MIDL_TYPE_FORMAT_STRING
{
    short Pad;
    unsigned char Format[ TYPE_FORMAT_STRING_SIZE ];
} MIDL_TYPE_FORMAT_STRING;

typedef struct _MIDL_PROC_FORMAT_STRING
{
    short Pad;
    unsigned char Format[ PROC_FORMAT_STRING_SIZE ];
} MIDL_PROC_FORMAT_STRING;

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 ITPCC_ServerInfo;
extern const MIDL_STUBLESS_PROXY_INFO ITPCC_ProxyInfo;

extern const USER_MARSHAL_ROUTINE_QUADRUPLE
UserMarshalRoutines[ WIRE_MARSHAL_TABLE_SIZE ];

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

static const MIDL_PROC_FORMAT_STRING __MIDL_ProcFormatString =
{
    {
        0,

        /* Procedure NewOrder */

        0x33, /* FC_AUTO_HANDLE */
        0x6c, /* Old Flags: object, Oi2 */
        /* 2 */ NdrFcLong( 0x0 ), /* 0 */
        /* 6 */ NdrFcShort( 0x3 ), /* 3 */
        /* 8 */ NdrFcShort( 0x30 ), /* ia64 Stack size/offset = 48 */
        /* 10 */ NdrFcShort( 0x0 ), /* 0 */
        /* 12 */ NdrFcShort( 0x8 ), /* 8 */
        /* 14 */ 0x47, /* Oi2 Flags: srv must size, clt must
size, has return, has ext, */
        0x3, /* 3 */
    }
}

```

```

/* 16 */      0xa,          /* 10 */
                0x7,          /* Ext Flags: new corr desc, clt
corr check, srv corr check, */
/* 18 */      NdrFcShort( 0x20 ), /* 32 */
/* 20 */      NdrFcShort( 0x20 ), /* 32 */
/* 22 */      NdrFcShort( 0x0 ), /* 0 */
/* 24 */      NdrFcShort( 0x0 ), /* 0 */

/* Parameter txn_in */

/* 26 */      NdrFcShort( 0x8b ), /* Flags: must size, must free, in,
by val, */
/* 28 */      NdrFcShort( 0x8 ), /* ia64 Stack size/offset = 8 */
/* 30 */      NdrFcShort( 0x3ce ), /* Type Offset=974 */

/* Parameter txn_out */

/* 32 */      NdrFcShort( 0x6113 ), /* Flags: must size, must free,
out, simple ref, srv alloc size=24 */
/* 34 */      NdrFcShort( 0x20 ), /* ia64 Stack size/offset = 32 */
/* 36 */      NdrFcShort( 0x3e0 ), /* Type Offset=992 */

/* Return value */

/* 38 */      NdrFcShort( 0x70 ), /* Flags: out, return, base type,
*/
/* 40 */      NdrFcShort( 0x28 ), /* ia64 Stack size/offset = 40 */
/* 42 */      0x8, /* FC_LONG */
                0x0, /* 0 */

/* Procedure Payment */

/* 44 */      0x33, /* FC_AUTO_HANDLE */
                0x6c, /* Old Flags: object, Oi2 */
/* 46 */      NdrFcLong( 0x0 ), /* 0 */
/* 50 */      NdrFcShort( 0x4 ), /* 4 */
/* 52 */      NdrFcShort( 0x30 ), /* ia64 Stack size/offset = 48 */
/* 54 */      NdrFcShort( 0x0 ), /* 0 */
/* 56 */      NdrFcShort( 0x8 ), /* 8 */
/* 58 */      0x47, /* Oi2 Flags: srv must size, clt must
size, has return, has ext, */
                0x3, /* 3 */
/* 60 */      0xa, /* 10 */
                0x7, /* Ext Flags: new corr desc, clt
corr check, srv corr check, */
/* 62 */      NdrFcShort( 0x20 ), /* 32 */
/* 64 */      NdrFcShort( 0x20 ), /* 32 */
/* 66 */      NdrFcShort( 0x0 ), /* 0 */
/* 68 */      NdrFcShort( 0x0 ), /* 0 */

/* Parameter txn_in */

/* 70 */      NdrFcShort( 0x8b ), /* Flags: must size, must free, in,
by val, */
/* 72 */      NdrFcShort( 0x8 ), /* ia64 Stack size/offset = 8 */
/* 74 */      NdrFcShort( 0x3ce ), /* Type Offset=974 */

/* Parameter txn_out */

/* 76 */      NdrFcShort( 0x6113 ), /* Flags: must size, must free,
out, simple ref, srv alloc size=24 */
/* 78 */      NdrFcShort( 0x20 ), /* ia64 Stack size/offset = 32 */
/* 80 */      NdrFcShort( 0x3e0 ), /* Type Offset=992 */

```

```

/* Return value */

/* 82 */      NdrFcShort( 0x70 ), /* Flags: out, return, base type,
*/
/* 84 */      NdrFcShort( 0x28 ), /* ia64 Stack size/offset = 40 */
/* 86 */      0x8, /* FC_LONG */
                0x0, /* 0 */

/* Procedure Delivery */

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

/* Parameter txn_in */

/* 114 */      NdrFcShort( 0x8b ), /* Flags: must size, must free, in,
by val, */
/* 116 */      NdrFcShort( 0x8 ), /* ia64 Stack size/offset = 8 */
/* 118 */      NdrFcShort( 0x3ce ), /* Type Offset=974 */

/* Parameter txn_out */

/* 120 */      NdrFcShort( 0x6113 ), /* Flags: must size, must free,
out, simple ref, srv alloc size=24 */
/* 122 */      NdrFcShort( 0x20 ), /* ia64 Stack size/offset = 32 */
/* 124 */      NdrFcShort( 0x3e0 ), /* Type Offset=992 */

/* Return value */

/* 126 */      NdrFcShort( 0x70 ), /* Flags: out, return, base type,
*/
/* 128 */      NdrFcShort( 0x28 ), /* ia64 Stack size/offset = 40 */
/* 130 */      0x8, /* FC_LONG */
                0x0, /* 0 */

/* Procedure StockLevel */

/* 132 */      0x33, /* FC_AUTO_HANDLE */
                0x6c, /* Old Flags: object, Oi2 */
/* 134 */      NdrFcLong( 0x0 ), /* 0 */
/* 138 */      NdrFcShort( 0x6 ), /* 6 */
/* 140 */      NdrFcShort( 0x30 ), /* ia64 Stack size/offset = 48 */
/* 142 */      NdrFcShort( 0x0 ), /* 0 */
/* 144 */      NdrFcShort( 0x8 ), /* 8 */
/* 146 */      0x47, /* Oi2 Flags: srv must size, clt must
size, has return, has ext, */
                0x3, /* 3 */

```

```

/* 148 */      0xa,          /* 10 */
                0x7,          /* Ext Flags: new corr desc, clt
corr check, srv corr check, */
/* 150 */      NdrFcShort( 0x20 ), /* 32 */
/* 152 */      NdrFcShort( 0x20 ), /* 32 */
/* 154 */      NdrFcShort( 0x0 ), /* 0 */
/* 156 */      NdrFcShort( 0x0 ), /* 0 */

/* Parameter txn_in */

/* 158 */      NdrFcShort( 0x8b ), /* Flags: must size, must free, in,
by val, */
/* 160 */      NdrFcShort( 0x8 ), /* ia64 Stack size/offset = 8 */
/* 162 */      NdrFcShort( 0x3ce ), /* Type Offset=974 */

/* Parameter txn_out */

/* 164 */      NdrFcShort( 0x6113 ), /* Flags: must size, must free,
out, simple ref, srv alloc size=24 */
/* 166 */      NdrFcShort( 0x20 ), /* ia64 Stack size/offset = 32 */
/* 168 */      NdrFcShort( 0x3e0 ), /* Type Offset=992 */

/* Return value */

/* 170 */      NdrFcShort( 0x70 ), /* Flags: out, return, base type,
*/
/* 172 */      NdrFcShort( 0x28 ), /* ia64 Stack size/offset = 40 */
/* 174 */      0x8, /* FC_LONG */
                0x0, /* 0 */

/* Procedure OrderStatus */

/* 176 */      0x33, /* FC_AUTO_HANDLE */
                0x6c, /* Old Flags: object, 0i2 */
/* 178 */      NdrFcLong( 0x0 ), /* 0 */
/* 182 */      NdrFcShort( 0x7 ), /* 7 */
/* 184 */      NdrFcShort( 0x30 ), /* ia64 Stack size/offset = 48 */
/* 186 */      NdrFcShort( 0x0 ), /* 0 */
/* 188 */      NdrFcShort( 0x8 ), /* 8 */
/* 190 */      0x47, /* 0i2 Flags: srv must size, clt must
size, has return, has ext, */
                0x3, /* 3 */
/* 192 */      0xa, /* 10 */
                0x7, /* Ext Flags: new corr desc, clt
corr check, srv corr check, */
/* 194 */      NdrFcShort( 0x20 ), /* 32 */
/* 196 */      NdrFcShort( 0x20 ), /* 32 */
/* 198 */      NdrFcShort( 0x0 ), /* 0 */
/* 200 */      NdrFcShort( 0x0 ), /* 0 */

/* Parameter txn_in */

/* 202 */      NdrFcShort( 0x8b ), /* Flags: must size, must free, in,
by val, */
/* 204 */      NdrFcShort( 0x8 ), /* ia64 Stack size/offset = 8 */
/* 206 */      NdrFcShort( 0x3ce ), /* Type Offset=974 */

/* Parameter txn_out */

/* 208 */      NdrFcShort( 0x6113 ), /* Flags: must size, must free,
out, simple ref, srv alloc size=24 */
/* 210 */      NdrFcShort( 0x20 ), /* ia64 Stack size/offset = 32 */
/* 212 */      NdrFcShort( 0x3e0 ), /* Type Offset=992 */

```

```

/* Return value */

/* 214 */      NdrFcShort( 0x70 ), /* Flags: out, return, base type,
*/
/* 216 */      NdrFcShort( 0x28 ), /* ia64 Stack size/offset = 40 */
/* 218 */      0x8, /* FC_LONG */
                0x0, /* 0 */

/* Procedure CallSetComplete */

/* 220 */      0x33, /* FC_AUTO_HANDLE */
                0x6c, /* Old Flags: object, 0i2 */
/* 222 */      NdrFcLong( 0x0 ), /* 0 */
/* 226 */      NdrFcShort( 0x8 ), /* 8 */
/* 228 */      NdrFcShort( 0x10 ), /* ia64 Stack size/offset = 16 */
/* 230 */      NdrFcShort( 0x0 ), /* 0 */
/* 232 */      NdrFcShort( 0x8 ), /* 8 */
/* 234 */      0x44, /* 0i2 Flags: has return, has ext, */
                0x1, /* 1 */
/* 236 */      0xa, /* 10 */
                0x1, /* Ext Flags: new corr desc, */
/* 238 */      NdrFcShort( 0x0 ), /* 0 */
/* 240 */      NdrFcShort( 0x0 ), /* 0 */
/* 242 */      NdrFcShort( 0x0 ), /* 0 */
/* 244 */      NdrFcShort( 0x0 ), /* 0 */

/* Return value */

/* 246 */      NdrFcShort( 0x70 ), /* Flags: out, return, base type,
*/
/* 248 */      NdrFcShort( 0x8 ), /* ia64 Stack size/offset = 8 */
/* 250 */      0x8, /* FC_LONG */
                0x0, /* 0 */
                0x0

};
}

static const MIDL_TYPE_FORMAT_STRING __MIDL_TypeFormatString =
{
    0,
    {
        NdrFcShort( 0x0 ), /* 0 */
/* 2 */      0x12, 0x0, /* FC_UP */
/* 4 */      NdrFcShort( 0x3b6 ), /* Offset= 950 (954) */
/* 6 */
        0x2b, /* FC_NON_ENCAPSULATED_UNION */
        0x9, /* FC_ULONG */
/* 8 */      0x7, /* Corr desc: FC_USHORT */
        0x0, /* */
/* 10 */      NdrFcShort( 0xffff8 ), /* -8 */
/* 12 */      NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 14 */      NdrFcShort( 0x2 ), /* Offset= 2 (16) */
/* 16 */      NdrFcShort( 0x10 ), /* 16 */
/* 18 */      NdrFcShort( 0x2f ), /* 47 */
/* 20 */      NdrFcLong( 0x14 ), /* 20 */
/* 24 */      NdrFcShort( 0x800b ), /* Simple arm type: FC_HYPER */
/* 26 */      NdrFcLong( 0x3 ), /* 3 */
/* 30 */      NdrFcShort( 0x8008 ), /* Simple arm type: FC_LONG */
/* 32 */      NdrFcLong( 0x11 ), /* 17 */
/* 36 */      NdrFcShort( 0x8001 ), /* Simple arm type: FC_BYTE */

```

```

/* 38 */      NdrFcLong( 0x2 ),      /* 2 */
/* 42 */      NdrFcShort( 0x8006 ), /* Simple arm type: FC_SHORT */
/* 44 */      NdrFcLong( 0x4 ),      /* 4 */
/* 48 */      NdrFcShort( 0x800a ), /* Simple arm type: FC_FLOAT */
/* 50 */      NdrFcLong( 0x5 ),      /* 5 */
/* 54 */      NdrFcShort( 0x800c ), /* Simple arm type: FC_DOUBLE */
/* 56 */      NdrFcLong( 0xb ),      /* 11 */
/* 60 */      NdrFcShort( 0x8006 ), /* Simple arm type: FC_SHORT */
/* 62 */      NdrFcLong( 0xa ),      /* 10 */
/* 66 */      NdrFcShort( 0x8008 ), /* Simple arm type: FC_LONG */
/* 68 */      NdrFcLong( 0x6 ),      /* 6 */
/* 72 */      NdrFcShort( 0xe8 ), /* Offset= 232 (304) */
/* 74 */      NdrFcLong( 0x7 ),      /* 7 */
/* 78 */      NdrFcShort( 0x800c ), /* Simple arm type: FC_DOUBLE */
/* 80 */      NdrFcLong( 0x8 ),      /* 8 */
/* 84 */      NdrFcShort( 0xe2 ), /* Offset= 226 (310) */
/* 86 */      NdrFcLong( 0xd ),      /* 13 */
/* 90 */      NdrFcShort( 0xf6 ), /* Offset= 246 (336) */
/* 92 */      NdrFcLong( 0x9 ),      /* 9 */
/* 96 */      NdrFcShort( 0x102 ), /* Offset= 258 (354) */
/* 98 */      NdrFcLong( 0x2000 ), /* 8192 */
/* 102 */     NdrFcShort( 0x10e ), /* Offset= 270 (372) */
/* 104 */     NdrFcLong( 0x24 ),      /* 36 */
/* 108 */     NdrFcShort( 0x304 ), /* Offset= 772 (880) */
/* 110 */     NdrFcLong( 0x4024 ), /* 16420 */
/* 114 */     NdrFcShort( 0x2fe ), /* Offset= 766 (880) */
/* 116 */     NdrFcLong( 0x4011 ), /* 16401 */
/* 120 */     NdrFcShort( 0x2fc ), /* Offset= 764 (884) */
/* 122 */     NdrFcLong( 0x4002 ), /* 16386 */
/* 126 */     NdrFcShort( 0x2fa ), /* Offset= 762 (888) */
/* 128 */     NdrFcLong( 0x4003 ), /* 16387 */
/* 132 */     NdrFcShort( 0x2f8 ), /* Offset= 760 (892) */
/* 134 */     NdrFcLong( 0x4014 ), /* 16404 */
/* 138 */     NdrFcShort( 0x2f6 ), /* Offset= 758 (896) */
/* 140 */     NdrFcLong( 0x4004 ), /* 16388 */
/* 144 */     NdrFcShort( 0x2f4 ), /* Offset= 756 (900) */
/* 146 */     NdrFcLong( 0x4005 ), /* 16389 */
/* 150 */     NdrFcShort( 0x2f2 ), /* Offset= 754 (904) */
/* 152 */     NdrFcLong( 0x400b ), /* 16395 */
/* 156 */     NdrFcShort( 0x2dc ), /* Offset= 732 (888) */
/* 158 */     NdrFcLong( 0x400a ), /* 16394 */
/* 162 */     NdrFcShort( 0x2da ), /* Offset= 730 (892) */
/* 164 */     NdrFcLong( 0x4006 ), /* 16390 */
/* 168 */     NdrFcShort( 0x2e4 ), /* Offset= 740 (908) */
/* 170 */     NdrFcLong( 0x4007 ), /* 16391 */
/* 174 */     NdrFcShort( 0x2da ), /* Offset= 730 (904) */
/* 176 */     NdrFcLong( 0x4008 ), /* 16392 */
/* 180 */     NdrFcShort( 0x2dc ), /* Offset= 732 (912) */
/* 182 */     NdrFcLong( 0x400d ), /* 16397 */
/* 186 */     NdrFcShort( 0x2da ), /* Offset= 730 (916) */
/* 188 */     NdrFcLong( 0x4009 ), /* 16393 */
/* 192 */     NdrFcShort( 0x2d8 ), /* Offset= 728 (920) */
/* 194 */     NdrFcLong( 0x6000 ), /* 24576 */
/* 198 */     NdrFcShort( 0x2d6 ), /* Offset= 726 (924) */
/* 200 */     NdrFcLong( 0x400c ), /* 16396 */
/* 204 */     NdrFcShort( 0x2d4 ), /* Offset= 724 (928) */
/* 206 */     NdrFcLong( 0x10 ),      /* 16 */
/* 210 */     NdrFcShort( 0x8002 ), /* Simple arm type: FC_CHAR */
/* 212 */     NdrFcLong( 0x12 ),      /* 18 */
/* 216 */     NdrFcShort( 0x8006 ), /* Simple arm type: FC_SHORT */
/* 218 */     NdrFcLong( 0x13 ),      /* 19 */
/* 222 */     NdrFcShort( 0x8008 ), /* Simple arm type: FC_LONG */
/* 224 */     NdrFcLong( 0x15 ),      /* 21 */

/* 228 */     NdrFcShort( 0x800b ), /* Simple arm type: FC_HYPER */
/* 230 */     NdrFcLong( 0x16 ),      /* 22 */
/* 234 */     NdrFcShort( 0x8008 ), /* Simple arm type: FC_LONG */
/* 236 */     NdrFcLong( 0x17 ),      /* 23 */
/* 240 */     NdrFcShort( 0x8008 ), /* Simple arm type: FC_LONG */
/* 242 */     NdrFcLong( 0xe ),      /* 14 */
/* 246 */     NdrFcShort( 0x2b2 ), /* Offset= 690 (936) */
/* 248 */     NdrFcLong( 0x400e ), /* 16398 */
/* 252 */     NdrFcShort( 0x2b6 ), /* Offset= 694 (946) */
/* 254 */     NdrFcLong( 0x4010 ), /* 16400 */
/* 258 */     NdrFcShort( 0x2b4 ), /* Offset= 692 (950) */
/* 260 */     NdrFcLong( 0x4012 ), /* 16402 */
/* 264 */     NdrFcShort( 0x270 ), /* Offset= 624 (888) */
/* 266 */     NdrFcLong( 0x4013 ), /* 16403 */
/* 270 */     NdrFcShort( 0x26e ), /* Offset= 622 (892) */
/* 272 */     NdrFcLong( 0x4015 ), /* 16405 */
/* 276 */     NdrFcShort( 0x26c ), /* Offset= 620 (896) */
/* 278 */     NdrFcLong( 0x4016 ), /* 16406 */
/* 282 */     NdrFcShort( 0x262 ), /* Offset= 610 (892) */
/* 284 */     NdrFcLong( 0x4017 ), /* 16407 */
/* 288 */     NdrFcShort( 0x25c ), /* Offset= 604 (892) */
/* 290 */     NdrFcLong( 0x0 ),      /* 0 */
/* 294 */     NdrFcShort( 0x0 ),      /* Offset= 0 (294) */
/* 296 */     NdrFcLong( 0x1 ),      /* 1 */
/* 300 */     NdrFcShort( 0x0 ),      /* Offset= 0 (300) */
/* 302 */     NdrFcShort( 0xffffffff ), /* Offset= -1 (301) */

/* 306 */     0x15, /* FC_STRUCT */
/* 308 */     0x7, /* 7 */
NdrFcShort( 0x8 ), /* 8 */
0xb, /* FC_HYPER */
/* 310 */     0x5b, /* FC_END */

/* 312 */     0x12, 0x0, /* FC_UP */
/* 314 */     NdrFcShort( 0xe ), /* Offset= 14 (326) */

/* 316 */     0x1b, /* FC_CARRAY */
/* 318 */     0x1, /* 1 */
NdrFcShort( 0x2 ), /* 2 */
0x9, /* Corr desc: FC_ULONG */
/* 320 */     0x0, /* -4 */
/* 322 */     NdrFcShort( 0xfffc ), /* Corr flags: early */
/* 324 */     0x6, /* FC_SHORT */
/* 326 */     0x5b, /* FC_END */

/* 328 */     0x17, /* FC_CSTRUCT */
/* 330 */     0x3, /* 3 */
/* 332 */     NdrFcShort( 0x8 ), /* 8 */
/* 334 */     NdrFcShort( 0xffffffff0 ), /* Offset= -16 (314) */
/* 336 */     0x8, /* FC_LONG */
/* 338 */     0x8, /* FC_LONG */
/* 340 */     0x5c, /* FC_PAD */
/* 342 */     0x5b, /* FC_END */

/* 344 */     0x2f, /* FC_IP */
/* 346 */     0x5a, /* FC_CONSTANT_IID */
/* 348 */     NdrFcLong( 0x0 ), /* 0 */
/* 350 */     NdrFcShort( 0x0 ), /* 0 */
/* 352 */     NdrFcShort( 0x0 ), /* 0 */
/* 354 */     0xc0, /* 192 */
/* 356 */     0x0, /* 0 */
/* 358 */     0x0, /* 0 */

```

```

/* 350 */      0x0,      /* 0 */
/* 352 */      0x0,      /* 0 */
/* 354 */      0x46,      /* 70 */
/* 356 */      0x2f,      /* FC_IP */
/* 360 */      0x5a,      /* FC_CONSTANT_IID */
/* 362 */      NdrFcLong( 0x20400 ), /* 132096 */
/* 364 */      NdrFcShort( 0x0 ), /* 0 */
/* 366 */      NdrFcShort( 0xc0 ), /* 192 */
/* 368 */      0x0,      /* 0 */
/* 370 */      0x0,      /* 0 */
/* 372 */      0x0,      /* 0 */
/* 374 */      0x0,      /* 0 */
/* 376 */      0x46,      /* 70 */
/* 378 */      0x12, 0x10, /* FC_UP [pointer_deref] */
/* 380 */      NdrFcShort( 0x2 ), /* Offset= 2 (376) */
/* 382 */      0x12, 0x0, /* FC_UP */
/* 384 */      NdrFcShort( 0x1e4 ), /* Offset= 484 (862) */
/* 386 */      0x2a,      /* FC_ENCAPSULATED_UNION */
/* 388 */      0x89,      /* 137 */
/* 390 */      NdrFcShort( 0x20 ), /* 32 */
/* 392 */      NdrFcShort( 0xa ), /* 10 */
/* 394 */      NdrFcLong( 0x8 ), /* 8 */
/* 396 */      NdrFcShort( 0x50 ), /* Offset= 80 (470) */
/* 398 */      NdrFcLong( 0xd ), /* 13 */
/* 400 */      NdrFcShort( 0x70 ), /* Offset= 112 (508) */
/* 402 */      NdrFcLong( 0x9 ), /* 9 */
/* 404 */      NdrFcShort( 0x90 ), /* Offset= 144 (546) */
/* 406 */      NdrFcLong( 0xc ), /* 12 */
/* 408 */      NdrFcShort( 0xb0 ), /* Offset= 176 (584) */
/* 410 */      NdrFcLong( 0x24 ), /* 36 */
/* 412 */      NdrFcShort( 0x102 ), /* Offset= 258 (672) */
/* 414 */      NdrFcLong( 0x800d ), /* 32781 */
/* 416 */      NdrFcShort( 0x11e ), /* Offset= 286 (706) */
/* 418 */      NdrFcLong( 0x10 ), /* 16 */
/* 420 */      NdrFcShort( 0x138 ), /* Offset= 312 (738) */
/* 422 */      NdrFcLong( 0x2 ), /* 2 */
/* 424 */      NdrFcShort( 0x14e ), /* Offset= 334 (766) */
/* 426 */      NdrFcLong( 0x3 ), /* 3 */
/* 428 */      NdrFcShort( 0x164 ), /* Offset= 356 (794) */
/* 430 */      NdrFcLong( 0x14 ), /* 20 */
/* 432 */      NdrFcShort( 0x17a ), /* Offset= 378 (822) */
/* 434 */      NdrFcShort( 0xfffffff ), /* Offset= -1 (445) */
/* 436 */      0x21,      /* FC_BOGUS_ARRAY */
/* 438 */      0x3,      /* 3 */
/* 440 */      0x19,      /* Corr desc: field pointer, FC_ULONG */
/* 442 */      0x0,      /* 0 */
/* 444 */      NdrFcShort( 0x0 ), /* 0 */
/* 446 */      NdrFcShort( 0x1 ), /* Corr flags: early */
/* 448 */      NdrFcLong( 0xffffffff ), /* -1 */
/* 450 */      NdrFcShort( 0x0 ), /* Corr flags: */
/* 452 */      0x12, 0x0, /* FC_UP */

```

```

/* 466 */      NdrFcShort( 0xffffffff74 ), /* Offset= -140 (326) */
/* 468 */      0x5c,      /* FC_PAD */
/* 470 */      0x5b,      /* FC_END */
/* 472 */      0x1a,      /* FC_BOGUS_STRUCT */
/* 474 */      0x3,      /* 3 */
/* 476 */      NdrFcShort( 0x10 ), /* 16 */
/* 478 */      NdrFcShort( 0x0 ), /* 0 */
/* 480 */      NdrFcShort( 0x6 ), /* Offset= 6 (482) */
/* 482 */      0x8,      /* FC_LONG */
/* 484 */      0x40,      /* FC_STRUCTPAD4 */
/* 486 */      0x36,      /* FC_POINTER */
/* 488 */      0x5b,      /* FC_END */
/* 490 */      0x11, 0x0, /* FC_RP */
/* 492 */      NdrFcShort( 0xfffffff ), /* Offset= -36 (448) */
/* 494 */      0x21,      /* FC_BOGUS_ARRAY */
/* 496 */      0x3,      /* 3 */
/* 498 */      NdrFcShort( 0x0 ), /* 0 */
/* 500 */      0x19,      /* Corr desc: field pointer, FC_ULONG */
/* 502 */      0x0,      /* 0 */
/* 504 */      NdrFcShort( 0x0 ), /* 0 */
/* 506 */      NdrFcShort( 0x1 ), /* Corr flags: early */
/* 508 */      NdrFcLong( 0xffffffff ), /* -1 */
/* 510 */      NdrFcShort( 0x0 ), /* Corr flags: */
/* 512 */      0x4c,      /* FC_EMBEDDED_COMPLEX */
/* 514 */      0x0,      /* 0 */
/* 516 */      NdrFcShort( 0xfffffff58 ), /* Offset= -168 (336) */
/* 518 */      0x5c,      /* FC_PAD */
/* 520 */      0x5b,      /* FC_END */
/* 522 */      0x1a,      /* FC_BOGUS_STRUCT */
/* 524 */      0x3,      /* 3 */
/* 526 */      NdrFcShort( 0x10 ), /* 16 */
/* 528 */      NdrFcShort( 0x0 ), /* 0 */
/* 530 */      NdrFcShort( 0x6 ), /* Offset= 6 (520) */
/* 532 */      0x8,      /* FC_LONG */
/* 534 */      0x40,      /* FC_STRUCTPAD4 */
/* 536 */      0x36,      /* FC_POINTER */
/* 538 */      0x5b,      /* FC_END */
/* 540 */      0x11, 0x0, /* FC_RP */
/* 542 */      NdrFcShort( 0xfffffff ), /* Offset= -36 (486) */
/* 544 */      0x21,      /* FC_BOGUS_ARRAY */
/* 546 */      0x3,      /* 3 */
/* 548 */      NdrFcShort( 0x0 ), /* 0 */
/* 550 */      0x19,      /* Corr desc: field pointer, FC_ULONG */
/* 552 */      0x0,      /* 0 */
/* 554 */      NdrFcShort( 0x0 ), /* 0 */
/* 556 */      NdrFcShort( 0x1 ), /* Corr flags: early */
/* 558 */      NdrFcLong( 0xffffffff ), /* -1 */
/* 560 */      NdrFcShort( 0x0 ), /* Corr flags: */
/* 562 */      0x4c,      /* FC_EMBEDDED_COMPLEX */
/* 564 */      0x0,      /* 0 */
/* 566 */      NdrFcShort( 0xfffffff44 ), /* Offset= -188 (354) */
/* 568 */      0x5c,      /* FC_PAD */
/* 570 */      0x5b,      /* FC_END */
/* 572 */      0x1a,      /* FC_BOGUS_STRUCT */
/* 574 */      0x3,      /* 3 */
/* 576 */      NdrFcShort( 0x10 ), /* 16 */

```

```

/* 550 */ NdrFcShort( 0x0 ), /* 0 */
/* 552 */ NdrFcShort( 0x6 ), /* Offset= 6 (558) */
/* 554 */ 0x8, /* FC_LONG */
/* 556 */ 0x40, /* FC_STRUCTPAD4 */
/* 558 */ 0x36, /* FC_POINTER */
/* 560 */ 0x5b, /* FC_END */
/* 562 */ 0x11, 0x0, /* FC_RP */
NdrFcShort( 0xfffffddc ), /* Offset= -36 (524) */
0x21, /* FC_BOOLEAN_ARRAY */
0x3, /* 3 */
/* 564 */ NdrFcShort( 0x0 ), /* 0 */
/* 566 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
0x0, /* */
/* 568 */ NdrFcShort( 0x0 ), /* 0 */
/* 570 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 572 */ NdrFcLong( 0xffffffff ), /* -1 */
/* 576 */ NdrFcShort( 0x0 ), /* Corr flags: */
/* 578 */ 0x12, 0x0, /* FC_UP */
/* 580 */ NdrFcShort( 0x176 ), /* Offset= 374 (954) */
/* 582 */ 0x5c, /* FC_PAD */
0x5b, /* FC_END */
/* 584 */ 0x1a, /* FC_BOOLEAN_STRUCT */
0x3, /* 3 */
/* 586 */ NdrFcShort( 0x10 ), /* 16 */
/* 588 */ NdrFcShort( 0x0 ), /* 0 */
/* 590 */ NdrFcShort( 0x6 ), /* Offset= 6 (596) */
/* 592 */ 0x8, /* FC_LONG */
0x40, /* FC_STRUCTPAD4 */
/* 594 */ 0x36, /* FC_POINTER */
0x5b, /* FC_END */
0x11, 0x0, /* FC_RP */
/* 598 */ NdrFcShort( 0xfffffddc ), /* Offset= -36 (562) */
/* 600 */ 0x2f, /* FC_IP */
0x5a, /* FC_CONSTANT_IID */
/* 602 */ NdrFcLong( 0x2f ), /* 47 */
/* 606 */ NdrFcShort( 0x0 ), /* 0 */
/* 608 */ NdrFcShort( 0x0 ), /* 0 */
/* 610 */ 0xc0, /* 192 */
0x0, /* 0 */
/* 612 */ 0x0, /* 0 */
0x0, /* 0 */
/* 614 */ 0x0, /* 0 */
0x0, /* 0 */
/* 616 */ 0x0, /* 0 */
/* 618 */ 0x46, /* 70 */
0x1b, /* FC_CARRAY */
0x0, /* 0 */
/* 620 */ NdrFcShort( 0x1 ), /* 1 */
/* 622 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
0x0, /* */
/* 624 */ NdrFcShort( 0x4 ), /* 4 */
/* 626 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 628 */ 0x1, /* FC_BYTE */
0x5b, /* FC_END */
/* 630 */ 0x1a, /* FC_BOOLEAN_STRUCT */

```

```

0x3, /* 3 */
/* 632 */ NdrFcShort( 0x18 ), /* 24 */
/* 634 */ NdrFcShort( 0x0 ), /* 0 */
/* 636 */ NdrFcShort( 0xa ), /* Offset= 10 (646) */
/* 638 */ 0x8, /* FC_LONG */
0x8, /* FC_LONG */
/* 640 */ 0x4c, /* FC_EMBEDDED_COMPLEX */
0x0, /* 0 */
/* 642 */ NdrFcShort( 0xffffffd6 ), /* Offset= -42 (600) */
/* 644 */ 0x36, /* FC_POINTER */
0x5b, /* FC_END */
0x12, 0x0, /* FC_UP */
/* 648 */ NdrFcShort( 0xffffffe2 ), /* Offset= -30 (618) */
/* 650 */ 0x21, /* FC_BOOLEAN_ARRAY */
0x3, /* 3 */
/* 652 */ NdrFcShort( 0x0 ), /* 0 */
/* 654 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
0x0, /* */
/* 656 */ NdrFcShort( 0x0 ), /* 0 */
/* 658 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 660 */ NdrFcLong( 0xffffffff ), /* -1 */
/* 662 */ NdrFcShort( 0x0 ), /* Corr flags: */
/* 666 */ 0x12, 0x0, /* FC_UP */
/* 668 */ NdrFcShort( 0xfffffdda ), /* Offset= -38 (630) */
/* 670 */ 0x5c, /* FC_PAD */
0x5b, /* FC_END */
0x1a, /* FC_BOOLEAN_STRUCT */
0x3, /* 3 */
/* 674 */ NdrFcShort( 0x10 ), /* 16 */
/* 676 */ NdrFcShort( 0x0 ), /* 0 */
/* 678 */ NdrFcShort( 0x6 ), /* Offset= 6 (684) */
/* 680 */ 0x8, /* FC_LONG */
0x40, /* FC_STRUCTPAD4 */
/* 682 */ 0x36, /* FC_POINTER */
0x5b, /* FC_END */
0x11, 0x0, /* FC_RP */
/* 684 */ NdrFcShort( 0xfffffddc ), /* Offset= -36 (650) */
0x1d, /* FC_SMFARRAY */
0x0, /* 0 */
/* 690 */ NdrFcShort( 0x8 ), /* 8 */
/* 692 */ 0x1, /* FC_BYTE */
0x5b, /* FC_END */
0x15, /* FC_STRUCT */
0x3, /* 3 */
/* 696 */ NdrFcShort( 0x10 ), /* 16 */
/* 698 */ 0x8, /* FC_LONG */
0x6, /* FC_SHORT */
/* 700 */ 0x6, /* FC_SHORT */
0x4c, /* FC_EMBEDDED_COMPLEX */
0x0, /* 0 */
/* 702 */ NdrFcShort( 0xfffffff1 ), /* Offset= -15 (688) */
0x5b, /* FC_END */
0x1a, /* FC_BOOLEAN_STRUCT */
0x3, /* 3 */

```



```

/* 708 */ NdrFcShort( 0x20 ), /* 32 */
/* 710 */ NdrFcShort( 0x0 ), /* 0 */
/* 712 */ NdrFcShort( 0xa ), /* Offset= 10 (722) */
/* 714 */ 0x8, /* FC_LONG */
/* 716 */ 0x40, /* FC_STRUCTPAD4 */
/* 718 */ 0x36, /* FC_POINTER */
/* 720 */ 0x4c, /* FC_EMBEDDED_COMPLEX */
/* 722 */ 0x0, /* 0 */
/* 724 */ NdrFcShort( 0xffffffe7 ), /* Offset= -25 (694) */
/* 726 */ 0x5b, /* FC_END */
/* 728 */ 0x11, 0x0, /* FC_RP */
/* 730 */ NdrFcShort( 0xfffffff12 ), /* Offset= -238 (486) */
/* 732 */ 0x1b, /* FC_CARRAY */
/* 734 */ 0x0, /* 0 */
/* 736 */ NdrFcShort( 0x1 ), /* 1 */
/* 738 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
/* 740 */ 0x0, /* 0 */
/* 742 */ NdrFcShort( 0x0 ), /* 0 */
/* 744 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 746 */ 0x1, /* FC_BYTE */
/* 748 */ 0x5b, /* FC_END */
/* 750 */ 0x1a, /* FC_BOGUS_STRUCT */
/* 752 */ 0x3, /* 3 */
/* 754 */ NdrFcShort( 0x10 ), /* 16 */
/* 756 */ NdrFcShort( 0x0 ), /* 0 */
/* 758 */ NdrFcShort( 0x6 ), /* Offset= 6 (750) */
/* 760 */ 0x8, /* FC_LONG */
/* 762 */ 0x40, /* FC_STRUCTPAD4 */
/* 764 */ 0x36, /* FC_POINTER */
/* 766 */ 0x5b, /* FC_END */
/* 768 */ 0x12, 0x0, /* FC_UP */
/* 770 */ NdrFcShort( 0xffffffe6 ), /* Offset= -26 (726) */
/* 772 */ 0x1b, /* FC_CARRAY */
/* 774 */ 0x1, /* 1 */
/* 776 */ NdrFcShort( 0x2 ), /* 2 */
/* 778 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
/* 780 */ 0x0, /* 0 */
/* 782 */ NdrFcShort( 0x0 ), /* 0 */
/* 784 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 786 */ 0x6, /* FC_SHORT */
/* 788 */ 0x5b, /* FC_END */
/* 790 */ 0x1a, /* FC_BOGUS_STRUCT */
/* 792 */ 0x3, /* 3 */
/* 794 */ NdrFcShort( 0x10 ), /* 16 */
/* 796 */ NdrFcShort( 0x0 ), /* 0 */
/* 798 */ NdrFcShort( 0x6 ), /* Offset= 6 (778) */
/* 800 */ 0x8, /* FC_LONG */
/* 802 */ 0x40, /* FC_STRUCTPAD4 */
/* 804 */ 0x36, /* FC_POINTER */
/* 806 */ 0x5b, /* FC_END */
/* 808 */ 0x12, 0x0, /* FC_UP */
/* 810 */ NdrFcShort( 0xffffffe6 ), /* Offset= -26 (754) */
/* 812 */ 0x1b, /* FC_CARRAY */
/* 814 */ 0x1, /* 1 */
/* 816 */ NdrFcShort( 0x2 ), /* 2 */
/* 818 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
/* 820 */ 0x0, /* 0 */
/* 822 */ NdrFcShort( 0x0 ), /* 0 */
/* 824 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 826 */ 0x6, /* FC_SHORT */
/* 828 */ 0x5b, /* FC_END */
/* 830 */ 0x1a, /* FC_BOGUS_STRUCT */
/* 832 */ 0x3, /* 3 */
/* 834 */ NdrFcShort( 0x10 ), /* 16 */
/* 836 */ NdrFcShort( 0x0 ), /* 0 */
/* 838 */ NdrFcShort( 0x6 ), /* Offset= 6 (834) */
/* 840 */ 0x8, /* FC_LONG */
/* 842 */ 0x40, /* FC_STRUCTPAD4 */
/* 844 */ 0x36, /* FC_POINTER */
/* 846 */ 0x5b, /* FC_END */
/* 848 */ 0x12, 0x0, /* FC_UP */
/* 850 */ NdrFcShort( 0xffffffe6 ), /* Offset= -26 (810) */
/* 852 */ 0x15, /* FC_STRUCT */
/* 854 */ 0x3, /* 3 */
/* 856 */ NdrFcShort( 0x8 ), /* 8 */
/* 858 */ 0x8, /* FC_LONG */
/* 860 */ 0x5c, /* FC_PAD */
/* 862 */ 0x5b, /* FC_END */
/* 864 */ 0x1b, /* FC_CARRAY */
/* 866 */ 0x3, /* 3 */
/* 868 */ NdrFcShort( 0x8 ), /* 8 */
/* 870 */ 0x7, /* Corr desc: FC_USHORT */
/* 872 */ 0x0, /* 0 */
/* 874 */ NdrFcShort( 0xffc8 ), /* -56 */
/* 876 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 878 */ 0x4c, /* FC_EMBEDDED_COMPLEX */
/* 880 */ 0x0, /* 0 */
/* 882 */ NdrFcShort( 0xffffffec ), /* Offset= -20 (838) */
/* 884 */ 0x5c, /* FC_PAD */

```

```

/* 862 */      0x5b,          /* FC_END */
/* 864 */      0x1a,          /* FC_BOGUS_STRUCT */
/* 866 */      0x3,           /* 3 */
/* 868 */      NdrFcShort( 0x38 ), /* 56 */
/* 870 */      NdrFcShort( 0xfffffec ), /* Offset= -20 (846) */
/* 872 */      NdrFcShort( 0x0 ), /* Offset= 0 (868) */
/* 874 */      0x6,          /* FC_SHORT */
/* 876 */      0x6,          /* FC_SHORT */
/* 878 */      0x8,          /* FC_LONG */
/* 880 */      0x8,          /* FC_LONG */
/* 882 */      0x40,         /* FC_STRUCTPAD4 */
/* 884 */      0x4c,         /* FC_EMBEDDED_COMPLEX */
/* 886 */      0x0,          /* 0 */
/* 888 */      NdrFcShort( 0xfffffe0f ), /* Offset= -497 (380) */
/* 890 */      0x5b,          /* FC_END */
/* 892 */      0x12, 0x0,     /* FC_UP */
/* 894 */      NdrFcShort( 0xffffff04 ), /* Offset= -252 (630) */
/* 896 */      0x1,          /* FC_UP [simple_pointer] */
/* 898 */      0x5c,         /* FC_BYTE */
/* 900 */      0x5c,         /* FC_PAD */
/* 902 */      0x12, 0x8,     /* FC_UP [simple_pointer] */
/* 904 */      0x6,          /* FC_SHORT */
/* 906 */      0x5c,         /* FC_PAD */
/* 908 */      0x12, 0x8,     /* FC_UP [simple_pointer] */
/* 910 */      0x8,          /* FC_LONG */
/* 912 */      0x5c,         /* FC_PAD */
/* 914 */      0x12, 0x8,     /* FC_UP [simple_pointer] */
/* 916 */      0xb,          /* FC_HYPER */
/* 918 */      0x5c,         /* FC_PAD */
/* 920 */      0x12, 0x8,     /* FC_UP [simple_pointer] */
/* 922 */      0xa,          /* FC_FLOAT */
/* 924 */      0x5c,         /* FC_PAD */
/* 926 */      0x12, 0x8,     /* FC_UP [simple_pointer] */
/* 928 */      0xc,          /* FC_DOUBLE */
/* 930 */      0x5c,         /* FC_PAD */
/* 932 */      0x12, 0x0,     /* FC_UP */
/* 934 */      NdrFcShort( 0xfffffda2 ), /* Offset= -606 (304) */
/* 936 */      0x12, 0x10,    /* FC_UP [pointer_deref] */
/* 938 */      NdrFcShort( 0xfffffda4 ), /* Offset= -604 (310) */
/* 940 */      0x12, 0x10,    /* FC_UP [pointer_deref] */
/* 942 */      NdrFcShort( 0xfffffdb8 ), /* Offset= -582 (336) */
/* 944 */      0x12, 0x10,    /* FC_UP [pointer_deref] */
/* 946 */      NdrFcShort( 0xfffffdbc ), /* Offset= -568 (354) */
/* 948 */      0x12, 0x10,    /* FC_UP [pointer_deref] */
/* 950 */      NdrFcShort( 0xfffffdd6 ), /* Offset= -554 (372) */
/* 952 */      0x12, 0x10,    /* FC_UP [pointer_deref] */
/* 954 */      NdrFcShort( 0x2 ), /* Offset= 2 (932) */

```

```

/* 934 */      0x12, 0x0,     /* FC_UP */
/* 936 */      NdrFcShort( 0x14 ), /* Offset= 20 (954) */
/* 938 */      0x15,          /* FC_STRUCT */
/* 940 */      0x7,           /* 7 */
/* 942 */      NdrFcShort( 0x10 ), /* 16 */
/* 944 */      0x6,          /* FC_SHORT */
/* 946 */      0x1,          /* FC_BYTE */
/* 948 */      0x1,          /* FC_BYTE */
/* 950 */      0x8,          /* FC_LONG */
/* 952 */      0xb,          /* FC_HYPER */
/* 954 */      0x5b,          /* FC_END */
/* 956 */      0x12, 0x0,     /* FC_UP */
/* 958 */      NdrFcShort( 0xfffffff4 ), /* Offset= -12 (936) */
/* 960 */      0x2,          /* FC_UP [simple_pointer] */
/* 962 */      0x12, 0x8,     /* FC_CHAR */
/* 964 */      0x5c,         /* FC_PAD */
/* 966 */      0x1a,          /* FC_BOGUS_STRUCT */
/* 968 */      0x7,           /* 7 */
/* 970 */      NdrFcShort( 0x20 ), /* 32 */
/* 972 */      NdrFcShort( 0x0 ), /* 0 */
/* 974 */      NdrFcShort( 0x0 ), /* Offset= 0 (960) */
/* 976 */      0x8,          /* FC_LONG */
/* 978 */      0x8,          /* FC_LONG */
/* 980 */      0x6,          /* FC_SHORT */
/* 982 */      0x6,          /* FC_SHORT */
/* 984 */      0x6,          /* FC_SHORT */
/* 986 */      0x6,          /* FC_SHORT */
/* 988 */      0x4c,         /* FC_EMBEDDED_COMPLEX */
/* 990 */      0x0,          /* 0 */
/* 992 */      NdrFcShort( 0xffffffc3c ), /* Offset= -964 (6) */
/* 994 */      0x5c,         /* FC_PAD */
/* 996 */      0x5b,          /* FC_END */
/* 998 */      0xb4,         /* FC_USER_MARSHAL */
/* 1000 */      0x83,        /* 131 */
/* 1002 */      NdrFcShort( 0x0 ), /* 0 */
/* 1004 */      NdrFcShort( 0x18 ), /* 24 */
/* 1006 */      NdrFcShort( 0x0 ), /* 0 */
/* 1008 */      NdrFcShort( 0xffffffc2c ), /* Offset= -980 (2) */
/* 1010 */      0x11, 0x4,     /* FC_RP [allocated_on_stack] */
/* 1012 */      NdrFcShort( 0x6 ), /* Offset= 6 (992) */
/* 1014 */      0x13, 0x0,     /* FC_OP */
/* 1016 */      NdrFcShort( 0xffffffdc ), /* Offset= -36 (954) */
/* 1018 */      0xb4,         /* FC_USER_MARSHAL */
/* 1020 */      0x83,        /* 131 */
/* 1022 */      NdrFcShort( 0x0 ), /* 0 */
/* 1024 */      NdrFcShort( 0x18 ), /* 24 */
/* 1026 */      NdrFcShort( 0x0 ), /* 0 */
/* 1028 */      NdrFcShort( 0xfffffff4 ), /* Offset= -12 (988) */
/* 1030 */      0x0
}
};

static const USER_MARSHAL_ROUTINE_QUADRUPLE
UserMarshalRoutines[ WIRE_MARSHAL_TABLE_SIZE ] =
{

```

```

        {
            VARIANT_UserSize
            ,VARIANT_UserMarshal
            ,VARIANT_UserUnmarshal
            ,VARIANT_UserFree
        }
    };

/* Standard interface: __MIDL_itf_tpcc_com_ps_0000, ver. 0.0,
GUID={0x00000000,0x0000,0x0000,{0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00}}
*/

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

/* Object interface: ITPCC, ver. 0.0,
GUID={0xFEEE6AA2,0x84B1,0x11d2,{0xBA,0x47,0x00,0xC0,0x4F,0xBF,0xE0,0x8B}}
*/

#pragma code_seg(".orpc")
static const unsigned short ITPCC_FormatStringOffsetTable[] =
{
    0,
    44,
    88,
    132,
    176,
    220
};

static const MIDL_STUBLESS_PROXY_INFO ITPCC_ProxyInfo =
{
    &Object_StubDesc,
    _MIDL_ProcFormatString.Format,
    &ITPCC_FormatStringOffsetTable[-3],
    0,
    0,
    0
};

static const MIDL_SERVER_INFO ITPCC_ServerInfo =
{
    &Object_StubDesc,
    0,
    _MIDL_ProcFormatString.Format,
    &ITPCC_FormatStringOffsetTable[-3],
    0,
    0,
    0,
    0
};
CINTERFACE_PROXY_VTABLE(9) _ITPCCProxyVtbl =
{
    &ITPCC_ProxyInfo,

```

```

    &IID_ITPCC,
    IUnknown_QueryInterface_Proxy,
    IUnknown_AddRef_Proxy,
    IUnknown_Release_Proxy,
    (void *) (INT_PTR) -1 /* ITPCC::NewOrder */ ,
    (void *) (INT_PTR) -1 /* ITPCC::Payment */ ,
    (void *) (INT_PTR) -1 /* ITPCC::Delivery */ ,
    (void *) (INT_PTR) -1 /* ITPCC::StockLevel */ ,
    (void *) (INT_PTR) -1 /* ITPCC::OrderStatus */ ,
    (void *) (INT_PTR) -1 /* ITPCC::CallSetComplete */
};

const CInterfaceStubVtbl _ITPCCStubVtbl =
{
    &IID_ITPCC,
    &ITPCC_ServerInfo,
    9,
    0, /* pure interpreted */
    CStdStubBuffer_METHODS
};

static const MIDL_STUB_DESC Object_StubDesc =
{
    0,
    NdrOleAllocate,
    NdrOleFree,
    0,
    0,
    0,
    0,
    0,
    _MIDL_TypeFormatString.Format,
    1, /* -error bounds_check flag */
    0x50002, /* Ndr library version */
    0,
    0x600015b, /* MIDL Version 6.0.347 */
    0,
    UserMarshalRoutines,
    0, /* notify & notify_flag routine table */
    0x1, /* MIDL flag */
    0, /* cs routines */
    0, /* proxy/server info */
    0 /* Reserved5 */
};

const CInterfaceProxyVtbl * _tpcc_com_ps_ProxyVtblList[] =
{
    {
        (CInterfaceProxyVtbl *) &_ITPCCProxyVtbl,
        0
    }
};

const CInterfaceStubVtbl * _tpcc_com_ps_StubVtblList[] =
{
    {
        (CInterfaceStubVtbl *) &_ITPCCStubVtbl,
        0
    }
};

PCInterfaceName const _tpcc_com_ps_InterfaceNamesList[] =
{
    "ITPCC",
    0
};

```

```

#define _tpcc_com_ps_CHECK_IID(n)    IID_GENERIC_CHECK_IID( _tpcc_com_ps,
pIID, n)

int __stdcall _tpcc_com_ps_IID_Lookup( const IID * pIID, int * pIndex )
{
    if(!_tpcc_com_ps_CHECK_IID(0))
    {
        *pIndex = 0;
        return 1;
    }

    return 0;
}

const ExtendedProxyFileInfo tpcc_com_ps_ProxyFileInfo =
{
    (PCInterfaceProxyVtblList *) & _tpcc_com_ps_ProxyVtblList,
    (PCInterfaceStubVtblList *) & _tpcc_com_ps_StubVtblList,
    (const PCInterfaceName * ) & _tpcc_com_ps_InterfaceNamesList,
    0, // no delegation
    & _tpcc_com_ps_IID_Lookup,
    1,
    2,
    0, /* table of [async_uuid] interfaces */
    0, /* Filler1 */
    0, /* Filler2 */
    0 /* Filler3 */
};

#endif /* defined(_M_IA64) || defined(_M_AMD64) */

```

A2. Back-End Source Code, Scripts

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 -DWIN64 -J -Zp8 -DREG_KIT_METHOD -G2
CFLAGS_OUT=/Fo
CFLAGS_DEBUG=

# Linker Configuration (MSVC)
LD_EXEC=link.exe
LD_STORP=link.exe
LDFLAGS_EXEC=
LDFLAGS_SHLIB=/DLL
LDFLAGS_STORP=$(LDFLAGS_SHLIB) /DEF:rpctpcc.def
LDFLAGS_LIB=/LIBPATH:$(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=CK040607

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

@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=%HOME%\tpc-c.ibm
set TPCC_SQLLIB=%HOME%\SQLLIB
set TPCC_RUNDATA=%HOME%\tpc-c.ibm\RUNS

@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

```

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 /* May be ignored by RTE */
#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_o_l_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
```

lval.h

```
#ifndef __LVAL_H
#define __LVAL_H
#define WAREHOUSES 19800
#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
```

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
```

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 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

```

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 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$(OBJEXT):    tpccdbg.c
tpccctx$(OBJEXT):    tpccctx.c
tpccmisc$(OBJEXT):   tpccmisc.c

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

```

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 - 2005
** All Rights Reserved.
**
** US Government Users Restricted Rights - Use, duplication or
** disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
*****/

/*
 * tccdbg.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_tmstp(&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);

```

```

    if ((err_fp = fopen(err_fn, "a+")) == NULL)
    {
        return;
    }

    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\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, "\t\tts_O_ID[%d]      = %d\n",
                j, delivery_ptr->s_O_ID[j]);
    }
    fprintf(debug_fp, "\t}\n\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, "\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_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 (%llX)\n",
        in_payment->s_H_DATE_time, in_payment->s_H_DATE_time);
fprintf(debug_fp, "\n}\n\n");

fprintf(debug_fp, "out_payment_struct {\n");
fprintf(debug_fp, "\ts_H_DATE      = %lld (%llX)\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 (%llX)\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\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=====");
}

```

```

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);
}

```

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);
}

```

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 - 2005
-- 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
                   ) 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 =

                                WHERE O_W_ID = DEL.W_ID
                                AND O_D_ID = DEL.D_ID
                                AND O_ID   = VAR.O_ID
                                ) AS U
                   ) AS U
  )

```

```

;
SET VAR.AMOUNT = ( SELECT SUM( OL_AMOUNT )

                  FROM OLD TABLE ( UPDATE ORDER_LINE

                                SET OL_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
                  ) 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 )
                                      ||
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
                                    || 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
                                                THEN S_QUANTITY
                                                ELSE S_QUANTITY
                        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

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 ) THEN S_DIST_10
END

WHERE S_I_ID = NEW_OL_ALL.I_ID
AND S_W_ID = NEW_OL_ALL.SUPP_W_ID
) AS U
) 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 THEN S_QUANTITY
- NEW_OL_LOCAL.I_QTY ELSE S_QUANTITY
- 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 ) THEN S_DIST_08
WHEN SMALLINT( 9 ) THEN S_DIST_09
WHEN SMALLINT( 10 ) THEN S_DIST_10

        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
$

```

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\rrpctpc!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\rrpctpcc!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\rrpctpcc!dels'
not fenced;

```

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

```

```

LDFLAGS =        $(LDFLAGS_STORP) $(LDFLAGS_LIB)

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

STORED_PROCS = new ord del

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

DLL =            rpctpcc$(SHLIBEXT)

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

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

clean:  connect uncatalog unexplain disconnect
        - $(ERASE) $(TPCC_SPDIR)$(SLASH)rpctpcc$(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 -1 -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 .sql

# d230437mte: 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 $.sql"
    -db2 prep $.sql $(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) $(LDFLAGS) $(UTIL_OBJ) tpcc_all_sql$(OBJEXT)
$(LDFLAGS_OUT)$@

#
#####
# 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

```

rpctpcc.def

```

LIBRARY rpctpcc
DESCRIPTION "Library of TPC-C Transactions (Stored Procedures)"
EXPORTS
news
ords
dels

```

tpcc_all_sql.sql

```

/*****
****
** 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.
*****/

/*
 * tpcc_all_sql.sql - 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 "lval.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     itemCount ;

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 stockDistrictInformation
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 inputItemCount 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 , 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_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 , :supply_w_id1 )

        ) AS X
        , TABLE( NEW_OL_ALL(
        ,      I_ID
        ,      I_QTY
        ,      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 )

( OL_NUMBER , I_ID , I_QTY      , I_SUPPLY_W_ID ) AS X
  ) AS ITEMLIST
        , TABLE( NEW_OL_ALL( I_ID
                               , I_QTY
                               , 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 )

( OL_NUMBER , I_ID , I_QTY      , I_SUPPLY_W_ID ) AS X
  ) AS ITEMLIST
        , TABLE( NEW_OL_ALL( I_ID
                               , I_QTY
                               , 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

```

```

        ) 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
            , 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 )

                ) AS X
            ( OL_NUMBER , I_ID , I_QTY      , I_SUPPLY_W_ID )
            ) AS ITEMLIST

            , TABLE( NEW_OL_ALL(
                I_ID
                , I_QTY
                , 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
                        , 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 )

```

```

) AS X
( OL_NUMBER , I_ID , I_QTY      , I_SUPPLY_W_ID ) AS ITEMLIST
) AS ITEMLIST

, TABLE( NEW_OL_ALL( I_ID
                      , I_QTY
                      , 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 , :supply_w_id5 )
( SMALLINT( 7 )      , :id6 , :ol_quantity6 , :supply_w_id6 )

                  ) AS X
                  , TABLE( NEW_OL_ALL( I_ID
                                          , I_QTY
                                          , 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 ( 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 )

( OL_NUMBER , I_ID , I_QTY      , I_SUPPLY_W_ID ) AS X
    ) AS ITEMLIST
        , TABLE( NEW_OL_ALL( I_ID
                                , I_QTY
                                , 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 )

( OL_NUMBER , I_ID , I_QTY      , I_SUPPLY_W_ID ) AS X
  ) AS ITEMLIST

      , TABLE( NEW_OL_ALL(      I_ID
                                , I_QTY
                                , 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
                                ,
                                ,
                                ,
                                )
              ) AS NEW_OL_ALL

```

```

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 )
( 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 )

) AS X
( OL_NUMBER , I_ID , I_QTY , I_SUPPLY_W_ID )
) AS ITEMLIST

, TABLE( NEW_OL_ALL(
I_ID
, I_QTY
, 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
  ,
  , 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 )

( OL_NUMBER , I_ID , I_QTY      , I_SUPPLY_W_ID
  ) AS X
  ) AS ITEMLIST

, TABLE( NEW_OL_ALL(      I_ID
                          , I_QTY
                          , 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 )
( 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 )

        ) AS X
( OL_NUMBER , I_ID , I_QTY      , I_SUPPLY_W_ID
        ) AS ITEMLIST

        , TABLE( NEW_OL_ALL( I_ID
        , I_QTY
        , 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
, 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 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_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
                          , 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
, 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 )

) 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

FROM DATA

```

```

        ) 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 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_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
      , 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_newword = (struct in_newword_struct *) pin ;
newword     = (struct out_newword_struct *) pout ;

#ifdef DEBUGIT
    new_debug( newword, in_newword, "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_newword-
>s_OL_CNT ; inputItemArrayIndex++ )
{
    i_priceArray[ inputItemArrayIndex ] = 0 ;
}

newword->deadlocks = -1 ;

retry_tran:

newword->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_0_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_da
ta , :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_da
ta , :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_da
ta , :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_da
ta , :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_da
ta , :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_da
ta , :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_da
ta , :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_da
ta , :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_da
ta , :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_da
ta , :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_da
ta , :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_da
ta , :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_da
ta , :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_da
ta , :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_da
ta , :s_quantity ;
            NEW_CURSOR_ERROR
        }
        break ;
    default:
        sqlerror(NEWORD_SQL, "Default switch on local
orderline/stock/index", __FILE__, __LINE__, &sqlca );
        goto ferror;
    }
}
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_da
ta , :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_da
ta , :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_da
ta , :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_da
ta , :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_da
ta , :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_da
ta , :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_da
ta , :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_da
ta , :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_da
ta , :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_da
ta , :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_da
ta , :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_da
ta , :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_da
ta , :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_da
ta , :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_da
ta , :s_quantity ;
            NEW_CURSOR_ERROR
        }
        break ;
    default:
        sqlerror(NEWORD_SQL, "Default switch on remote
orderline/stock/index", __FILE__, __LINE__, &sqlca);
        goto ferror;
    }
}

for ( inputItemArrayIndex = 0 ;
inputItemArrayIndex < in_neword->s_0_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_0_OL_CNT == in_neword->s_0_OL_CNT )
    {
        neword->s_transtatus = TRAN_OK ;

        EXEC SQL COMMIT;

        if( sqlca.sqlcode != 0 )
        {
            sqlerror(NEWORD_SQL, "COMMIT", __FILE__, __LINE__,
&sqlca ) ;
            goto ferror;
        }
    }
    else
    {
        neword->s_transtatus = INVALID_ITEM ;

        EXEC SQL ROLLBACK WORK ;
    }
}

```

```

        if ( sqlca.sqlcode != 0 )
        {
            neword->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( neword, in_neword, "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:

    neword->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] = {8,8,8,8,8,8,8,8,8,8, /*0-9*/
8,8,8,8,8,8,8,8,8,8, /*10-19*/
8,8,8,8,8,8,8,8,8,8, /*20-29*/
8,8,8,8,8,8,8,8,8,8, /*30-39*/
8,8,8,8,8,8,8,8,8,8, /*40-49*/
8,8,8,8,8,8,8,8,8,8, /*50-59*/
8,8,8,8,8,1,8,8,8,8, /*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*/
8,8,8,8,8,8,8,8,8,8, /*100-109*/
8,8,8,8,8,8,8,8,8,8, /*110-119*/
8,8,8,8,8,8,8,8,8,8, /*120-129*/
8,8,8,8,8,8,8,8,8,8, /*130-139*/
8,8,8,8,8,8,8,8,8,8, /*140-149*/
8,8,8,8,8,8,8,8,8,8, /*150-159*/
8,8,8,8,8,8,8,8,8,8, /*160-169*/
8,8,8,8,8,8,8,8,8,8, /*170-179*/
8,8,8,8,8,8,8,8,8,8, /*180-189*/
8,8,8,8,8,8,8,8,8,8, /*190-199*/
8,8,8,8,8,8,8,8,8,8, /*200-209*/
8,8,8,8,8,8,8,8,8,8, /*210-219*/
8,8,8,8,8,8,8,8,8,8, /*220-229*/
8,8,8,8,8,8,8,8,8,8, /*230-239*/
8,8,8,8,8,8,8,8,8,8, /*240-249*/
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_delive
ry_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 ;
    }

#ifdef DEBUGIT
    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 ;

#ifdef DEBUGIT
    del_debug( delivery, in_delivery, "SP upon entry");
#endif

    // Deadlock Handling
    // -----
    // Since we COMMIT inside the for() loop, we must take special
    // care while handling deadlocks. This is best explained by
    // an example.
    //
    // Assume we deadlock on d_id=6. This means that an order from the
    // first 5 districts have already been delivered. We will then
    // restart the loop (retry_tran). However, the loop will restart
    // at d_id = 1! This means that the second (and all subsequent)
    // time through the loop, we will deliver orders for districts that
    // have already been delivered, with the net result being more than
    // 10 orders being delivered.
    //
    // The solution to this problem is to initialize the starting point
    // of the loop *before* the retry_tran label. This will ensure that
    // if we deadlock, we will restart the loop with the same district
    // that we deadlocked on, and we won't deliver any extra orders.
    //
    // NOTE: If we ever change this back to one COMMIT per transaction
    // (instead of one COMMIT per iteration), then the initialization
    // of d_id must be moved back into the for loop. (A rollback due
    // to deadlock in this case would rollback all delivered orders so
    // far, so we'd need to re-deliver them all on the next iteration.)

    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 ;
    }

#ifdef DEBUGIT
    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 ) ;
}

```

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 - 2005
-- 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 $

```

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;

```

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 ***** IMPORTANT ***** ;
ECHO USAGE: db2 -tf EXPLAIN.DDL ;
ECHO ***** IMPORTANT ***** ;
ECHO ***** IMPORTANT ***** ;
ECHO ***** IMPORTANT ***** ;

```

```

ECHO
--
--
-- Set autocommit off
--
UPDATE COMMAND OPTIONS USING C OFF;
--
-- EXPLAIN INSTANCE
--
-- (must be defined first due to referential integrity definitions)
--
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,

```

NOT LOGGED,	STATEMENT_TEXT	CLOB(2M)	NOT NULL	INDEX IN USERSPACE1;			
				--			
LOGGED,	SNAPSHOT	BLOB(10M)	NOT	-- EXPLAIN_OBJECT			
				--			
	QUERY_DEGREE	INTEGER	NOT NULL,	CREATE TABLE EXPLAIN_OBJECT (	EXPLAIN_REQUESTER	VARCHAR(128)	NOT NULL,
	PRIMARY KEY (EXPLAIN_REQUESTER,				EXPLAIN_TIME	TIMESTAMP	NOT NULL,
	EXPLAIN_TIME,				SOURCE_NAME	VARCHAR(128)	NOT NULL,
	SOURCE_NAME,				SOURCE_SCHEMA	VARCHAR(128)	NOT NULL,
	SOURCE_SCHEMA,				SOURCE_VERSION	VARCHAR(64)	NOT NULL,
	SOURCE_VERSION,				EXPLAIN_LEVEL	CHAR(1)	NOT NULL,
	EXPLAIN_LEVEL,				STMTNO	INTEGER	NOT NULL,
	STMTNO,				SECTNO	INTEGER	NOT NULL,
	SECTNO),				OBJECT_SCHEMA	VARCHAR(128)	NOT NULL,
	FOREIGN KEY (EXPLAIN_REQUESTER,				OBJECT_NAME	VARCHAR(128)	NOT NULL,
	EXPLAIN_TIME,				OBJECT_TYPE	CHAR(2)	NOT NULL,
	SOURCE_NAME,				CREATE_TIME	TIMESTAMP,	
	SOURCE_SCHEMA,				STATISTICS_TIME	TIMESTAMP,	
	SOURCE_VERSION)				COLUMN_COUNT	SMALLINT	NOT NULL,
	REFERENCES EXPLAIN_INSTANCE				ROW_COUNT	BIGINT	NOT NULL,
	ON DELETE CASCADE)				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,
					EXTENTS_SIZE	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,			

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,				NULL,			
	OPERATOR_ID	INTEGER	NOT				
NULL,				NULL,			
	ARGUMENT_TYPE	CHAR(8)	NOT				
NULL,				NULL,			
	ARGUMENT_VALUE	VARCHAR(1024),					
LOGGED,	LONG_ARGUMENT_VALUE	CLOB(2M)	NOT	NULL,			
	FOREIGN KEY (EXPLAIN_REQUESTER,			NULL,			
	EXPLAIN_TIME,						
	SOURCE_NAME,			NULL,			
	SOURCE_SCHEMA,						
	SOURCE_VERSION,			NULL,			
	EXPLAIN_LEVEL,						
	STMTNO,			NULL,			
	SECTNO)						
	REFERENCES EXPLAIN_STATEMENT			NULL,			
	ON DELETE CASCADE)						
				NULL,			
IN USERSPACE1							

<pre> 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, </pre>	<pre> LOGGED, 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; </pre>	<pre> 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 </pre>
---	---	---

```

--
-- EXPLAIN_DIAGNOSTIC
--
CREATE TABLE EXPLAIN_DIAGNOSTIC ( 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,
                                DIAGNOSTIC_ID        INTEGER      NOT
NULL,
                                CODE                 INTEGER       NOT
NULL,
                                PRIMARY KEY (EXPLAIN_REQUESTER,
                                           EXPLAIN_TIME,
                                           SOURCE_NAME,
                                           SOURCE_SCHEMA,
                                           SOURCE_VERSION,
                                           EXPLAIN_LEVEL,
                                           STMTNO,
                                           SECTNO,
                                           DIAGNOSTIC_ID),
                                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_DIAGNOSTIC_TOKEN
--
CREATE TABLE EXPLAIN_DIAGNOSTIC_DATA ( 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,
                                DIAGNOSTIC_ID        INTEGER
NOT NULL,
                                ORDINAL              INTEGER
NOT NULL,
                                TOKEN                VARCHAR(1000),
                                TOKEN_LONG           BLOB(3M)
NOT LOGGED,
                                FOREIGN KEY (EXPLAIN_REQUESTER,
                                           EXPLAIN_TIME,
                                           SOURCE_NAME,
                                           SOURCE_SCHEMA,
                                           SOURCE_VERSION,
                                           EXPLAIN_LEVEL,
                                           STMTNO,
                                           SECTNO,
                                           DIAGNOSTIC_ID)
                                REFERENCES EXPLAIN_DIAGNOSTIC
                                ON DELETE CASCADE)

IN USERSPACE1
INDEX IN USERSPACE1;

--
-- ADVISE TABLES
--
--
-- ADVISE_INSTANCE
--
-- (must be defined first due to referential integrity definitions)
--
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,
CLUSTERRATIO     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)  NOT NULL 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
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
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
                                ON DELETE CASCADE)
(START_TIME)
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
                                ON DELETE CASCADE)
(START_TIME)
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 EXP_DIAG_DAT_I1 on
    EXPLAIN_DIAGNOSTIC_DATA(EXPLAIN_TIME, EXPLAIN_LEVEL, STMTNO, SECTNO,
DIAGNOSTIC_ID, ORDINAL);

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;

--
-- The following function formats the explain diagnostic table messages
--
CREATE FUNCTION EXPLAIN_GET_MSGS( EXPLAIN_REQUESTER VARCHAR(128),
                                EXPLAIN_TIME        TIMESTAMP,
                                SOURCE_NAME          VARCHAR(128),
                                SOURCE_SCHEMA        VARCHAR(128),
                                SOURCE_VERSION        VARCHAR(64),
                                EXPLAIN_LEVEL        CHAR(1),
                                STMTNO               INTEGER,
                                SECTNO               INTEGER,
                                LOCALE                VARCHAR(33) )
```

```

RETURNS TABLE ( EXPLAIN_REQUESTER VARCHAR(128),
                  EXPLAIN_TIME      TIMESTAMP,
                  SOURCE_NAME        VARCHAR(128),
                  SOURCE_SCHEMA      VARCHAR(128),
                  SOURCE_VERSION     VARCHAR(64),
                  EXPLAIN_LEVEL      CHAR(1),
                  STMTNO             INTEGER,
                  SECTNO             INTEGER,
                  DIAGNOSTIC_ID      INTEGER,
                  LOCALE             VARCHAR(33),
                  MESSAGE_TEXT       VARCHAR(4096) )
SPECIFIC EXPLAIN_GET_MSGS
LANGUAGE SQL
DETERMINISTIC
NO EXTERNAL ACTION
READS SQL DATA
RETURN SELECT A.A_EXPLAIN_REQUESTER,
              A.A_EXPLAIN_TIME,
              A.A_SOURCE_NAME,
              A.A_SOURCE_SCHEMA,
              A.A_SOURCE_VERSION,
              A.A_EXPLAIN_LEVEL,
              A.A_STMTNO,
              A.A_SECTNO,
              A.A_DIAGNOSTIC_ID,
              F.LOCALE,
              F.MESSAGE_TEXT
FROM EXPLAIN_DIAGNOSTIC A( A_EXPLAIN_REQUESTER,
                          A_EXPLAIN_TIME,
                          A_SOURCE_NAME,
                          A_SOURCE_SCHEMA,
                          A_SOURCE_VERSION,
                          A_EXPLAIN_LEVEL,
                          A_STMTNO,
                          A_SECTNO,
                          A_DIAGNOSTIC_ID,
                          A_CODE ),
TABLE( SYSPROC.EXPLAIN_GET_MSG(
      CAST( NULL AS VARCHAR(33) ),
      A.A_CODE,
      ( SELECT TOKEN FROM EXPLAIN_DIAGNOSTIC_DATA B
        WHERE A.A_EXPLAIN_REQUESTER =
          B.EXPLAIN_REQUESTER
          AND A.A_EXPLAIN_TIME      = B.EXPLAIN_TIME
          AND A.A_SOURCE_NAME        = B.SOURCE_NAME
          AND A.A_SOURCE_SCHEMA      = B.SOURCE_SCHEMA
          AND A.A_SOURCE_VERSION     = B.SOURCE_VERSION
          AND A.A_EXPLAIN_LEVEL      = B.EXPLAIN_LEVEL
          AND A.A_STMTNO             = B.STMTNO
          AND A.A_SECTNO             = B.SECTNO
          AND A.A_DIAGNOSTIC_ID      = B.DIAGNOSTIC_ID
          AND B.ORDINAL=1 ),
      ( SELECT TOKEN FROM EXPLAIN_DIAGNOSTIC_DATA B
        WHERE A.A_EXPLAIN_REQUESTER =
          B.EXPLAIN_REQUESTER
          AND A.A_EXPLAIN_TIME      = B.EXPLAIN_TIME
          AND A.A_SOURCE_NAME        = B.SOURCE_NAME
          AND A.A_SOURCE_SCHEMA      = B.SOURCE_SCHEMA
          AND A.A_SOURCE_VERSION     = B.SOURCE_VERSION
          AND A.A_EXPLAIN_LEVEL      = B.EXPLAIN_LEVEL
          AND A.A_STMTNO             = B.STMTNO
          AND A.A_SECTNO             = B.SECTNO
          AND A.A_DIAGNOSTIC_ID      = B.DIAGNOSTIC_ID
          AND B.ORDINAL=3 ) ) ) F

```

```

--
-- Commit work
--
COMMIT WORK;

```

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 - 2005
-- 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;

```

Appendix B : Database Design

B1. Database build scripts

ALTTBSP_PF_0.ddl

```

activate db TPCC;
connect to TPCC;
alter tablespace cust_01 prefetchsize 0;
alter tablespace cust_02 prefetchsize 0;
alter tablespace cust_03 prefetchsize 0;
alter tablespace cust_04 prefetchsize 0;
alter tablespace cust_05 prefetchsize 0;
alter tablespace cust_06 prefetchsize 0;
alter tablespace cust_07 prefetchsize 0;
alter tablespace cust_08 prefetchsize 0;
alter tablespace cust_09 prefetchsize 0;
alter tablespace cust_10 prefetchsize 0;
alter tablespace cust_11 prefetchsize 0;
alter tablespace cust_12 prefetchsize 0;
alter tablespace cust_idx_01 prefetchsize 0;
alter tablespace cust_idx_02 prefetchsize 0;
alter tablespace cust_idx_03 prefetchsize 0;
alter tablespace cust_idx_04 prefetchsize 0;
alter tablespace cust_idx_05 prefetchsize 0;
alter tablespace cust_idx_06 prefetchsize 0;
alter tablespace cust_idx_07 prefetchsize 0;
alter tablespace cust_idx_08 prefetchsize 0;
alter tablespace cust_idx_09 prefetchsize 0;
alter tablespace cust_idx_10 prefetchsize 0;
alter tablespace cust_idx_11 prefetchsize 0;
alter tablespace cust_idx_12 prefetchsize 0;
alter tablespace dist_01 prefetchsize 0;
alter tablespace dist_02 prefetchsize 0;
alter tablespace dist_03 prefetchsize 0;
alter tablespace dist_04 prefetchsize 0;
alter tablespace dist_05 prefetchsize 0;
alter tablespace dist_06 prefetchsize 0;
alter tablespace dist_07 prefetchsize 0;
alter tablespace dist_08 prefetchsize 0;
alter tablespace dist_09 prefetchsize 0;
alter tablespace dist_10 prefetchsize 0;

```

```

alter tablespace dist_11 prefetchsize 0;
alter tablespace dist_12 prefetchsize 0;
alter tablespace history_01 prefetchsize 0;
alter tablespace history_02 prefetchsize 0;
alter tablespace history_03 prefetchsize 0;
alter tablespace history_04 prefetchsize 0;
alter tablespace history_05 prefetchsize 0;
alter tablespace history_06 prefetchsize 0;
alter tablespace history_07 prefetchsize 0;
alter tablespace history_08 prefetchsize 0;
alter tablespace history_09 prefetchsize 0;
alter tablespace history_10 prefetchsize 0;
alter tablespace history_11 prefetchsize 0;
alter tablespace history_12 prefetchsize 0;
alter tablespace item prefetchsize 0;
alter tablespace norderA_01 prefetchsize 0;
alter tablespace norderA_02 prefetchsize 0;
alter tablespace norderA_03 prefetchsize 0;
alter tablespace norderA_04 prefetchsize 0;
alter tablespace norderA_05 prefetchsize 0;
alter tablespace norderA_06 prefetchsize 0;
alter tablespace norderA_07 prefetchsize 0;
alter tablespace norderA_08 prefetchsize 0;
alter tablespace norderA_09 prefetchsize 0;
alter tablespace norderA_10 prefetchsize 0;
alter tablespace norderA_11 prefetchsize 0;
alter tablespace norderA_12 prefetchsize 0;
alter tablespace norderB_01 prefetchsize 0;
alter tablespace norderB_02 prefetchsize 0;
alter tablespace norderB_03 prefetchsize 0;
alter tablespace norderB_04 prefetchsize 0;
alter tablespace norderB_05 prefetchsize 0;
alter tablespace norderB_06 prefetchsize 0;
alter tablespace norderB_07 prefetchsize 0;
alter tablespace norderB_08 prefetchsize 0;
alter tablespace norderB_09 prefetchsize 0;
alter tablespace norderB_10 prefetchsize 0;
alter tablespace norderB_11 prefetchsize 0;
alter tablespace norderB_12 prefetchsize 0;
alter tablespace order_01 prefetchsize 0;
alter tablespace order_02 prefetchsize 0;
alter tablespace order_03 prefetchsize 0;
alter tablespace order_04 prefetchsize 0;
alter tablespace order_05 prefetchsize 0;
alter tablespace order_06 prefetchsize 0;
alter tablespace order_07 prefetchsize 0;
alter tablespace order_08 prefetchsize 0;
alter tablespace order_09 prefetchsize 0;
alter tablespace order_10 prefetchsize 0;
alter tablespace order_11 prefetchsize 0;
alter tablespace order_12 prefetchsize 0;
alter tablespace order_idx_01 prefetchsize 0;
alter tablespace order_idx_02 prefetchsize 0;
alter tablespace order_idx_03 prefetchsize 0;
alter tablespace order_idx_04 prefetchsize 0;
alter tablespace order_idx_05 prefetchsize 0;
alter tablespace order_idx_06 prefetchsize 0;
alter tablespace order_idx_07 prefetchsize 0;
alter tablespace order_idx_08 prefetchsize 0;

```

ALTTBSP_Pf_4096.ddl

TPC Benchmark™ C Full Disclosure Report
Copyright © 2005 NEC Corporation

215

```

alter tablespace norderB_12 prefetchsize 4096;
alter tablespace order_01 prefetchsize 4096;
alter tablespace order_02 prefetchsize 4096;
alter tablespace order_03 prefetchsize 4096;
alter tablespace order_04 prefetchsize 4096;
alter tablespace order_05 prefetchsize 4096;
alter tablespace order_06 prefetchsize 4096;
alter tablespace order_07 prefetchsize 4096;
alter tablespace order_08 prefetchsize 4096;
alter tablespace order_09 prefetchsize 4096;
alter tablespace order_10 prefetchsize 4096;
alter tablespace order_11 prefetchsize 4096;
alter tablespace order_12 prefetchsize 4096;
alter tablespace order_idx_01 prefetchsize 4096;
alter tablespace order_idx_02 prefetchsize 4096;
alter tablespace order_idx_03 prefetchsize 4096;
alter tablespace order_idx_04 prefetchsize 4096;
alter tablespace order_idx_05 prefetchsize 4096;
alter tablespace order_idx_06 prefetchsize 4096;
alter tablespace order_idx_07 prefetchsize 4096;
alter tablespace order_idx_08 prefetchsize 4096;
alter tablespace order_idx_09 prefetchsize 4096;
alter tablespace order_idx_10 prefetchsize 4096;
alter tablespace order_idx_11 prefetchsize 4096;
alter tablespace order_idx_12 prefetchsize 4096;
alter tablespace orderl_01 prefetchsize 4096;
alter tablespace orderl_02 prefetchsize 4096;
alter tablespace orderl_03 prefetchsize 4096;
alter tablespace orderl_04 prefetchsize 4096;
alter tablespace orderl_05 prefetchsize 4096;
alter tablespace orderl_06 prefetchsize 4096;
alter tablespace orderl_07 prefetchsize 4096;
alter tablespace orderl_08 prefetchsize 4096;
alter tablespace orderl_09 prefetchsize 4096;
alter tablespace orderl_10 prefetchsize 4096;
alter tablespace orderl_11 prefetchsize 4096;
alter tablespace orderl_12 prefetchsize 4096;
alter tablespace stock_01 prefetchsize 4096;
alter tablespace stock_02 prefetchsize 4096;
alter tablespace stock_03 prefetchsize 4096;
alter tablespace stock_04 prefetchsize 4096;
alter tablespace stock_05 prefetchsize 4096;
alter tablespace stock_06 prefetchsize 4096;
alter tablespace stock_07 prefetchsize 4096;
alter tablespace stock_08 prefetchsize 4096;
alter tablespace stock_09 prefetchsize 4096;
alter tablespace stock_10 prefetchsize 4096;
alter tablespace stock_11 prefetchsize 4096;
alter tablespace stock_12 prefetchsize 4096;
alter tablespace ware_01 prefetchsize 4096;
alter tablespace ware_02 prefetchsize 4096;
alter tablespace ware_03 prefetchsize 4096;
alter tablespace ware_04 prefetchsize 4096;
alter tablespace ware_05 prefetchsize 4096;
alter tablespace ware_06 prefetchsize 4096;
alter tablespace ware_07 prefetchsize 4096;
alter tablespace ware_08 prefetchsize 4096;
alter tablespace ware_09 prefetchsize 4096;
alter tablespace ware_10 prefetchsize 4096;

```

```

alter tablespace ware_11 prefetchsize 4096;
alter tablespace ware_12 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 1650);
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
1651 AND 3300);
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
3301 AND 4950);
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
4951 AND 6600);
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
6601 AND 8250);
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
8251 AND 9900);
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
9901 AND 11550);
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

```



```

11551 AND 13200);
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
13201 AND 14850);
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 14851 AND 16500);
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 16501 AND 18150);
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 >=
18151);
SET INTEGRITY FOR CUSTOMER12 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 1650);
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
1651 AND 3300);
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
3301 AND 4950);
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

```

```

4951 AND 6600);
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
6601 AND 8250);
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
8251 AND 9900);
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
9901 AND 11550);
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
11551 AND 13200);
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
13201 AND 14850);
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 14851 AND 16500);
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 16501 AND 18150);
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 >=
18151);
SET INTEGRITY FOR DISTRICT12 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 1650);
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
1651 AND 3300);
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
3301 AND 4950);
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
4951 AND 6600);
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
6601 AND 8250);
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
8251 AND 9900);
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
9901 AND 11550);
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
11551 AND 13200);
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
13201 AND 14850);
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
14851 AND 16500);
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
16501 AND 18150);
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 >=
18151);
SET INTEGRITY FOR HISTORY12 ALL IMMEDIATE UNCHECKED;
connect reset;
```

CRCONST_NEW_ORDER.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 1650) AND (NO_O_ID <= 3607));
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 1651 AND 3300) AND (NO_O_ID <= 3607));
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 3301 AND 4950) AND (NO_O_ID <= 3607));
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 4951 AND 6600) AND (NO_O_ID <= 3607));
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 6601 AND 8250) AND (NO_O_ID <= 3607));
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 8251 AND 9900) AND (NO_O_ID <= 3607));
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 9901 AND 11550) AND (NO_O_ID <= 3607));
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 11551 AND 13200) AND (NO_O_ID <= 3607));
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 13201 AND 14850) AND (NO_O_ID <= 3607));
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 14851 AND 16500) AND (NO_O_ID <= 3607));
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 16501 AND 18150) AND (NO_O_ID <= 3607));
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
>= 18151) AND (NO_O_ID <= 3607));
SET INTEGRITY FOR NEW_ORDERA12 ALL IMMEDIATE UNCHECKED;
connect reset;
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 1650) AND (NO_O_ID >= 3608));
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 1651 AND 3300) AND (NO_O_ID >= 3608));
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 3301 AND 4950) AND (NO_O_ID >= 3608));
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 4951 AND 6600) AND (NO_O_ID >= 3608));
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 6601 AND 8250) AND (NO_O_ID >= 3608));
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 8251 AND 9900) AND (NO_O_ID >= 3608));
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 9901 AND 11550) AND (NO_O_ID >= 3608));
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 11551 AND 13200) AND (NO_O_ID >= 3608));
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 13201 AND 14850) AND (NO_O_ID >= 3608));
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 14851 AND 16500) AND (NO_O_ID >= 3608));
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 16501 AND 18150) AND (NO_O_ID >= 3608));
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
>= 18151) AND (NO_O_ID >= 3608));
SET INTEGRITY FOR NEW_ORDERB12 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 1650);
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 1651 AND 3300);
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 3301 AND 4950);
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 4951 AND 6600);
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 6601 AND 8250);
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 8251 AND 9900);
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 9901 AND 11550);
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 11551 AND 13200);
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 13201 AND 14850);
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 14851 AND 16500);
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 16501 AND 18150);
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
>= 18151);
SET INTEGRITY FOR ORDER_LINE12 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 1650);
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 1651
AND 3300);
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 3301
AND 4950);
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 4951
AND 6600);
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 6601
AND 8250);
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 8251
AND 9900);
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 9901
AND 11550);
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
11551 AND 13200);
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
13201 AND 14850);
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
14851 AND 16500);
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
16501 AND 18150);
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 >= 18151);
SET INTEGRITY FOR ORDERS12 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
1650);
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 1651
AND 3300);
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 3301
AND 4950);
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 4951
AND 6600);
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 6601
AND 8250);
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 8251
AND 9900);
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 9901
AND 11550);
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 11551
AND 13200);
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 13201
AND 14850);
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
14851 AND 16500);
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
16501 AND 18150);
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 >= 18151);
SET INTEGRITY FOR STOCK12 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 1650);
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

```

```

1651 AND 3300);
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
3301 AND 4950);
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
4951 AND 6600);
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
6601 AND 8250);
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
8251 AND 9900);
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
9901 AND 11550);
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
11551 AND 13200);
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
13201 AND 14850);
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 14851 AND 16500);
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 16501 AND 18150);
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 >=
18151);
SET INTEGRITY FOR WAREHOUSE12 ALL IMMEDIATE UNCHECKED;
connect reset;

```

bufferpool-tablespace.ddl

```
connect to tpcc;
```

```
alter bufferpool ibmdefaultbp size 256;
```

```

create bufferpool ware_bp1 deferred          size      200  pagesize
4K;
create bufferpool ware_bp2 deferred          size      200  pagesize
4K;
create bufferpool ware_bp3 deferred          size      200  pagesize
4K;
create bufferpool ware_bp4 deferred          size      200  pagesize
4K;

create bufferpool dist_bp1 deferred          size     2100  pagesize
4K;
create bufferpool dist_bp2 deferred          size     2100  pagesize
4K;
create bufferpool dist_bp3 deferred          size     2100  pagesize
4K;
create bufferpool dist_bp4 deferred          size     2100  pagesize
4K;

create bufferpool item_bp deferred           size     5000  pagesize 4K;
create bufferpool stock_bp1 deferred         size    11600000  pagesize
4K;
create bufferpool stock_bp2 deferred         size    11600000  pagesize
4K;
create bufferpool stock_bp3 deferred         size    11600000  pagesize
4K;
create bufferpool stock_bp4 deferred         size    11600000  pagesize
4K;

create bufferpool cust_bp1 deferred          size     650000  pagesize
4K;
create bufferpool cust_bp2 deferred          size     650000  pagesize
4K;
create bufferpool cust_bp3 deferred          size     650000  pagesize
4K;
create bufferpool cust_bp4 deferred          size     650000  pagesize
4K;

```

```

create bufferpool norder_bp1 deferred        size    215000  pagesize
4K;
create bufferpool norder_bp2 deferred        size    215000  pagesize
4K;
create bufferpool norder_bp3 deferred        size    215000  pagesize
4K;
create bufferpool norder_bp4 deferred        size    215000  pagesize
4K;

create bufferpool cust_idx_bp1 deferred      size    325000  pagesize
8K;
create bufferpool cust_idx_bp2 deferred      size    325000  pagesize
8K;
create bufferpool cust_idx_bp3 deferred      size    325000  pagesize
8K;
create bufferpool cust_idx_bp4 deferred      size    325000  pagesize
8K;

create bufferpool order_bp1 deferred         size    162500  pagesize
8K;
create bufferpool order_bp2 deferred         size    162500  pagesize
8K;
create bufferpool order_bp3 deferred         size    162500  pagesize
8K;
create bufferpool order_bp4 deferred         size    162500  pagesize
8K;

create bufferpool order_idx_bp1 deferred     size    500000  pagesize
8K;
create bufferpool order_idx_bp2 deferred     size    500000  pagesize
8K;
create bufferpool order_idx_bp3 deferred     size    500000  pagesize
8K;
create bufferpool order_idx_bp4 deferred     size    500000  pagesize
8K;

create bufferpool orderl_bp1 deferred        size    345000  pagesize
8K;
create bufferpool orderl_bp2 deferred        size    345000  pagesize
8K;
create bufferpool orderl_bp3 deferred        size    345000  pagesize
8K;
create bufferpool orderl_bp4 deferred        size    345000  pagesize
8K;

create bufferpool history_bp1 deferred       size       350  pagesize
16K;
create bufferpool history_bp2 deferred       size       350  pagesize
16K;
create bufferpool history_bp3 deferred       size       350  pagesize
16K;
create bufferpool history_bp4 deferred       size       350  pagesize
16K;

alter tablespace ware_01 bufferpool ware_bp1;
alter tablespace ware_02 bufferpool ware_bp1;
alter tablespace ware_03 bufferpool ware_bp1;
alter tablespace ware_04 bufferpool ware_bp2;
alter tablespace ware_05 bufferpool ware_bp2;

```

```
alter tablespace nordera_11 bufferpool nordor_bp4;  
alter tablespace nordera_12 bufferpool nordor_bp4;
```

```
alter tablespace norderb_01 bufferpool norder_bp1;
alter tablespace norderb_02 bufferpool norder_bp1;
alter tablespace norderb_03 bufferpool norder_bp1;
alter tablespace norderb_04 bufferpool norder_bp2;
alter tablespace norderb_05 bufferpool norder_bp2;
alter tablespace norderb_06 bufferpool norder_bp2;
alter tablespace norderb_07 bufferpool norder_bp3;
alter tablespace norderb_08 bufferpool norder_bp3;
alter tablespace norderb_09 bufferpool norder_bp3;
alter tablespace norderb_10 bufferpool norder_bp4;
alter tablespace norderb_11 bufferpool norder_bp4;
alter tablespace norderb_12 bufferpool norder_bp4;
```

```
alter tablespace cust_idx_01 bufferpool cust_idx_bp1;
alter tablespace cust_idx_02 bufferpool cust_idx_bp1;
alter tablespace cust_idx_03 bufferpool cust_idx_bp1;
alter tablespace cust_idx_04 bufferpool cust_idx_bp2;
alter tablespace cust_idx_05 bufferpool cust_idx_bp2;
alter tablespace cust_idx_06 bufferpool cust_idx_bp2;
alter tablespace cust_idx_07 bufferpool cust_idx_bp3;
alter tablespace cust_idx_08 bufferpool cust_idx_bp3;
alter tablespace cust_idx_09 bufferpool cust_idx_bp3;
alter tablespace cust_idx_10 bufferpool cust_idx_bp4;
alter tablespace cust_idx_11 bufferpool cust_idx_bp4;
alter tablespace cust_idx_12 bufferpool cust_idx_bp4;
```

```
alter tablespace order_01 bufferpool order_bp1;
alter tablespace order_02 bufferpool order_bp1;
alter tablespace order_03 bufferpool order_bp1;
alter tablespace order_04 bufferpool order_bp2;
alter tablespace order_05 bufferpool order_bp2;
alter tablespace order_06 bufferpool order_bp2;
alter tablespace order_07 bufferpool order_bp3;
alter tablespace order_08 bufferpool order_bp3;
alter tablespace order_09 bufferpool order_bp3;
alter tablespace order_10 bufferpool order_bp4;
alter tablespace order_11 bufferpool order_bp4;
alter tablespace order_12 bufferpool order_bp4;
```

```
alter tablespace order_idx_01 bufferpool order_idx_bp1;
alter tablespace order_idx_02 bufferpool order_idx_bp1;
alter tablespace order_idx_03 bufferpool order_idx_bp1;
alter tablespace order_idx_04 bufferpool order_idx_bp2;
alter tablespace order_idx_05 bufferpool order_idx_bp2;
alter tablespace order_idx_06 bufferpool order_idx_bp2;
alter tablespace order_idx_07 bufferpool order_idx_bp3;
alter tablespace order_idx_08 bufferpool order_idx_bp3;
alter tablespace order_idx_09 bufferpool order_idx_bp3;
alter tablespace order_idx_10 bufferpool order_idx_bp4;
alter tablespace order_idx_11 bufferpool order_idx_bp4;
alter tablespace order_idx_12 bufferpool order_idx_bp4;
```

```
alter tablespace orderl_01 bufferpool orderl_bp1;
alter tablespace orderl_02 bufferpool orderl_bp1;
alter tablespace orderl_03 bufferpool orderl_bp1;
alter tablespace orderl_04 bufferpool orderl_bp2;
```



```

alter tablespace orderl_05 bufferpool orderl_bp2;
alter tablespace orderl_06 bufferpool orderl_bp2;
alter tablespace orderl_07 bufferpool orderl_bp3;
alter tablespace orderl_08 bufferpool orderl_bp3;
alter tablespace orderl_09 bufferpool orderl_bp3;
alter tablespace orderl_10 bufferpool orderl_bp4;
alter tablespace orderl_11 bufferpool orderl_bp4;
alter tablespace orderl_12 bufferpool orderl_bp4;

```

```

alter tablespace history_01 bufferpool history_bp1;
alter tablespace history_02 bufferpool history_bp1;
alter tablespace history_03 bufferpool history_bp1;
alter tablespace history_04 bufferpool history_bp2;
alter tablespace history_05 bufferpool history_bp2;
alter tablespace history_06 bufferpool history_bp2;
alter tablespace history_07 bufferpool history_bp3;
alter tablespace history_08 bufferpool history_bp3;
alter tablespace history_09 bufferpool history_bp3;
alter tablespace history_10 bufferpool history_bp4;
alter tablespace history_11 bufferpool history_bp4;
alter tablespace history_12 bufferpool history_bp4;

```

```

drop bufferpool IBMDEFAULT8K;
drop bufferpool IBMDEFAULT16K;

```

```

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;
alter bufferpool IBMDEFAULTBP size 1000000;
create bufferpool IBMDEFAULT8K size 500000 pagesize 8192;
create bufferpool IBMDEFAULT16K size 500000 pagesize 16384;
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.
-----

```

```

drop database tpcc;
create database tpcc on s: collate using identity;

```

CRIDX_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;

```

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;

```

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 cust_01
INDEX IN cust_idx_01
ORGANIZE BY KEY SEQUENCE (
  C_ID STARTING FROM 1 ENDING AT 3000,
  C_W_ID STARTING FROM 1 ENDING AT 1650,
  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 cust_02
INDEX IN cust_idx_02
ORGANIZE BY KEY SEQUENCE (
  C_ID STARTING FROM 1 ENDING AT 3000,
  C_W_ID STARTING FROM 1651 ENDING AT 3300,
  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 cust_03
INDEX IN cust_idx_03
ORGANIZE BY KEY SEQUENCE (
  C_ID STARTING FROM 1 ENDING AT 3000,
  C_W_ID STARTING FROM 3301 ENDING AT 4950,
  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,

```

```

        C_PAYMENT_CNT    INTEGER        NOT NULL
    )
    IN cust_04
    INDEX IN cust_idx_04
    ORGANIZE BY KEY SEQUENCE (
        C_ID STARTING FROM 1 ENDING AT 3000,
        C_W_ID STARTING FROM 4951 ENDING AT 6600,
        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 cust_05
INDEX IN cust_idx_05
ORGANIZE BY KEY SEQUENCE (
    C_ID STARTING FROM 1 ENDING AT 3000,
    C_W_ID STARTING FROM 6601 ENDING AT 8250,
    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 cust_06
INDEX IN cust_idx_06
ORGANIZE BY KEY SEQUENCE (
    C_ID STARTING FROM 1 ENDING AT 3000,
    C_W_ID STARTING FROM 8251 ENDING AT 9900,
    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 cust_07
INDEX IN cust_idx_07
ORGANIZE BY KEY SEQUENCE (
    C_ID STARTING FROM 1 ENDING AT 3000,
    C_W_ID STARTING FROM 9901 ENDING AT 11550,
    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 cust_08
INDEX IN cust_idx_08
ORGANIZE BY KEY SEQUENCE (
  C_ID STARTING FROM 1 ENDING AT 3000,
  C_W_ID STARTING FROM 11551 ENDING AT 13200,
  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 cust_09
INDEX IN cust_idx_09
ORGANIZE BY KEY SEQUENCE (
  C_ID STARTING FROM 1 ENDING AT 3000,
  C_W_ID STARTING FROM 13201 ENDING AT 14850,
  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 cust_10
INDEX IN cust_idx_10
ORGANIZE BY KEY SEQUENCE (
  C_ID STARTING FROM 1 ENDING AT 3000,
  C_W_ID STARTING FROM 14851 ENDING AT 16500,
  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 cust_11
INDEX IN cust_idx_11
ORGANIZE BY KEY SEQUENCE (
  C_ID STARTING FROM 1 ENDING AT 3000,
  C_W_ID STARTING FROM 16501 ENDING AT 18150,
  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 cust_12
INDEX IN cust_idx_12
ORGANIZE BY KEY SEQUENCE (
  C_ID STARTING FROM 1 ENDING AT 3000,
  C_W_ID STARTING FROM 18151 ENDING AT 19800,
  C_D_ID STARTING FROM 1 ENDING AT 10
)
ALLOW OVERFLOW;

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 dist_01
INDEX IN dist_01
ORGANIZE BY KEY SEQUENCE (
  D_ID STARTING FROM 1 ENDING AT 10,
  D_W_ID STARTING FROM 1 ENDING AT 1650
)
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 dist_02
INDEX IN dist_02
ORGANIZE BY KEY SEQUENCE (
  D_ID STARTING FROM 1 ENDING AT 10,
  D_W_ID STARTING FROM 1651 ENDING AT 3300
)
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 dist_03
    INDEX IN dist_03
    ORGANIZE BY KEY SEQUENCE (
        D_ID STARTING FROM 1 ENDING AT 10,
        D_W_ID STARTING FROM 3301 ENDING AT 4950
    )
    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 dist_04
INDEX IN dist_04
ORGANIZE BY KEY SEQUENCE (
    D_ID STARTING FROM 1 ENDING AT 10,
    D_W_ID STARTING FROM 4951 ENDING AT 6600
)
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 dist_05
INDEX IN dist_05
ORGANIZE BY KEY SEQUENCE (
    D_ID STARTING FROM 1 ENDING AT 10,
    D_W_ID STARTING FROM 6601 ENDING AT 8250
)
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 dist_06
INDEX IN dist_06
ORGANIZE BY KEY SEQUENCE (
    D_ID STARTING FROM 1 ENDING AT 10,
    D_W_ID STARTING FROM 8251 ENDING AT 9900
)
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 dist_07
INDEX IN dist_07
ORGANIZE BY KEY SEQUENCE (
    D_ID STARTING FROM 1 ENDING AT 10,
    D_W_ID STARTING FROM 9901 ENDING AT 11550
)
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 dist_08
    INDEX IN dist_08
    ORGANIZE BY KEY SEQUENCE (
        D_ID STARTING FROM 1 ENDING AT 10,
        D_W_ID STARTING FROM 11551 ENDING AT 13200
    )
    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 dist_09
INDEX IN dist_09
ORGANIZE BY KEY SEQUENCE (
    D_ID STARTING FROM 1 ENDING AT 10,
    D_W_ID STARTING FROM 13201 ENDING AT 14850
)
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 dist_10
INDEX IN dist_10
ORGANIZE BY KEY SEQUENCE (
    D_ID STARTING FROM 1 ENDING AT 10,
    D_W_ID STARTING FROM 14851 ENDING AT 16500

```

```

    )
    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 dist_11
INDEX IN dist_11
ORGANIZE BY KEY SEQUENCE (
    D_ID STARTING FROM 1 ENDING AT 10,
    D_W_ID STARTING FROM 16501 ENDING AT 18150
)
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 dist_12
INDEX IN dist_12
ORGANIZE BY KEY SEQUENCE (
    D_ID STARTING FROM 1 ENDING AT 10,
    D_W_ID STARTING FROM 18151 ENDING AT 19800
)
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 history_01
    INDEX IN history_01;
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 history_02
    INDEX IN history_02;
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 history_03
    INDEX IN history_03;
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 history_04
        INDEX IN history_04;
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 history_05
    INDEX IN history_05;
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 history_06
    INDEX IN history_06;
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 history_07
    INDEX IN history_07;
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 history_08
    INDEX IN history_08;
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 history_09
INDEX IN history_09;
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 history_10
INDEX IN history_10;
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 history_11
INDEX IN history_11;

```

```

    )
    IN history_11
    INDEX IN history_11;
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 history_12
INDEX IN history_12;
ALTER TABLE HISTORY12 APPEND ON;
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 item
INDEX IN item
ORGANIZE BY KEY SEQUENCE (
    I_ID STARTING FROM 1 ENDING AT 100000
)
ALLOW OVERFLOW;
ALTER TABLE ITEM LOCKSIZE TABLE;
connect reset;

```

CRTB_NEW_ORDER.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 norderA_01
INDEX IN norderA_01
ORGANIZE BY KEY SEQUENCE (
    NO_W_ID STARTING FROM 1 ENDING AT 1650,
    NO_D_ID STARTING FROM 1 ENDING AT 10,

```

```

        NO_O_ID STARTING FROM 1900 ENDING AT 3607
    )
    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 norderA_02
INDEX IN norderA_02
ORGANIZE BY KEY SEQUENCE (
    NO_W_ID STARTING FROM 1651 ENDING AT 3300,
    NO_D_ID STARTING FROM 1 ENDING AT 10,
    NO_O_ID STARTING FROM 1900 ENDING AT 3607
)
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 norderA_03
INDEX IN norderA_03
ORGANIZE BY KEY SEQUENCE (
    NO_W_ID STARTING FROM 3301 ENDING AT 4950,
    NO_D_ID STARTING FROM 1 ENDING AT 10,
    NO_O_ID STARTING FROM 1900 ENDING AT 3607
)
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 norderA_04
INDEX IN norderA_04
ORGANIZE BY KEY SEQUENCE (
    NO_W_ID STARTING FROM 4951 ENDING AT 6600,
    NO_D_ID STARTING FROM 1 ENDING AT 10,
    NO_O_ID STARTING FROM 1900 ENDING AT 3607
)
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 norderA_05
INDEX IN norderA_05
ORGANIZE BY KEY SEQUENCE (
    NO_W_ID STARTING FROM 6601 ENDING AT 8250,
    NO_D_ID STARTING FROM 1 ENDING AT 10,
    NO_O_ID STARTING FROM 1900 ENDING AT 3607
)
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 norderA_06
INDEX IN norderA_06
ORGANIZE BY KEY SEQUENCE (
    NO_W_ID STARTING FROM 8251 ENDING AT 9900,
    NO_D_ID STARTING FROM 1 ENDING AT 10,
    NO_O_ID STARTING FROM 1900 ENDING AT 3607
)
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 norderA_07
INDEX IN norderA_07
ORGANIZE BY KEY SEQUENCE (
    NO_W_ID STARTING FROM 9901 ENDING AT 11550,
    NO_D_ID STARTING FROM 1 ENDING AT 10,
    NO_O_ID STARTING FROM 1900 ENDING AT 3607
)
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 norderA_08
INDEX IN norderA_08
ORGANIZE BY KEY SEQUENCE (
    NO_W_ID STARTING FROM 11551 ENDING AT 13200,

```

```

        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 1900 ENDING AT 3607
    )
    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 norderA_09
INDEX IN norderA_09
ORGANIZE BY KEY SEQUENCE (
    NO_W_ID STARTING FROM 13201 ENDING AT 14850,
    NO_D_ID STARTING FROM 1 ENDING AT 10,
    NO_O_ID STARTING FROM 1900 ENDING AT 3607
)
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 norderA_10
INDEX IN norderA_10
ORGANIZE BY KEY SEQUENCE (
    NO_W_ID STARTING FROM 14851 ENDING AT 16500,
    NO_D_ID STARTING FROM 1 ENDING AT 10,
    NO_O_ID STARTING FROM 1900 ENDING AT 3607
)
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 norderA_11
INDEX IN norderA_11
ORGANIZE BY KEY SEQUENCE (
    NO_W_ID STARTING FROM 16501 ENDING AT 18150,
    NO_D_ID STARTING FROM 1 ENDING AT 10,
    NO_O_ID STARTING FROM 1900 ENDING AT 3607
)
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 norderA_12
INDEX IN norderA_12
ORGANIZE BY KEY SEQUENCE (
    NO_W_ID STARTING FROM 18151 ENDING AT 19800,
    NO_D_ID STARTING FROM 1 ENDING AT 10,
    NO_O_ID STARTING FROM 1900 ENDING AT 3607
)
ALLOW OVERFLOW;
connect reset;
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 norderB_01
INDEX IN norderB_01
ORGANIZE BY KEY SEQUENCE (
    NO_W_ID STARTING FROM 1 ENDING AT 1650,
    NO_D_ID STARTING FROM 1 ENDING AT 10,
    NO_O_ID STARTING FROM 3608 ENDING AT 5315
)
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 norderB_02
INDEX IN norderB_02
ORGANIZE BY KEY SEQUENCE (
    NO_W_ID STARTING FROM 1651 ENDING AT 3300,
    NO_D_ID STARTING FROM 1 ENDING AT 10,
    NO_O_ID STARTING FROM 3608 ENDING AT 5315
)
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 norderB_03
INDEX IN norderB_03
ORGANIZE BY KEY SEQUENCE (

```

```

        NO_W_ID STARTING FROM 3301 ENDING AT 4950,
        NO_D_ID STARTING FROM 1 ENDING AT 10,
        NO_O_ID STARTING FROM 3608 ENDING AT 5315
    )
    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 norderB_04
INDEX IN norderB_04
ORGANIZE BY KEY SEQUENCE (
    NO_W_ID STARTING FROM 4951 ENDING AT 6600,
    NO_D_ID STARTING FROM 1 ENDING AT 10,
    NO_O_ID STARTING FROM 3608 ENDING AT 5315
)
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 norderB_05
INDEX IN norderB_05
ORGANIZE BY KEY SEQUENCE (
    NO_W_ID STARTING FROM 6601 ENDING AT 8250,
    NO_D_ID STARTING FROM 1 ENDING AT 10,
    NO_O_ID STARTING FROM 3608 ENDING AT 5315
)
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 norderB_06
INDEX IN norderB_06
ORGANIZE BY KEY SEQUENCE (
    NO_W_ID STARTING FROM 8251 ENDING AT 9900,
    NO_D_ID STARTING FROM 1 ENDING AT 10,
    NO_O_ID STARTING FROM 3608 ENDING AT 5315
)
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 norderB_07
INDEX IN norderB_07
ORGANIZE BY KEY SEQUENCE (
    NO_W_ID STARTING FROM 9901 ENDING AT 11550,
    NO_D_ID STARTING FROM 1 ENDING AT 10,
    NO_O_ID STARTING FROM 3608 ENDING AT 5315
)
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 norderB_08
INDEX IN norderB_08
ORGANIZE BY KEY SEQUENCE (
    NO_W_ID STARTING FROM 11551 ENDING AT 13200,
    NO_D_ID STARTING FROM 1 ENDING AT 10,
    NO_O_ID STARTING FROM 3608 ENDING AT 5315
)
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 norderB_09
INDEX IN norderB_09
ORGANIZE BY KEY SEQUENCE (
    NO_W_ID STARTING FROM 13201 ENDING AT 14850,
    NO_D_ID STARTING FROM 1 ENDING AT 10,
    NO_O_ID STARTING FROM 3608 ENDING AT 5315
)
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 norderB_10
INDEX IN norderB_10

```

```

        ORGANIZE BY KEY SEQUENCE (
          NO_W_ID STARTING FROM 14851 ENDING AT 16500,
          NO_D_ID STARTING FROM 1 ENDING AT 10,
          NO_O_ID STARTING FROM 3608 ENDING AT 5315
        )
        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 norderB_11
INDEX IN norderB_11
ORGANIZE BY KEY SEQUENCE (
  NO_W_ID STARTING FROM 16501 ENDING AT 18150,
  NO_D_ID STARTING FROM 1 ENDING AT 10,
  NO_O_ID STARTING FROM 3608 ENDING AT 5315
)
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 norderB_12
INDEX IN norderB_12
ORGANIZE BY KEY SEQUENCE (
  NO_W_ID STARTING FROM 18151 ENDING AT 19800,
  NO_D_ID STARTING FROM 1 ENDING AT 10,
  NO_O_ID STARTING FROM 3608 ENDING AT 5315
)
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 orderl_01
INDEX IN orderl_01
ORGANIZE BY KEY SEQUENCE (
  OL_W_ID STARTING FROM 1 ENDING AT 1650,
  OL_D_ID STARTING FROM 1 ENDING AT 10,
  OL_O_ID STARTING FROM 0 ENDING AT 3607,
  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 orderl_02
INDEX IN orderl_02
ORGANIZE BY KEY SEQUENCE (
  OL_W_ID STARTING FROM 1651 ENDING AT 3300,
  OL_D_ID STARTING FROM 1 ENDING AT 10,
  OL_O_ID STARTING FROM 0 ENDING AT 3607,
  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 orderl_03
INDEX IN orderl_03
ORGANIZE BY KEY SEQUENCE (
  OL_W_ID STARTING FROM 3301 ENDING AT 4950,
  OL_D_ID STARTING FROM 1 ENDING AT 10,
  OL_O_ID STARTING FROM 0 ENDING AT 3607,
  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 orderl_04
INDEX IN orderl_04
ORGANIZE BY KEY SEQUENCE (
    OL_W_ID STARTING FROM 4951 ENDING AT 6600,
    OL_D_ID STARTING FROM 1 ENDING AT 10,
    OL_O_ID STARTING FROM 0 ENDING AT 3607,
    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 orderl_05
INDEX IN orderl_05
ORGANIZE BY KEY SEQUENCE (
    OL_W_ID STARTING FROM 6601 ENDING AT 8250,
    OL_D_ID STARTING FROM 1 ENDING AT 10,
    OL_O_ID STARTING FROM 0 ENDING AT 3607,
    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 orderl_06
INDEX IN orderl_06
ORGANIZE BY KEY SEQUENCE (
    OL_W_ID STARTING FROM 8251 ENDING AT 9900,
    OL_D_ID STARTING FROM 1 ENDING AT 10,
    OL_O_ID STARTING FROM 0 ENDING AT 3607,
    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 orderl_07
INDEX IN orderl_07
ORGANIZE BY KEY SEQUENCE (
    OL_W_ID STARTING FROM 9901 ENDING AT 11550,
    OL_D_ID STARTING FROM 1 ENDING AT 10,
    OL_O_ID STARTING FROM 0 ENDING AT 3607,
    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 orderl_08
INDEX IN orderl_08

```

```

        ORGANIZE BY KEY SEQUENCE (
            OL_W_ID STARTING FROM 11551 ENDING AT 13200,
            OL_D_ID STARTING FROM 1 ENDING AT 10,
            OL_O_ID STARTING FROM 0 ENDING AT 3607,
            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 orderl_09
INDEX IN orderl_09
ORGANIZE BY KEY SEQUENCE (
    OL_W_ID STARTING FROM 13201 ENDING AT 14850,
    OL_D_ID STARTING FROM 1 ENDING AT 10,
    OL_O_ID STARTING FROM 0 ENDING AT 3607,
    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 orderl_10
INDEX IN orderl_10
ORGANIZE BY KEY SEQUENCE (
    OL_W_ID STARTING FROM 14851 ENDING AT 16500,
    OL_D_ID STARTING FROM 1 ENDING AT 10,
    OL_O_ID STARTING FROM 0 ENDING AT 3607,
    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 orderl_11
INDEX IN orderl_11
ORGANIZE BY KEY SEQUENCE (
    OL_W_ID STARTING FROM 16501 ENDING AT 18150,
    OL_D_ID STARTING FROM 1 ENDING AT 10,
    OL_O_ID STARTING FROM 0 ENDING AT 3607,
    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 orderl_12
INDEX IN orderl_12
ORGANIZE BY KEY SEQUENCE (
    OL_W_ID STARTING FROM 18151 ENDING AT 19800,
    OL_D_ID STARTING FROM 1 ENDING AT 10,
    OL_O_ID STARTING FROM 0 ENDING AT 3607,
    OL_NUMBER STARTING FROM 1 ENDING AT 15
)
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 order_01
INDEX IN order_idx_01
ORGANIZE BY KEY SEQUENCE (
  O_ID STARTING FROM 0 ENDING AT 103000,
  O_W_ID STARTING FROM 1 ENDING AT 10,
  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 order_02
INDEX IN order_idx_02
ORGANIZE BY KEY SEQUENCE (
  O_ID STARTING FROM 0 ENDING AT 103000,
  O_W_ID STARTING FROM 11 ENDING AT 20,
  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 order_03
INDEX IN order_idx_03
ORGANIZE BY KEY SEQUENCE (
  O_ID STARTING FROM 0 ENDING AT 103000,
  O_W_ID STARTING FROM 21 ENDING AT 30,
  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 order_04
INDEX IN order_idx_04
ORGANIZE BY KEY SEQUENCE (
  O_ID STARTING FROM 0 ENDING AT 103000,
  O_W_ID STARTING FROM 31 ENDING AT 40,
  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 order_05
INDEX IN order_idx_05
ORGANIZE BY KEY SEQUENCE (
  O_ID STARTING FROM 0 ENDING AT 103000,
  O_W_ID STARTING FROM 41 ENDING AT 50,
  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 order_06
INDEX IN order_idx_06

```

```

        ORGANIZE BY KEY SEQUENCE (
            O_ID STARTING FROM 0 ENDING AT 103000,
            O_W_ID STARTING FROM 51 ENDING AT 60,
            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 order_07
INDEX IN order_idx_07
ORGANIZE BY KEY SEQUENCE (
    O_ID STARTING FROM 0 ENDING AT 103000,
    O_W_ID STARTING FROM 61 ENDING AT 70,
    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 order_08
INDEX IN order_idx_08
ORGANIZE BY KEY SEQUENCE (
    O_ID STARTING FROM 0 ENDING AT 103000,
    O_W_ID STARTING FROM 71 ENDING AT 80,
    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 order_09
    INDEX IN order_idx_09
    ORGANIZE BY KEY SEQUENCE (
        O_ID STARTING FROM 0 ENDING AT 103000,
        O_W_ID STARTING FROM 81 ENDING AT 90,
        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 order_10
INDEX IN order_idx_10
ORGANIZE BY KEY SEQUENCE (
    O_ID STARTING FROM 0 ENDING AT 103000,
    O_W_ID STARTING FROM 91 ENDING AT 100,
    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 order_11
INDEX IN order_idx_11
ORGANIZE BY KEY SEQUENCE (
    O_ID STARTING FROM 0 ENDING AT 103000,
    O_W_ID STARTING FROM 101 ENDING AT 110,
    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 order_12
INDEX IN order_idx_12
ORGANIZE BY KEY SEQUENCE (
    O_ID STARTING FROM 0 ENDING AT 103000,
    O_W_ID STARTING FROM 111 ENDING AT 120,
    O_D_ID STARTING FROM 1 ENDING AT 10
)
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 stock_01
INDEX IN stock_01
ORGANIZE BY KEY SEQUENCE (
    S_I_ID STARTING FROM 1 ENDING AT 100000,
    S_W_ID STARTING FROM 1 ENDING AT 1650
)
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 stock_02
INDEX IN stock_02
ORGANIZE BY KEY SEQUENCE (
    S_I_ID STARTING FROM 1 ENDING AT 100000,
    S_W_ID STARTING FROM 1651 ENDING AT 3300
)
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 stock_03
INDEX IN stock_03
ORGANIZE BY KEY SEQUENCE (
    S_I_ID STARTING FROM 1 ENDING AT 100000,
    S_W_ID STARTING FROM 3301 ENDING AT 4950
)
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 stock_04
INDEX IN stock_04
ORGANIZE BY KEY SEQUENCE (
  S_I_ID STARTING FROM 1 ENDING AT 100000,
  S_W_ID STARTING FROM 4951 ENDING AT 6600
)
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 stock_05
INDEX IN stock_05
ORGANIZE BY KEY SEQUENCE (
  S_I_ID STARTING FROM 1 ENDING AT 100000,
  S_W_ID STARTING FROM 6601 ENDING AT 8250
)
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 stock_06
INDEX IN stock_06
ORGANIZE BY KEY SEQUENCE (
  S_I_ID STARTING FROM 1 ENDING AT 100000,
  S_W_ID STARTING FROM 8251 ENDING AT 9900
)
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 stock_07
INDEX IN stock_07
ORGANIZE BY KEY SEQUENCE (
  S_I_ID STARTING FROM 1 ENDING AT 100000,
  S_W_ID STARTING FROM 9901 ENDING AT 11550
)
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 stock_08
INDEX IN stock_08
ORGANIZE BY KEY SEQUENCE (
  S_I_ID STARTING FROM 1 ENDING AT 100000,
  S_W_ID STARTING FROM 11551 ENDING AT 13200
)
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 stock_09
INDEX IN stock_09
ORGANIZE BY KEY SEQUENCE (
  S_I_ID STARTING FROM 1 ENDING AT 100000,
  S_W_ID STARTING FROM 13201 ENDING AT 14850
)
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 stock_10
INDEX IN stock_10
ORGANIZE BY KEY SEQUENCE (
  S_I_ID STARTING FROM 1 ENDING AT 100000,
  S_W_ID STARTING FROM 14851 ENDING AT 16500
)
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 stock_11
INDEX IN stock_11
ORGANIZE BY KEY SEQUENCE (
  S_I_ID STARTING FROM 1 ENDING AT 100000,
  S_W_ID STARTING FROM 16501 ENDING AT 18150
)
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 stock_12
INDEX IN stock_12
ORGANIZE BY KEY SEQUENCE (
    S_I_ID STARTING FROM 1 ENDING AT 100000,
    S_W_ID STARTING FROM 18151 ENDING AT 19800
)
ALLOW OVERFLOW;

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 ware_01
INDEX IN ware_01
ORGANIZE BY KEY SEQUENCE (
    W_ID STARTING FROM 1 ENDING AT 1650
)
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 ware_02
INDEX IN ware_02
ORGANIZE BY KEY SEQUENCE (
    W_ID STARTING FROM 1651 ENDING AT 3300
)
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 ware_03
INDEX IN ware_03
ORGANIZE BY KEY SEQUENCE (
    W_ID STARTING FROM 3301 ENDING AT 4950
)
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 ware_04
INDEX IN ware_04
ORGANIZE BY KEY SEQUENCE (
    W_ID STARTING FROM 4951 ENDING AT 6600
)
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 ware_05
INDEX IN ware_05
ORGANIZE BY KEY SEQUENCE (
    W_ID STARTING FROM 6601 ENDING AT 8250
)
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 ware_06
INDEX IN ware_06
ORGANIZE BY KEY SEQUENCE (
    W_ID STARTING FROM 8251 ENDING AT 9900
)
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 ware_07
INDEX IN ware_07
ORGANIZE BY KEY SEQUENCE (
    W_ID STARTING FROM 9901 ENDING AT 11550
)

```

```

)
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 ware_08
INDEX IN ware_08
ORGANIZE BY KEY SEQUENCE (
    W_ID STARTING FROM 11551 ENDING AT 13200
)
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 ware_09
INDEX IN ware_09
ORGANIZE BY KEY SEQUENCE (
    W_ID STARTING FROM 13201 ENDING AT 14850
)
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 ware_10
        INDEX IN ware_10
        ORGANIZE BY KEY SEQUENCE (
            W_ID STARTING FROM 14851 ENDING AT 16500
        )
        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 ware_11
INDEX IN ware_11
ORGANIZE BY KEY SEQUENCE (
    W_ID STARTING FROM 16501 ENDING AT 18150
)
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 ware_12
INDEX IN ware_12
ORGANIZE BY KEY SEQUENCE (
    W_ID STARTING FROM 18151 ENDING AT 19800
)
ALLOW OVERFLOW;
connect reset;

```

create_tablespace.ddl

```
connect to tpcc;
```

```

drop tablespace ware_01;
create regular tablespace ware_01
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\ware\01' 8800,

```

```

        DEVICE 's:\Containers\ware\02' 8800,
        DEVICE 's:\Containers\ware\03' 8800)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace ware_02;
create regular tablespace ware_02
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\ware\04' 8800,
        DEVICE 's:\Containers\ware\05' 8800,
        DEVICE 's:\Containers\ware\06' 8800)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace ware_03;
create regular tablespace ware_03
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\ware\07' 8800,
        DEVICE 's:\Containers\ware\08' 8800,
        DEVICE 's:\Containers\ware\09' 8800)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace ware_04;
create regular tablespace ware_04
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\ware\10' 8800,
        DEVICE 's:\Containers\ware\11' 8800,
        DEVICE 's:\Containers\ware\12' 8800)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace ware_05;
create regular tablespace ware_05
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\ware\13' 8800,
        DEVICE 's:\Containers\ware\14' 8800,
        DEVICE 's:\Containers\ware\15' 8800)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace ware_06;
create regular tablespace ware_06
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\ware\16' 8800,
        DEVICE 's:\Containers\ware\17' 8800,
        DEVICE 's:\Containers\ware\18' 8800)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace ware_07;
create regular tablespace ware_07
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\ware\19' 8800,
        DEVICE 's:\Containers\ware\20' 8800,
        DEVICE 's:\Containers\ware\21' 8800)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

```



```

drop tablespace ware_08;
create regular tablespace ware_08
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\ware\22' 8800,
          DEVICE 's:\Containers\ware\23' 8800,
          DEVICE 's:\Containers\ware\24' 8800)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace ware_09;
create regular tablespace ware_09
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\ware\25' 8800,
          DEVICE 's:\Containers\ware\26' 8800,
          DEVICE 's:\Containers\ware\27' 8800)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace ware_10;
create regular tablespace ware_10
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\ware\28' 8800,
          DEVICE 's:\Containers\ware\29' 8800,
          DEVICE 's:\Containers\ware\30' 8800)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace ware_11;
create regular tablespace ware_11
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\ware\31' 8800,
          DEVICE 's:\Containers\ware\32' 8800,
          DEVICE 's:\Containers\ware\33' 8800)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace ware_12;
create regular tablespace ware_12
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\ware\34' 8800,
          DEVICE 's:\Containers\ware\35' 8800,
          DEVICE 's:\Containers\ware\36' 8800)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace dist_01;
create regular tablespace dist_01
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\dist\01' 8800,
          DEVICE 's:\Containers\dist\02' 8800,
          DEVICE 's:\Containers\dist\03' 8800)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace dist_02;
create regular tablespace dist_02
    PAGESIZE 4K managed by database

```

```

    using (DEVICE 's:\Containers\dist\04' 8800,
          DEVICE 's:\Containers\dist\05' 8800,
          DEVICE 's:\Containers\dist\06' 8800)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace dist_03;
create regular tablespace dist_03
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\dist\07' 8800,
          DEVICE 's:\Containers\dist\08' 8800,
          DEVICE 's:\Containers\dist\09' 8800)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace dist_04;
create regular tablespace dist_04
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\dist\10' 8800,
          DEVICE 's:\Containers\dist\11' 8800,
          DEVICE 's:\Containers\dist\12' 8800)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace dist_05;
create regular tablespace dist_05
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\dist\13' 8800,
          DEVICE 's:\Containers\dist\14' 8800,
          DEVICE 's:\Containers\dist\15' 8800)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace dist_06;
create regular tablespace dist_06
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\dist\16' 8800,
          DEVICE 's:\Containers\dist\17' 8800,
          DEVICE 's:\Containers\dist\18' 8800)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace dist_07;
create regular tablespace dist_07
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\dist\19' 8800,
          DEVICE 's:\Containers\dist\20' 8800,
          DEVICE 's:\Containers\dist\21' 8800)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace dist_08;
create regular tablespace dist_08
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\dist\22' 8800,
          DEVICE 's:\Containers\dist\23' 8800,
          DEVICE 's:\Containers\dist\24' 8800)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

```

```

drop tablespace dist_09;
create regular tablespace dist_09
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\dist\25' 8800,
          DEVICE 's:\Containers\dist\26' 8800,
          DEVICE 's:\Containers\dist\27' 8800)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace dist_10;
create regular tablespace dist_10
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\dist\28' 8800,
          DEVICE 's:\Containers\dist\29' 8800,
          DEVICE 's:\Containers\dist\30' 8800)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace dist_11;
create regular tablespace dist_11
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\dist\31' 8800,
          DEVICE 's:\Containers\dist\32' 8800,
          DEVICE 's:\Containers\dist\33' 8800)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace dist_12;
create regular tablespace dist_12
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\dist\34' 8800,
          DEVICE 's:\Containers\dist\35' 8800,
          DEVICE 's:\Containers\dist\36' 8800)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

-- ITEM -- 1GB/container
drop tablespace item;
create regular tablespace item
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\item\01' 8800,
          DEVICE 's:\Containers\item\02' 8800,
          DEVICE 's:\Containers\item\03' 8800,
          DEVICE 's:\Containers\item\04' 8800,
          DEVICE 's:\Containers\item\05' 8800,
          DEVICE 's:\Containers\item\06' 8800,
          DEVICE 's:\Containers\item\07' 8800,
          DEVICE 's:\Containers\item\08' 8800,
          DEVICE 's:\Containers\item\09' 8800,
          DEVICE 's:\Containers\item\10' 8800,
          DEVICE 's:\Containers\item\11' 8800,
          DEVICE 's:\Containers\item\12' 8800,
          DEVICE 's:\Containers\item\13' 8800,
          DEVICE 's:\Containers\item\14' 8800,
          DEVICE 's:\Containers\item\15' 8800,
          DEVICE 's:\Containers\item\16' 8800,
          DEVICE 's:\Containers\item\17' 8800,
          DEVICE 's:\Containers\item\18' 8800,
          DEVICE 's:\Containers\item\19' 8800,
          DEVICE 's:\Containers\item\20' 8800,
          DEVICE 's:\Containers\item\21' 8800,
          DEVICE 's:\Containers\item\22' 8800,
          DEVICE 's:\Containers\item\23' 8800,
          DEVICE 's:\Containers\item\24' 8800,
          DEVICE 's:\Containers\item\25' 8800,
          DEVICE 's:\Containers\item\26' 8800,
          DEVICE 's:\Containers\item\27' 8800,
          DEVICE 's:\Containers\item\28' 8800,
          DEVICE 's:\Containers\item\29' 8800,
          DEVICE 's:\Containers\item\30' 8800,
          DEVICE 's:\Containers\item\31' 8800,
          DEVICE 's:\Containers\item\32' 8800,
          DEVICE 's:\Containers\item\33' 8800,
          DEVICE 's:\Containers\item\34' 8800,
          DEVICE 's:\Containers\item\35' 8800,
          DEVICE 's:\Containers\item\36' 8800)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace stock_01;
create regular tablespace stock_01
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\stock\01' 4961452,
          DEVICE 's:\Containers\stock\02' 4961452,
          DEVICE 's:\Containers\stock\03' 4961452)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace stock_02;
create regular tablespace stock_02
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\stock\04' 4961452,
          DEVICE 's:\Containers\stock\05' 4961452,
          DEVICE 's:\Containers\stock\06' 4961452)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace stock_03;
create regular tablespace stock_03
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\stock\07' 4961452,
          DEVICE 's:\Containers\stock\08' 4961452,
          DEVICE 's:\Containers\stock\09' 4961452)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace stock_04;
create regular tablespace stock_04
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\stock\10' 4961452,
          DEVICE 's:\Containers\stock\11' 4961452,
          DEVICE 's:\Containers\stock\12' 4961452)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

```

```

drop tablespace stock_05;
create regular tablespace stock_05
  PAGESIZE 4K managed by database
  using (DEVICE 's:\Containers\stock\13' 4961452,
        DEVICE 's:\Containers\stock\14' 4961452,
        DEVICE 's:\Containers\stock\15' 4961452)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULTBP;

drop tablespace stock_06;
create regular tablespace stock_06
  PAGESIZE 4K managed by database
  using (DEVICE 's:\Containers\stock\16' 4961452,
        DEVICE 's:\Containers\stock\17' 4961452,
        DEVICE 's:\Containers\stock\18' 4961452)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULTBP;

drop tablespace stock_07;
create regular tablespace stock_07
  PAGESIZE 4K managed by database
  using (DEVICE 's:\Containers\stock\19' 4961452,
        DEVICE 's:\Containers\stock\20' 4961452,
        DEVICE 's:\Containers\stock\21' 4961452)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULTBP;

drop tablespace stock_08;
create regular tablespace stock_08
  PAGESIZE 4K managed by database
  using (DEVICE 's:\Containers\stock\22' 4961452,
        DEVICE 's:\Containers\stock\23' 4961452,
        DEVICE 's:\Containers\stock\24' 4961452)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULTBP;

drop tablespace stock_09;
create regular tablespace stock_09
  PAGESIZE 4K managed by database
  using (DEVICE 's:\Containers\stock\25' 4961452,
        DEVICE 's:\Containers\stock\26' 4961452,
        DEVICE 's:\Containers\stock\27' 4961452)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULTBP;

drop tablespace stock_10;
create regular tablespace stock_10
  PAGESIZE 4K managed by database
  using (DEVICE 's:\Containers\stock\28' 4961452,
        DEVICE 's:\Containers\stock\29' 4961452,
        DEVICE 's:\Containers\stock\30' 4961452)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULTBP;

drop tablespace stock_11;
create regular tablespace stock_11
  PAGESIZE 4K managed by database
  using (DEVICE 's:\Containers\stock\31' 4961452,

```

```

        DEVICE 's:\Containers\stock\32' 4961452,
        DEVICE 's:\Containers\stock\33' 4961452)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULTBP;

drop tablespace stock_12;
create regular tablespace stock_12
  PAGESIZE 4K managed by database
  using (DEVICE 's:\Containers\stock\34' 4961452,
        DEVICE 's:\Containers\stock\35' 4961452,
        DEVICE 's:\Containers\stock\36' 4961452)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULTBP;

drop tablespace cust_01;
create regular tablespace cust_01
  PAGESIZE 4K managed by database
  using (DEVICE 's:\Containers\cust\01' 3573248,
        DEVICE 's:\Containers\cust\02' 3573248,
        DEVICE 's:\Containers\cust\03' 3573248)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULTBP;

drop tablespace cust_02;
create regular tablespace cust_02
  PAGESIZE 4K managed by database
  using (DEVICE 's:\Containers\cust\04' 3573248,
        DEVICE 's:\Containers\cust\05' 3573248,
        DEVICE 's:\Containers\cust\06' 3573248)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULTBP;

drop tablespace cust_03;
create regular tablespace cust_03
  PAGESIZE 4K managed by database
  using (DEVICE 's:\Containers\cust\07' 3573248,
        DEVICE 's:\Containers\cust\08' 3573248,
        DEVICE 's:\Containers\cust\09' 3573248)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULTBP;

drop tablespace cust_04;
create regular tablespace cust_04
  PAGESIZE 4K managed by database
  using (DEVICE 's:\Containers\cust\10' 3573248,
        DEVICE 's:\Containers\cust\11' 3573248,
        DEVICE 's:\Containers\cust\12' 3573248)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULTBP;

drop tablespace cust_05;
create regular tablespace cust_05
  PAGESIZE 4K managed by database
  using (DEVICE 's:\Containers\cust\13' 3573248,
        DEVICE 's:\Containers\cust\14' 3573248,
        DEVICE 's:\Containers\cust\15' 3573248)
  EXTENTSIZE 256 PREFETCHSIZE 0

```

```

        BUFFERPOOL IBMDEFAULTBP;

drop tablespace cust_06;
create regular tablespace cust_06
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\cust\16' 3573248,
           DEVICE 's:\Containers\cust\17' 3573248,
           DEVICE 's:\Containers\cust\18' 3573248)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace cust_07;
create regular tablespace cust_07
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\cust\19' 3573248,
           DEVICE 's:\Containers\cust\20' 3573248,
           DEVICE 's:\Containers\cust\21' 3573248)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace cust_08;
create regular tablespace cust_08
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\cust\22' 3573248,
           DEVICE 's:\Containers\cust\23' 3573248,
           DEVICE 's:\Containers\cust\24' 3573248)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace cust_09;
create regular tablespace cust_09
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\cust\25' 3573248,
           DEVICE 's:\Containers\cust\26' 3573248,
           DEVICE 's:\Containers\cust\27' 3573248)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace cust_10;
create regular tablespace cust_10
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\cust\28' 3573248,
           DEVICE 's:\Containers\cust\29' 3573248,
           DEVICE 's:\Containers\cust\30' 3573248)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace cust_11;
create regular tablespace cust_11
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\cust\31' 3573248,
           DEVICE 's:\Containers\cust\32' 3573248,
           DEVICE 's:\Containers\cust\33' 3573248)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace cust_12;
create regular tablespace cust_12
    PAGESIZE 4K managed by database

```

```

    using (DEVICE 's:\Containers\cust\34' 3573248,
           DEVICE 's:\Containers\cust\35' 3573248,
           DEVICE 's:\Containers\cust\36' 3573248)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace cust_idx_01;
create regular tablespace cust_idx_01
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\cust_idx\01' 108032,
           DEVICE 's:\Containers\cust_idx\02' 108032,
           DEVICE 's:\Containers\cust_idx\03' 108032)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULT8K;

drop tablespace cust_idx_02;
create regular tablespace cust_idx_02
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\cust_idx\04' 108032,
           DEVICE 's:\Containers\cust_idx\05' 108032,
           DEVICE 's:\Containers\cust_idx\06' 108032)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULT8K;

drop tablespace cust_idx_03;
create regular tablespace cust_idx_03
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\cust_idx\07' 108032,
           DEVICE 's:\Containers\cust_idx\08' 108032,
           DEVICE 's:\Containers\cust_idx\09' 108032)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULT8K;

drop tablespace cust_idx_04;
create regular tablespace cust_idx_04
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\cust_idx\10' 108032,
           DEVICE 's:\Containers\cust_idx\11' 108032,
           DEVICE 's:\Containers\cust_idx\12' 108032)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULT8K;

drop tablespace cust_idx_05;
create regular tablespace cust_idx_05
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\cust_idx\13' 108032,
           DEVICE 's:\Containers\cust_idx\14' 108032,
           DEVICE 's:\Containers\cust_idx\15' 108032)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULT8K;

drop tablespace cust_idx_06;
create regular tablespace cust_idx_06
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\cust_idx\16' 108032,
           DEVICE 's:\Containers\cust_idx\17' 108032,
           DEVICE 's:\Containers\cust_idx\18' 108032)

```

```

EXTENTSIZE 256 PREFETCHSIZE 0
BUFFERPOOL IBMDEFAULT8K;

drop tablespace cust_idx_07;
create regular tablespace cust_idx_07
  PAGESIZE 8K managed by database
  using (DEVICE 's:\Containers\cust_idx\19' 108032,
        DEVICE 's:\Containers\cust_idx\20' 108032,
        DEVICE 's:\Containers\cust_idx\21' 108032)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULT8K;

drop tablespace cust_idx_08;
create regular tablespace cust_idx_08
  PAGESIZE 8K managed by database
  using (DEVICE 's:\Containers\cust_idx\22' 108032,
        DEVICE 's:\Containers\cust_idx\23' 108032,
        DEVICE 's:\Containers\cust_idx\24' 108032)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULT8K;

drop tablespace cust_idx_09;
create regular tablespace cust_idx_09
  PAGESIZE 8K managed by database
  using (DEVICE 's:\Containers\cust_idx\25' 108032,
        DEVICE 's:\Containers\cust_idx\26' 108032,
        DEVICE 's:\Containers\cust_idx\27' 108032)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULT8K;

drop tablespace cust_idx_10;
create regular tablespace cust_idx_10
  PAGESIZE 8K managed by database
  using (DEVICE 's:\Containers\cust_idx\28' 108032,
        DEVICE 's:\Containers\cust_idx\29' 108032,
        DEVICE 's:\Containers\cust_idx\30' 108032)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULT8K;

drop tablespace cust_idx_11;
create regular tablespace cust_idx_11
  PAGESIZE 8K managed by database
  using (DEVICE 's:\Containers\cust_idx\31' 108032,
        DEVICE 's:\Containers\cust_idx\32' 108032,
        DEVICE 's:\Containers\cust_idx\33' 108032)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULT8K;

drop tablespace cust_idx_12;
create regular tablespace cust_idx_12
  PAGESIZE 8K managed by database
  using (DEVICE 's:\Containers\cust_idx\34' 108032,
        DEVICE 's:\Containers\cust_idx\35' 108032,
        DEVICE 's:\Containers\cust_idx\36' 108032)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULT8K;

```

```

drop tablespace norderA_01;
create regular tablespace norderA_01
  PAGESIZE 4K managed by database
  using (DEVICE 's:\Containers\norderA\01' 59740,
        DEVICE 's:\Containers\norderA\02' 59740,
        DEVICE 's:\Containers\norderA\03' 59740)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULTBP;

drop tablespace norderA_02;
create regular tablespace norderA_02
  PAGESIZE 4K managed by database
  using (DEVICE 's:\Containers\norderA\04' 59740,
        DEVICE 's:\Containers\norderA\05' 59740,
        DEVICE 's:\Containers\norderA\06' 59740)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULTBP;

drop tablespace norderA_03;
create regular tablespace norderA_03
  PAGESIZE 4K managed by database
  using (DEVICE 's:\Containers\norderA\07' 59740,
        DEVICE 's:\Containers\norderA\08' 59740,
        DEVICE 's:\Containers\norderA\09' 59740)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULTBP;

drop tablespace norderA_04;
create regular tablespace norderA_04
  PAGESIZE 4K managed by database
  using (DEVICE 's:\Containers\norderA\10' 59740,
        DEVICE 's:\Containers\norderA\11' 59740,
        DEVICE 's:\Containers\norderA\12' 59740)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULTBP;

drop tablespace norderA_05;
create regular tablespace norderA_05
  PAGESIZE 4K managed by database
  using (DEVICE 's:\Containers\norderA\13' 59740,
        DEVICE 's:\Containers\norderA\14' 59740,
        DEVICE 's:\Containers\norderA\15' 59740)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULTBP;

drop tablespace norderA_06;
create regular tablespace norderA_06
  PAGESIZE 4K managed by database
  using (DEVICE 's:\Containers\norderA\16' 59740,
        DEVICE 's:\Containers\norderA\17' 59740,
        DEVICE 's:\Containers\norderA\18' 59740)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULTBP;

drop tablespace norderA_07;
create regular tablespace norderA_07
  PAGESIZE 4K managed by database
  using (DEVICE 's:\Containers\norderA\19' 59740,
        DEVICE 's:\Containers\norderA\20' 59740,

```

```

        DEVICE 's:\Containers\norderA\21' 59740)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace norderA_08;
create regular tablespace norderA_08
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\norderA\22' 59740,
        DEVICE 's:\Containers\norderA\23' 59740,
        DEVICE 's:\Containers\norderA\24' 59740)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace norderA_09;
create regular tablespace norderA_09
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\norderA\25' 59740,
        DEVICE 's:\Containers\norderA\26' 59740,
        DEVICE 's:\Containers\norderA\27' 59740)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace norderA_10;
create regular tablespace norderA_10
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\norderA\28' 59740,
        DEVICE 's:\Containers\norderA\29' 59740,
        DEVICE 's:\Containers\norderA\30' 59740)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace norderA_11;
create regular tablespace norderA_11
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\norderA\31' 59740,
        DEVICE 's:\Containers\norderA\32' 59740,
        DEVICE 's:\Containers\norderA\33' 59740)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace norderA_12;
create regular tablespace norderA_12
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\norderA\34' 59740,
        DEVICE 's:\Containers\norderA\35' 59740,
        DEVICE 's:\Containers\norderA\36' 59740)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace norderB_01;
create regular tablespace norderB_01
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\norderB\01' 59740,
        DEVICE 's:\Containers\norderB\02' 59740,
        DEVICE 's:\Containers\norderB\03' 59740)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

```

```

drop tablespace norderB_02;
create regular tablespace norderB_02
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\norderB\04' 59740,
        DEVICE 's:\Containers\norderB\05' 59740,
        DEVICE 's:\Containers\norderB\06' 59740)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace norderB_03;
create regular tablespace norderB_03
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\norderB\07' 59740,
        DEVICE 's:\Containers\norderB\08' 59740,
        DEVICE 's:\Containers\norderB\09' 59740)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace norderB_04;
create regular tablespace norderB_04
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\norderB\10' 59740,
        DEVICE 's:\Containers\norderB\11' 59740,
        DEVICE 's:\Containers\norderB\12' 59740)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace norderB_05;
create regular tablespace norderB_05
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\norderB\13' 59740,
        DEVICE 's:\Containers\norderB\14' 59740,
        DEVICE 's:\Containers\norderB\15' 59740)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace norderB_06;
create regular tablespace norderB_06
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\norderB\16' 59740,
        DEVICE 's:\Containers\norderB\17' 59740,
        DEVICE 's:\Containers\norderB\18' 59740)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace norderB_07;
create regular tablespace norderB_07
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\norderB\19' 59740,
        DEVICE 's:\Containers\norderB\20' 59740,
        DEVICE 's:\Containers\norderB\21' 59740)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace norderB_08;
create regular tablespace norderB_08
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\norderB\22' 59740,

```

```

        DEVICE 's:\Containers\norderB\23' 59740,
        DEVICE 's:\Containers\norderB\24' 59740)
EXTENTSIZ 256 PREFETCHSIZE 0
BUFFERPOOL IBMDEFAULTBP;

drop tablespace norderB_09;
create regular tablespace norderB_09
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\norderB\25' 59740,
        DEVICE 's:\Containers\norderB\26' 59740,
        DEVICE 's:\Containers\norderB\27' 59740)
    EXTENTSIZ 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace norderB_10;
create regular tablespace norderB_10
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\norderB\28' 59740,
        DEVICE 's:\Containers\norderB\29' 59740,
        DEVICE 's:\Containers\norderB\30' 59740)
    EXTENTSIZ 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace norderB_11;
create regular tablespace norderB_11
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\norderB\31' 59740,
        DEVICE 's:\Containers\norderB\32' 59740,
        DEVICE 's:\Containers\norderB\33' 59740)
    EXTENTSIZ 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace norderB_12;
create regular tablespace norderB_12
    PAGESIZE 4K managed by database
    using (DEVICE 's:\Containers\norderB\34' 59740,
        DEVICE 's:\Containers\norderB\35' 59740,
        DEVICE 's:\Containers\norderB\36' 59740)
    EXTENTSIZ 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULTBP;

drop tablespace order_01;
create regular tablespace order_01
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\order\01' 118330,
        DEVICE 's:\Containers\order\02' 118330,
        DEVICE 's:\Containers\order\03' 118330)
    EXTENTSIZ 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULT8K;

drop tablespace order_02;
create regular tablespace order_02
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\order\04' 118330,
        DEVICE 's:\Containers\order\05' 118330,
        DEVICE 's:\Containers\order\06' 118330)
    EXTENTSIZ 256 PREFETCHSIZE 0

```

```

    BUFFERPOOL IBMDEFAULT8K;

drop tablespace order_03;
create regular tablespace order_03
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\order\07' 118330,
        DEVICE 's:\Containers\order\08' 118330,
        DEVICE 's:\Containers\order\09' 118330)
    EXTENTSIZ 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULT8K;

drop tablespace order_04;
create regular tablespace order_04
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\order\10' 118330,
        DEVICE 's:\Containers\order\11' 118330,
        DEVICE 's:\Containers\order\12' 118330)
    EXTENTSIZ 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULT8K;

drop tablespace order_05;
create regular tablespace order_05
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\order\13' 118330,
        DEVICE 's:\Containers\order\14' 118330,
        DEVICE 's:\Containers\order\15' 118330)
    EXTENTSIZ 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULT8K;

drop tablespace order_06;
create regular tablespace order_06
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\order\16' 118330,
        DEVICE 's:\Containers\order\17' 118330,
        DEVICE 's:\Containers\order\18' 118330)
    EXTENTSIZ 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULT8K;

drop tablespace order_07;
create regular tablespace order_07
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\order\19' 118330,
        DEVICE 's:\Containers\order\20' 118330,
        DEVICE 's:\Containers\order\21' 118330)
    EXTENTSIZ 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULT8K;

drop tablespace order_08;
create regular tablespace order_08
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\order\22' 118330,
        DEVICE 's:\Containers\order\23' 118330,
        DEVICE 's:\Containers\order\24' 118330)
    EXTENTSIZ 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULT8K;

drop tablespace order_09;
create regular tablespace order_09
    PAGESIZE 8K managed by database

```

```

        using (DEVICE 's:\Containers\order\25' 118330,
              DEVICE 's:\Containers\order\26' 118330,
              DEVICE 's:\Containers\order\27' 118330)
        EXTENTSIZE 256 PREFETCHSIZE 0
        BUFFERPOOL IBMDEFAULT8K;

drop tablespace order_10;
create regular tablespace order_10
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\order\28' 118330,
          DEVICE 's:\Containers\order\29' 118330,
          DEVICE 's:\Containers\order\30' 118330)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULT8K;

drop tablespace order_11;
create regular tablespace order_11
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\order\31' 118330,
          DEVICE 's:\Containers\order\32' 118330,
          DEVICE 's:\Containers\order\33' 118330)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULT8K;

drop tablespace order_12;
create regular tablespace order_12
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\order\34' 118330,
          DEVICE 's:\Containers\order\35' 118330,
          DEVICE 's:\Containers\order\36' 118330)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULT8K;

drop tablespace order_idx_01;
create regular tablespace order_idx_01
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\order_idx\01' 118330,
          DEVICE 's:\Containers\order_idx\02' 118330,
          DEVICE 's:\Containers\order_idx\03' 118330)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULT8K;

drop tablespace order_idx_02;
create regular tablespace order_idx_02
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\order_idx\04' 118330,
          DEVICE 's:\Containers\order_idx\05' 118330,
          DEVICE 's:\Containers\order_idx\06' 118330)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULT8K;

drop tablespace order_idx_03;
create regular tablespace order_idx_03
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\order_idx\07' 118330,
          DEVICE 's:\Containers\order_idx\08' 118330,
          DEVICE 's:\Containers\order_idx\09' 118330)

```

```

        EXTENTSIZE 256 PREFETCHSIZE 0
        BUFFERPOOL IBMDEFAULT8K;

drop tablespace order_idx_04;
create regular tablespace order_idx_04
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\order_idx\10' 118330,
          DEVICE 's:\Containers\order_idx\11' 118330,
          DEVICE 's:\Containers\order_idx\12' 118330)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULT8K;

drop tablespace order_idx_05;
create regular tablespace order_idx_05
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\order_idx\13' 118330,
          DEVICE 's:\Containers\order_idx\14' 118330,
          DEVICE 's:\Containers\order_idx\15' 118330)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULT8K;

drop tablespace order_idx_06;
create regular tablespace order_idx_06
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\order_idx\16' 118330,
          DEVICE 's:\Containers\order_idx\17' 118330,
          DEVICE 's:\Containers\order_idx\18' 118330)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULT8K;

drop tablespace order_idx_07;
create regular tablespace order_idx_07
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\order_idx\19' 118330,
          DEVICE 's:\Containers\order_idx\20' 118330,
          DEVICE 's:\Containers\order_idx\21' 118330)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULT8K;

drop tablespace order_idx_08;
create regular tablespace order_idx_08
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\order_idx\22' 118330,
          DEVICE 's:\Containers\order_idx\23' 118330,
          DEVICE 's:\Containers\order_idx\24' 118330)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULT8K;

drop tablespace order_idx_09;
create regular tablespace order_idx_09
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\order_idx\25' 118330,
          DEVICE 's:\Containers\order_idx\26' 118330,
          DEVICE 's:\Containers\order_idx\27' 118330)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULT8K;

drop tablespace order_idx_10;
create regular tablespace order_idx_10

```



```

        PAGESIZE 8K managed by database
        using (DEVICE 's:\Containers\order_idx\28' 118330,
              DEVICE 's:\Containers\order_idx\29' 118330,
              DEVICE 's:\Containers\order_idx\30' 118330)
        EXTENTSIZE 256 PREFETCHSIZE 0
        BUFFERPOOL IBMDEFAULT8K;

drop tablespace order_idx_11;
create regular tablespace order_idx_11
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\order_idx\31' 118330,
          DEVICE 's:\Containers\order_idx\32' 118330,
          DEVICE 's:\Containers\order_idx\33' 118330)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULT8K;

drop tablespace order_idx_12;
create regular tablespace order_idx_12
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\order_idx\34' 118330,
          DEVICE 's:\Containers\order_idx\35' 118330,
          DEVICE 's:\Containers\order_idx\36' 118330)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULT8K;

drop tablespace orderl_01;
create regular tablespace orderl_01
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\orderl\01' 3066320,
          DEVICE 's:\Containers\orderl\02' 3066320,
          DEVICE 's:\Containers\orderl\03' 3066320)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULT8K;

drop tablespace orderl_02;
create regular tablespace orderl_02
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\orderl\04' 3066320,
          DEVICE 's:\Containers\orderl\05' 3066320,
          DEVICE 's:\Containers\orderl\06' 3066320)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULT8K;

drop tablespace orderl_03;
create regular tablespace orderl_03
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\orderl\07' 3066320,
          DEVICE 's:\Containers\orderl\08' 3066320,
          DEVICE 's:\Containers\orderl\09' 3066320)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULT8K;

drop tablespace orderl_04;
create regular tablespace orderl_04
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\orderl\10' 3066320,
          DEVICE 's:\Containers\orderl\11' 3066320,

```

```

          DEVICE 's:\Containers\orderl\12' 3066320)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULT8K;

drop tablespace orderl_05;
create regular tablespace orderl_05
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\orderl\13' 3066320,
          DEVICE 's:\Containers\orderl\14' 3066320,
          DEVICE 's:\Containers\orderl\15' 3066320)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULT8K;

drop tablespace orderl_06;
create regular tablespace orderl_06
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\orderl\16' 3066320,
          DEVICE 's:\Containers\orderl\17' 3066320,
          DEVICE 's:\Containers\orderl\18' 3066320)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULT8K;

drop tablespace orderl_07;
create regular tablespace orderl_07
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\orderl\19' 3066320,
          DEVICE 's:\Containers\orderl\20' 3066320,
          DEVICE 's:\Containers\orderl\21' 3066320)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULT8K;

drop tablespace orderl_08;
create regular tablespace orderl_08
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\orderl\22' 3066320,
          DEVICE 's:\Containers\orderl\23' 3066320,
          DEVICE 's:\Containers\orderl\24' 3066320)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULT8K;

drop tablespace orderl_09;
create regular tablespace orderl_09
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\orderl\25' 3066320,
          DEVICE 's:\Containers\orderl\26' 3066320,
          DEVICE 's:\Containers\orderl\27' 3066320)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULT8K;

drop tablespace orderl_10;
create regular tablespace orderl_10
    PAGESIZE 8K managed by database
    using (DEVICE 's:\Containers\orderl\28' 3066320,
          DEVICE 's:\Containers\orderl\29' 3066320,
          DEVICE 's:\Containers\orderl\30' 3066320)
    EXTENTSIZE 256 PREFETCHSIZE 0
    BUFFERPOOL IBMDEFAULT8K;

drop tablespace orderl_11;

```

```

create regular tablespace orderl_11
  PAGESIZE 8K managed by database
  using (DEVICE 's:\Containers\orderl\31' 3066320,
        DEVICE 's:\Containers\orderl\32' 3066320,
        DEVICE 's:\Containers\orderl\33' 3066320)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULT8K;

drop tablespace orderl_12;
create regular tablespace orderl_12
  PAGESIZE 8K managed by database
  using (DEVICE 's:\Containers\orderl\34' 3066320,
        DEVICE 's:\Containers\orderl\35' 3066320,
        DEVICE 's:\Containers\orderl\36' 3066320)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULT8K;

drop tablespace history_01;
create regular tablespace history_01
  PAGESIZE 16K managed by database
  using (DEVICE 's:\Containers\history\01' 114350,
        DEVICE 's:\Containers\history\02' 114350,
        DEVICE 's:\Containers\history\03' 114350)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULT16K;

drop tablespace history_02;
create regular tablespace history_02
  PAGESIZE 16K managed by database
  using (DEVICE 's:\Containers\history\04' 114350,
        DEVICE 's:\Containers\history\05' 114350,
        DEVICE 's:\Containers\history\06' 114350)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULT16K;

drop tablespace history_03;
create regular tablespace history_03
  PAGESIZE 16K managed by database
  using (DEVICE 's:\Containers\history\07' 114350,
        DEVICE 's:\Containers\history\08' 114350,
        DEVICE 's:\Containers\history\09' 114350)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULT16K;

drop tablespace history_04;
create regular tablespace history_04
  PAGESIZE 16K managed by database
  using (DEVICE 's:\Containers\history\10' 114350,
        DEVICE 's:\Containers\history\11' 114350,
        DEVICE 's:\Containers\history\12' 114350)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULT16K;

drop tablespace history_05;
create regular tablespace history_05
  PAGESIZE 16K managed by database
  using (DEVICE 's:\Containers\history\13' 114350,

```

```

        DEVICE 's:\Containers\history\14' 114350,
        DEVICE 's:\Containers\history\15' 114350)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULT16K;

drop tablespace history_06;
create regular tablespace history_06
  PAGESIZE 16K managed by database
  using (DEVICE 's:\Containers\history\16' 114350,
        DEVICE 's:\Containers\history\17' 114350,
        DEVICE 's:\Containers\history\18' 114350)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULT16K;

drop tablespace history_07;
create regular tablespace history_07
  PAGESIZE 16K managed by database
  using (DEVICE 's:\Containers\history\19' 114350,
        DEVICE 's:\Containers\history\20' 114350,
        DEVICE 's:\Containers\history\21' 114350)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULT16K;

drop tablespace history_08;
create regular tablespace history_08
  PAGESIZE 16K managed by database
  using (DEVICE 's:\Containers\history\22' 114350,
        DEVICE 's:\Containers\history\23' 114350,
        DEVICE 's:\Containers\history\24' 114350)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULT16K;

drop tablespace history_09;
create regular tablespace history_09
  PAGESIZE 16K managed by database
  using (DEVICE 's:\Containers\history\25' 114350,
        DEVICE 's:\Containers\history\26' 114350,
        DEVICE 's:\Containers\history\27' 114350)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULT16K;

drop tablespace history_10;
create regular tablespace history_10
  PAGESIZE 16K managed by database
  using (DEVICE 's:\Containers\history\28' 114350,
        DEVICE 's:\Containers\history\29' 114350,
        DEVICE 's:\Containers\history\30' 114350)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULT16K;

drop tablespace history_11;
create regular tablespace history_11
  PAGESIZE 16K managed by database
  using (DEVICE 's:\Containers\history\31' 114350,
        DEVICE 's:\Containers\history\32' 114350,
        DEVICE 's:\Containers\history\33' 114350)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULT16K;

```

```

drop tablespace history_12;
create regular tablespace history_12
  PAGESIZE 16K managed by database
  using (DEVICE 's:\Containers\history\34' 114350,
        DEVICE 's:\Containers\history\35' 114350,
        DEVICE 's:\Containers\history\36' 114350)
  EXTENTSIZE 256 PREFETCHSIZE 0
  BUFFERPOOL IBMDEFAULT16K;

connect reset;
terminate;

```

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
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
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
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_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
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
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
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
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
WITH ROW MOVEMENT;
COMMIT WORK;
connect reset;

```

make_flat1.cmd

```

gendata.exe -t 3 -r 1 19800 -a 12 -P1 S:\flats\flat -f warehouse >
ware.out
gendata.exe -t 6 -r 1 19800 -a 12 -P1 S:\flats\flat -f stock > stock.out

```

make_flat2.cmd

```

gendata.exe -t 4 -r 1 19800 -a 12 -P1 S:\flats\flat -f district >
dist.out
gendata.exe -t 7 -r 1 19800 -a 12 -P1 S:\flats\flat -f customer >
cust.out

```

make_flat3.cmd

```

gendata.exe -t 5 -r 1 19800 -a 12 -P1 S:\flats\flat -f item > item.out

```

```

gendata.exe -t 9 -r 1 19800 -a 12 -P1 S:\flats\flat -P2 S:\flats\flat -
f1 orders -f2 orderline > orders.out

```

make_flat4.cmd

```

gendata.exe -t 8 -r 1 19800 -a 12 -P1 S:\flats\flat -f history >
hist.out
gendata.exe -t 11 -r 1 19800 -a 12 -P1 S:\flats\flat -f neworder >
nordr.out

```

LOAD_CUSTOMER.ddl

```

CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE CUSTOMER1 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM S:\flats\flat_01\customer_01_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 49500000 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 S:\flats\flat_02\customer_02_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 49500000 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 S:\flats\flat_03\customer_03_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 49500000 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 S:\flats\flat_04\customer_04_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 49500000 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 S:\flats\flat_05\customer_05_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 49500000 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 S:\flats\flat_06\customer_06_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 49500000 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 S:\flats\flat_07\customer_07_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 49500000 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 S:\flats\flat_08\customer_08_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 49500000 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 S:\flats\flat_09\customer_09_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 49500000 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 S:\flats\flat_10\customer_10_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 49500000 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 S:\flats\flat_11\customer_11_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 49500000 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 S:\flats\flat_12\customer_12_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 49500000 INSERT INTO
CUSTOMER12;
COMMIT WORK;
CONNECT RESET;

```

LOAD_DISTRICT.ddl

```

CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM S:\flats\flat_01\district_01_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 S:\flats\flat_02\district_02_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 S:\flats\flat_03\district_03_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 S:\flats\flat_04\district_04_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 S:\flats\flat_05\district_05_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 S:\flats\flat_06\district_06_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 S:\flats\flat_07\district_07_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 S:\flats\flat_08\district_08_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 S:\flats\flat_09\district_09_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 S:\flats\flat_10\district_10_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 S:\flats\flat_11\district_11_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 S:\flats\flat_12\district_12_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO DISTRICT12;
COMMIT WORK;
CONNECT RESET;

```

LOAD_HISTORY.ddl

```

connect to TPCC in share mode;
LOAD FROM S:\flats\flat_01\history_01_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS FASTPARSE REPLACE INTO HISTORY1 NONRECOVERABLE DATA BUFFER
100000 CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM S:\flats\flat_02\history_02_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS FASTPARSE REPLACE INTO HISTORY2 NONRECOVERABLE DATA BUFFER
100000 CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM S:\flats\flat_03\history_03_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS FASTPARSE REPLACE INTO HISTORY3 NONRECOVERABLE DATA BUFFER
100000 CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM S:\flats\flat_04\history_04_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS FASTPARSE REPLACE INTO HISTORY4 NONRECOVERABLE DATA BUFFER
100000 CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM S:\flats\flat_05\history_05_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS FASTPARSE REPLACE INTO HISTORY5 NONRECOVERABLE DATA BUFFER
100000 CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM S:\flats\flat_06\history_06_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS FASTPARSE REPLACE INTO HISTORY6 NONRECOVERABLE DATA BUFFER
100000 CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM S:\flats\flat_07\history_07_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS FASTPARSE REPLACE INTO HISTORY7 NONRECOVERABLE DATA BUFFER
100000 CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM S:\flats\flat_08\history_08_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS FASTPARSE REPLACE INTO HISTORY8 NONRECOVERABLE DATA BUFFER
100000 CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM S:\flats\flat_09\history_09_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS FASTPARSE REPLACE INTO HISTORY9 NONRECOVERABLE DATA BUFFER
100000 CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM S:\flats\flat_10\history_10_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS FASTPARSE REPLACE INTO HISTORY10 NONRECOVERABLE DATA BUFFER
100000 CPU_PARALLELISM 4 ;
connect reset;

```

```

connect to TPCC in share mode;
LOAD FROM S:\flats\flat_11\history_11_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS FASTPARSE REPLACE INTO HISTORY11 NONRECOVERABLE DATA BUFFER
100000 CPU_PARALLELISM 4 ;
connect reset;
connect to TPCC in share mode;
LOAD FROM S:\flats\flat_12\history_12_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS FASTPARSE REPLACE INTO HISTORY12 NONRECOVERABLE DATA BUFFER
100000 CPU_PARALLELISM 4 ;
connect reset;

```

LOAD_ITEM_1.ddl

```

CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM S:\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.ddl

```

CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM S:\flats\flat_01\neworder_01_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDERA1;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM S:\flats\flat_02\neworder_02_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDERA2;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM S:\flats\flat_03\neworder_03_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDERA3;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM S:\flats\flat_04\neworder_04_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDERA4;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM S:\flats\flat_05\neworder_05_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDERA5;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM S:\flats\flat_06\neworder_06_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDERA6;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM S:\flats\flat_07\neworder_07_1.dat OF DEL MODIFIED BY

```

```

COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDERA7;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM S:\flats\flat_08\neworder_08_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDERA8;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM S:\flats\flat_09\neworder_09_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDERA9;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM S:\flats\flat_10\neworder_10_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDERA10;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM S:\flats\flat_11\neworder_11_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDERA11;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM S:\flats\flat_12\neworder_12_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO NEW_ORDERA12;
COMMIT WORK;
CONNECT RESET;

```

LOAD_ORDER_LINE.ddl

```

CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE ORDER_LINE1 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM S:\flats\flat_01\orderline_01_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 544500000 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 S:\flats\flat_02\orderline_02_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 544500000 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 S:\flats\flat_03\orderline_03_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 544500000 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 S:\flats\flat_04\orderline_04_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 544500000 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 S:\flats\flat_05\orderline_05_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 544500000 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 S:\flats\flat_06\orderline_06_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 544500000 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 S:\flats\flat_07\orderline_07_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 544500000 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 S:\flats\flat_08\orderline_08_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 544500000 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 S:\flats\flat_09\orderline_09_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 544500000 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 S:\flats\flat_10\orderline_10_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 544500000 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 S:\flats\flat_11\orderline_11_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS COMPOUND=50 COMMITCOUNT 544500000 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 S:\flats\flat_12\orderline_12_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 COMMITCOUNT 544500000 INSERT INTO
ORDER_LINE12;
COMMIT WORK;
CONNECT RESET;

```

LOAD_ORDERS.ddl

```

CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM S:\flats\flat_01\orders_01_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000 INSERT INTO
ORDERS1;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM S:\flats\flat_02\orders_02_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000 INSERT INTO
ORDERS2;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM S:\flats\flat_03\orders_03_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000 INSERT INTO
ORDERS3;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM S:\flats\flat_04\orders_04_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000 INSERT INTO
ORDERS4;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM S:\flats\flat_05\orders_05_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000 INSERT INTO
ORDERS5;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM S:\flats\flat_06\orders_06_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000 INSERT INTO
ORDERS6;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM S:\flats\flat_07\orders_07_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000 INSERT INTO
ORDERS7;
COMMIT WORK;

```

```

CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM S:\flats\flat_08\orders_08_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000 INSERT INTO
ORDERS8;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM S:\flats\flat_09\orders_09_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000 INSERT INTO
ORDERS9;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM S:\flats\flat_10\orders_10_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000 INSERT INTO
ORDERS10;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM S:\flats\flat_11\orders_11_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000 INSERT INTO
ORDERS11;
COMMIT WORK;
CONNECT RESET;
CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM S:\flats\flat_12\orders_12_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000 INSERT INTO
ORDERS12;
COMMIT WORK;
CONNECT RESET;

```

LOAD_STOCK.ddl

```

CONNECT TO TPCC IN SHARE MODE;
UPDATE COMMAND OPTIONS USING C OFF;
ALTER TABLE STOCK1 ACTIVATE NOT LOGGED INITIALLY;
IMPORT FROM S:\flats\flat_01\stock_01_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS COMPOUND=50 COMMITCOUNT 165000000 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 S:\flats\flat_02\stock_02_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS COMPOUND=50 COMMITCOUNT 165000000 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 S:\flats\flat_03\stock_03_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS COMPOUND=50 COMMITCOUNT 165000000 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 S:\flats\flat_04\stock_04_1.dat OF DEL MODIFIED BY COLDEL|

```

```

KEEPBLANKS COMPOUND=50  COMMITCOUNT 165000000 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 S:\flats\flat_05\stock_05_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS COMPOUND=50  COMMITCOUNT 165000000 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 S:\flats\flat_06\stock_06_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS COMPOUND=50  COMMITCOUNT 165000000 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 S:\flats\flat_07\stock_07_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS COMPOUND=50  COMMITCOUNT 165000000 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 S:\flats\flat_08\stock_08_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS COMPOUND=50  COMMITCOUNT 165000000 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 S:\flats\flat_09\stock_09_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS COMPOUND=50  COMMITCOUNT 165000000 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 S:\flats\flat_10\stock_10_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS COMPOUND=50  COMMITCOUNT 165000000 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 S:\flats\flat_11\stock_11_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS COMPOUND=50  COMMITCOUNT 165000000 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 S:\flats\flat_12\stock_12_1.dat OF DEL MODIFIED BY COLDEL|
KEEPBLANKS COMPOUND=50  COMMITCOUNT 165000000 INSERT INTO STOCK12;
COMMIT WORK;
CONNECT RESET;

```

LOAD_WAREHOUSE.ddl

```

CONNECT TO TPCC IN SHARE MODE;
IMPORT FROM S:\flats\flat_01\warehouse_01_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 S:\flats\flat_02\warehouse_02_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 S:\flats\flat_03\warehouse_03_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 S:\flats\flat_04\warehouse_04_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 S:\flats\flat_05\warehouse_05_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 S:\flats\flat_06\warehouse_06_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 S:\flats\flat_07\warehouse_07_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 S:\flats\flat_08\warehouse_08_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 S:\flats\flat_09\warehouse_09_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 S:\flats\flat_10\warehouse_10_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 S:\flats\flat_11\warehouse_11_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 S:\flats\flat_12\warehouse_12_1.dat OF DEL MODIFIED BY
COLDEL| KEEPBLANKS COMPOUND=50 ALLOW WRITE ACCESS COMMITCOUNT 1000
INSERT INTO WAREHOUSE12;
COMMIT WORK;
CONNECT RESET;

```

RNST_CUSTOMER.ddl

```

connect to TPCC in share mode;
RUNSTATS ON TABLE nec.CUSTOMER1 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.CUSTOMER2 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.CUSTOMER3 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.CUSTOMER4 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.CUSTOMER5 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.CUSTOMER6 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.CUSTOMER7 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.CUSTOMER8 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.CUSTOMER9 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.CUSTOMER10 AND INDEXES ALL;
COMMIT WORK;
connect reset;

```

```

connect to TPCC in share mode;
RUNSTATS ON TABLE nec.CUSTOMER11 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.CUSTOMER12 AND INDEXES ALL;
COMMIT WORK;
connect reset;

```

RNST_DISTRICT.ddl

```

connect to TPCC in share mode;
RUNSTATS ON TABLE nec.DISTRICT1 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.DISTRICT2 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.DISTRICT3 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.DISTRICT4 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.DISTRICT5 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.DISTRICT6 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.DISTRICT7 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.DISTRICT8 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.DISTRICT9 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.DISTRICT10 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.DISTRICT11 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.DISTRICT12 AND INDEXES ALL;
COMMIT WORK;
connect reset;

```

RNST_HISTORY.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.HISTORY1 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.HISTORY2 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.HISTORY3 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.HISTORY4 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.HISTORY5 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.HISTORY6 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.HISTORY7 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.HISTORY8 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.HISTORY9 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.HISTORY10 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.HISTORY11 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.HISTORY12 AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_ITEM.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.ITEM AND INDEXES ALL;
COMMIT WORK;
connect reset;
```

RNST_NEW_ORDER.ddl

```
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.NEW_ORDERA1 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.NEW_ORDERA2 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.NEW_ORDERA3 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.NEW_ORDERA4 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.NEW_ORDERA5 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.NEW_ORDERA6 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.NEW_ORDERA7 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.NEW_ORDERA8 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.NEW_ORDERA9 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.NEW_ORDERA10 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.NEW_ORDERA11 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.NEW_ORDERA12 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.NEW_ORDERB1 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.NEW_ORDERB2 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
```

```

RUNSTATS ON TABLE nec.NEW_ORDERB3 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.NEW_ORDERB4 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.NEW_ORDERB5 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.NEW_ORDERB6 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.NEW_ORDERB7 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.NEW_ORDERB8 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.NEW_ORDERB9 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.NEW_ORDERB10 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.NEW_ORDERB11 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.NEW_ORDERB12 AND INDEXES ALL;
COMMIT WORK;
connect reset;

```

RNST_ORDER_LINE.ddl

```

connect to TPCC in share mode;
RUNSTATS ON TABLE nec.ORDER_LINE1 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.ORDER_LINE2 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.ORDER_LINE3 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.ORDER_LINE4 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;

```

```

RUNSTATS ON TABLE nec.ORDER_LINE5 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.ORDER_LINE6 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.ORDER_LINE7 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.ORDER_LINE8 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.ORDER_LINE9 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.ORDER_LINE10 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.ORDER_LINE11 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.ORDER_LINE12 AND INDEXES ALL;
COMMIT WORK;
connect reset;

```

RNST_ORDERS.ddl

```

connect to TPCC in share mode;
RUNSTATS ON TABLE nec.ORDERS1 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.ORDERS2 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.ORDERS3 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.ORDERS4 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.ORDERS5 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.ORDERS6 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;

```

```

RUNSTATS ON TABLE nec.ORDERS7 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.ORDERS8 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.ORDERS9 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.ORDERS10 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.ORDERS11 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.ORDERS12 AND INDEXES ALL;
COMMIT WORK;
connect reset;

```

RNST_STOCK.ddl

```

connect to TPCC in share mode;
RUNSTATS ON TABLE nec.STOCK1 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.STOCK2 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.STOCK3 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.STOCK4 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.STOCK5 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.STOCK6 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.STOCK7 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.STOCK8 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;

```

```

RUNSTATS ON TABLE nec.STOCK9 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.STOCK10 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.STOCK11 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.STOCK12 AND INDEXES ALL;
COMMIT WORK;
connect reset;

```

RNST_WAREHOUSE.ddl

```

connect to TPCC in share mode;
RUNSTATS ON TABLE nec.WAREHOUSE1 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.WAREHOUSE2 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.WAREHOUSE3 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.WAREHOUSE4 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.WAREHOUSE5 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.WAREHOUSE6 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.WAREHOUSE7 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.WAREHOUSE8 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.WAREHOUSE9 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.WAREHOUSE10 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;

```

```

RUNSTATS ON TABLE nec.WAREHOUSE11 AND INDEXES ALL;
COMMIT WORK;
connect reset;
connect to TPCC in share mode;
RUNSTATS ON TABLE nec.WAREHOUSE12 AND INDEXES ALL;
COMMIT WORK;
connect reset;

```

B2. Loader Source 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 -DWIN64 -J -Zp8 -DREG_KIT_METHOD -G2
CFLAGS_OUT=/Fo
CFLAGS_DEBUG=

# Linker Configuration (MSVC)
LD_EXEC=link.exe
LD_STORP=link.exe
LDFLAGS_EXEC=
LDFLAGS_SHLIB=/DLL
LDFLAGS_STORP=$(LDFLAGS_SHLIB) /DEF:rpctpcc.def
LDFLAGS_LIB=/LIBPATH:$(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=CK040607

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

@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

```

```
@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
```

```

/*****
*****
** 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.
*****/

```

```
/* 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 );
```

```

/*-----
*/
/*      main
*/
/*-----
*/
int main (int argc, char *argv[])
{
    int option = -1;
    char *delim = NULL;

    /* *****
*/
/* Compute Warehouse Ranges
*/
/* *****
*/

```



```

ware_start = 1;
ware_end = WAREHOUSES;

/* *****
/* Compute Warehouse arms Ranges
/* *****
*/
num_part = 1;
part_range = WAREHOUSES;

/* *****
/* Process Command Line Arguments
/* *****
*/

/* Valid Command Line Options
* -----
* Table Option:          -t <table>          (-t 3 for warehouse)
* Output Column Delimiter: -d <char>          (-d ' ', -d '|', etc)
* Output to File:        -f[n] <file>         (-f customer.dat)
* Output to Pipe:        -p[n] <pipename>      (-p tpccpipe.000)
* Warehouse Range:      -r <start> <end>      (-r 1 100)
* Scaling Report:       -s
* Quiet Mode:          -q
*   * Num of ARMS          -a
*
* The -f[n] and/or -p[n] options are required.
* The -t, -d, -r, -s and -q options are optional.
*
* If -d is omitted, the vertical bar (pipe) symbol '|' will be
used.
* If -r is omitted, the range [1..WAREHOUSES] will be used.
*
* Due to the TPC-C spec requiring that orders and orderline be
* generated at the same time, there is an extension to the -f and -p
* options to specify one of the two output streams for each argument.
*
* -f1 orders.dat -f2 orderline.dat will output to two files
* -f1 orders.dat -p2 tpccpipe.000 will output to a file and a pipe
*
* -f1/-p1 specifies the destination for the orders table
* -f2/-p2 specifies the destination for the orderline table
*
*/

/* Read Arguments */
for (i=1; i<argc; i++)
{
    if (strcmp(argv[i], "-t") == 0) {
        option = atoi(argv[i+1]);
        i++;
    } else if (strcmp(argv[i], "-r") == 0) {
        ware_start = atoi(argv[i+1]);
        ware_end = atoi(argv[i+2]);
        i += 2;

```

```

    } else if (strcmp(argv[i], "-d") == 0) {
        delim = argv[i+1];
        i++;
    } else if ((strcmp(argv[i], "-f") == 0) ||
                (strcmp(argv[i], "-f1") == 0)) {
        outtype1 = IOH_FILE;
        outname1 = argv[i+1];
        i++;
    } else if (strcmp(argv[i], "-f2") == 0) {
        outtype2 = IOH_FILE;
        outname2 = argv[i+1];
        i++;
    } else if ((strcmp(argv[i], "-p") == 0) ||
                (strcmp(argv[i], "-p1") == 0)) {
        outtype1 = IOH_PIPE;
        outname1 = argv[i+1];
        i++;
    } else if (strcmp(argv[i], "-p2") == 0) {
        outtype2 = IOH_PIPE;
        outname2 = argv[i+1];
        i++;
    } else if (strcmp(argv[i], "-a") == 0) {
        num_part = atoi(argv[i+1]);
        i++;
    } else if (strcmp(argv[i], "-P1") == 0) {
        pre_pass = argv[i+1];
        i++;
    } else if (strcmp(argv[i], "-P2") == 0) {
        pre_pass2 = argv[i+1];
        i++;
    } else if (strcmp(argv[i], "-s") == 0) {
        ScalingReport();
        exit(0);
    } else if (strcmp(argv[i], "-q") == 0) {
        quiet_mode = 1;
    } else {
        fprintf(stderr, "gendata: Don't understand
argument: %s\n", argv[i]);
        exit(-1);
    }
}

/* *****
/* Validate Command Line Arguments
/* *****
*/

/* Validate Table Argument */
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

```

```

neccessary
    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 || strcmp(pre_pass2, "") == 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 neccessary.
*/
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);
}

/* Validate Range of partitions */
if (((ware_end - ware_start + 1) % num_part) != 0 )
{
    fprintf(stderr,"gendata: Invalid part number selected. warehouse
number must be dividable by num of part\n");
    exit(-1);
}

if ( pre_pass == NULL )
{
    fprintf(stderr,"gendata: Please input outfile pass\n");
    exit(-1);
}

part_range = (ware_end - ware_start + 1) / num_part;

/* *****
*/
/* 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];

    initialize_random(13,42);
    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();

```

```

    elapse = timestamp2 - timestamp1;
    if (rc == 0) {
        fprintf(stdout, "\nITEM table generated in %.2f
seconds.\n\n", elapse);
        fflush(stdout);
    } else {
        fprintf(stderr, "\nITEM table FAILED at (I %d) after %.2f
seconds.\n\n", item_num, elapse);
        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] ;

    char temp_char[25];
    int next_start;
    int this_end;

    IOH_NUM numBytes;
    ioHandle hnd;
    char Buffer[1024];

    initialize_random(7,11);
    timestamp1 = current_time();

    next_start = ware_start;

    for (startpoint = 1; startpoint <= num_part; startpoint++)
    {
        if(startpoint < 10)
            sprintf(temp_char, "_0%d\\", startpoint);
        else
            sprintf(temp_char, "%d\\", startpoint);

        strcpy(full_name, pre_pass);

```

```

    strcat(full_name, temp_char);
    strcat(full_name, outname1);

    if(startpoint < 10)
        sprintf(post_name, "_%d_1.dat", startpoint);
    else
        sprintf(post_name, "_%d_1.dat", startpoint);

    strcat(full_name, post_name);

    rc = GenericOpen(&hnd, outtype1, full_name);
    if (rc != 0) { goto stock_done; }

//   rc = GenericOpen(&hnd, outtype1, outname1);

    this_end = next_start + part_range - 1;

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 = next_start; ware_num <= this_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... */

    next_start = this_end + 1;

    rc = GenericClose(&hnd);

}

stock_done:

    timestamp2 = current_time();
    elapse = timestamp2 - timestamp1;
    if (rc == 0) {
        fprintf(stdout, "\nSTOCK table generated in %8.2f
seconds.\n\n", elapse);
        fflush(stdout);
    } else {
        fprintf(stderr, "\nSTOCK table FAILED at (S %d W %d) after %8.2f
seconds.\n\n", stock_num, ware_num, elapse);
        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];

    char temp_char[25];
    int next_start;

```

```

int    this_end;

next_start = ware_start;

initialize_random(23,111);
timestamp1 = current_time();

startpoint++ for (startpoint = 1; startpoint <= num_part;
{

    if(startpoint < 10)
        sprintf(temp_char, "_0%d\\", startpoint);
    else
        sprintf(temp_char, "_%d\\", startpoint);

    strcpy(full_name, pre_pass);
    strcat(full_name, temp_char);
    strcat(full_name, outname1);

    if(startpoint < 10)
        sprintf(post_name, "_0%d.1.dat", startpoint);
    else
        sprintf(post_name, "_%d.1.dat", startpoint);

    strcat(full_name, post_name);

    rc = GenericOpen(&hnd, outtype1, full_name);
    if (rc != 0) { goto ware_done; }

    this_end = next_start + part_range - 1;

//    rc = GenericOpen(&hnd, outtype1, outname1);

    for (ware_num = next_start; ware_num <= this_end; ware_num++)
    {
        if (!quiet_mode && ((ware_num % 500) == 0))
            // @dxxxxxmte
            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 = 300000000;

    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 */

    next_start = this_end + 1;

    rc = GenericClose(&hnd);

}

ware_done:

    timestamp2 = current_time();
    elapse = timestamp2 - timestamp1;
    if (rc == 0) {
        fprintf(stdout, "\\nWAREHOUSE table generated in %8.2f
seconds.\\n\\n", elapse);
        fflush(stdout);
    } else {
        fprintf(stderr, "\\nWAREHOUSE table FAILED at (W %d) after %8.2f
seconds.\\n\\n", ware_num, elapse);
        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];

char temp_char[25];
int next_start;
int this_end;

next_start = ware_start;

next_o_id = CUSTOMERS_PER_DISTRICT + 1;
initialize_random(44,73);
timestamp1 = current_time();

for (startpoint = 1; startpoint <= num_part; startpoint++)
{
    if(startpoint < 10)
        sprintf(temp_char,"_0%d\\",startpoint);
    else
        sprintf(temp_char,"_%d\\",startpoint);

    strcpy(full_name, pre_pass);
    strcat(full_name, temp_char);
    strcat(full_name, outname1);

    if(startpoint < 10)
        sprintf(post_name,"_0%d_1.dat",startpoint);
    else
        sprintf(post_name,"_%d_1.dat",startpoint);

    strcat(full_name, post_name);

    rc = GenericOpen(&hnd, outtype1, full_name);
    if (rc != 0) { goto dist_done; }

    this_end = next_start + part_range - 1;

// rc = GenericOpen(&hnd, outtype1, outname1);
// if (rc != 0) { goto dist_done; }

// for (ware_num = ware_start; ware_num <= ware_end; ware_num++)
// for (ware_num = next_start; ware_num <= this_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... */
    next_start = this_end + 1;

    rc = GenericClose(&hnd);

}

dist_done:

    timestamp2 = current_time();
    elapse = timestamp2 - timestamp1;
    if (rc == 0) {
        fprintf(stdout, "\\n\\nDISTRICT table generated in %8.2f
seconds.\\n\\n",elapse);
        fflush(stdout);
    } else {
        fprintf(stderr, "\\n\\nDISTRICT table FAILED at (W %d D %d) after %8.2f
seconds.\\n\\n",ware_num,dist_num,elapse);

```

```

        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;

    char temp_char[25];
    int next_start;
    int this_end;

    next_start = ware_start;

    initialize_random(10,64);
    timestamp1 = current_time();

    for (startpoint = 1; startpoint <= num_part; startpoint++)
    {
        if(startpoint < 10)
            sprintf(temp_char, "_%d\\", startpoint);
        else
            sprintf(temp_char, "%d\\", startpoint);

        strcpy(full_name, pre_pass);

```

```

        strcat(full_name, temp_char);
        strcat(full_name, outname1);

        if(startpoint < 10)
            sprintf(post_name, "_%d_1.dat", startpoint);
        else
            sprintf(post_name, "%d_1.dat", startpoint);

        strcat(full_name, post_name);

        rc = GenericOpen(&hnd, outtype1, full_name);

        this_end = next_start + part_range - 1;

// 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 (ware_num = next_start; ware_num <= this_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-rctload) */
if (using_rctload == 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;

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... */
next_start = this_end + 1;

rc = GenericClose(&hnd);
}

cust_done:

timestamp2 = current_time();
elapsed = timestamp2 - timestamp1;

```

```

    if (rc == 0) {
        fprintf(stdout, "\nCUSTOMER table generated in %8.2f
seconds.\n\n", elapsed);
        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, elapsed);
        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 currtmstp;

    IOH_NUM numBytes;
    ioHandle hnd;
    char Buffer[1024];

    char temp_char[25];
    int next_start;
    int this_end;

    next_start = ware_start;

    initialize_random(15,63);
    timestamp1 = current_time();

    for (startpoint = 1; startpoint <= num_part; startpoint++)
    {

        if(startpoint < 10)
            sprintf(temp_char, "_0%d\\", startpoint);
        else
            sprintf(temp_char, "%d\\", startpoint);

        strcpy(full_name, pre_pass);
        strcat(full_name, temp_char);
        strcat(full_name, outname1);

        if(startpoint < 10)
            sprintf(post_name, "_0%d_1.dat", startpoint);
        else
            sprintf(post_name, "%d_1.dat", startpoint);
    }
}

```



```

        strcat(full_name, post_name);

        rc = GenericOpen(&hnd, outtype1, full_name);
        if (rc != 0) { goto hist_done; }

        this_end = next_start + part_range - 1;

//   rc = GenericOpen(&hnd, outtype1, outname1);
        currtmstp = time(NULL);

        for (ware_num = next_start; ware_num <= this_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,
                                        currtmstp,
                                        1000,
                                        hist_data);

                    rc = GenericWrite(&hnd, Buffer, numBytes);
                    if (rc != 0) { goto hist_done; }

                } /* end for customer... */
            } /* end for district... */
        } /* end for warehouse... */

        next_start = this_end + 1;

        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];

    char temp_char[25];
    int next_start;
    int this_end;

    next_start = ware_start;

    /* 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");
    }
    initialize_random(99,37);
    timestamp1 = current_time();

    for (startpoint = 1; startpoint <= num_part; startpoint++)
    {

```

```

    if(startpoint < 10)
        sprintf(temp_char, "_%0d\\", startpoint);
    else
        sprintf(temp_char, "%d\\", startpoint);

    strcpy(full_name, pre_pass);
    strcat(full_name, temp_char);
    strcat(full_name, outname1);

    if(startpoint < 10)
        sprintf(post_name, "_%0d_1.dat", startpoint);
    else
        sprintf(post_name, "%d_1.dat", startpoint);

    strcat(full_name, post_name);

    rc = GenericOpen(&hnd, outtype1, full_name);
    if (rc != 0) { goto neword_done; }

    this_end = next_start + part_range - 1;

// rc = GenericOpen(&hnd, outtype1, outname1);

/* We generate in O/W/D order for non-RCT tables. With the
 * data clustered on O_ID, this gives us good bufferpool
 * characteristics. We also create a btree index in W/D/O
 * order, to satisfy MIN(O_ID) queries.
 *
 * For RCT tables *with* RCT Jump Cache, we *should* generate
 * the data in W/D/O order (to match the table definition.)
 * We don't since it would push schema decisions into flat file
 * generation (and I don't want to do that.) It's just as easy
 * to sort the flat files afterwards.
 */

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 = next_start; ware_num <= this_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... */

    next_start = this_end + 1;

    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 ordr_carrier_id;
    sqlint32 ordr_ol_cnt;
    sqlint32 online_ol_num;
    sqlint32 online_item_num;

    sqlint32 online_amount;
    char online_dist_info[25];
    sqlint64 nulltmstp = 0;
    sqlint64 currtmstp;

    IOH_NUM numBytes;
    ioHandle hnd1, hnd2;
    char Buffer[1024];

    char temp_char[25];
    int next_start;
    int this_end;

    next_start = ware_start;

```

```

oline_dist_info[24] = '\0';
initialize_random(42,13);
timestamp1 = current_time();

for (startpoint = 1; startpoint <= num_part; startpoint++)
{

    if(startpoint < 10)
        sprintf(temp_char, "_0%d\\", startpoint);
    else
        sprintf(temp_char, "%d\\", startpoint);

    strcpy(full_name, pre_pass);
    strcpy(full_name2, pre_pass2);

    strcat(full_name, temp_char);
    strcat(full_name2, temp_char);

    strcat(full_name, outname1);
    strcat(full_name2, outname2);

    if(startpoint < 10)
        sprintf(post_name, "_0%d_1.dat", startpoint);
    else
        sprintf(post_name, "%d_1.dat", startpoint);

    strcat(full_name, post_name);
    strcat(full_name2, post_name);

    rc1 = GenericOpen(&hnd1, outtype1, full_name);
    if (rc1 != 0) { goto ool_done; }

    rc2 = GenericOpen(&hnd2, outtype2, full_name2);
    if (rc2 != 0) { goto ool_done; }

    this_end = next_start + part_range - 1;

    currtmstp = time(NULL);

    for (ware_num = next_start; ware_num <= this_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, fmt0rdr,
                                cust_num,
                                currtmstp,
                                ord_carrier_id,
                                ord_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_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) ? currtmstp :
nulltmstp),
                                    ((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 */

next_start = this_end + 1;

```

```

        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(fmtOrder, '|')) { *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");
}

```

```

}

```

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)

LDFLAGS =        $(LDFLAGS_EXEC) $(LDFLAGS_LIB)

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

OBJS =           tpccrnd$(OBJEXT) \
$(TPCC_ROOT)/Src.Common/tpccmisc$(OBJEXT)
OBJ_EEE =        $(TPCC_ROOT)/Src.Common/tpccclwh$(OBJEXT)

EXEC =           gendata$(BINEXT)

#
#####
# End-User Targets
#
#####

all:      $(EXEC)

```

```

clean:
- $(ERASE) *$(OBJEXT) $(EXEC)

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

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

$(EXEC):
$(LD_EXEC) $(LDFLAGS) $(OBS) *$(OBJEXT) $(LDFLAGS_OUT)$@

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

# Link Dependencies
gendata$(BINEXT):      $(OBS) gendata$(OBJEXT)

# Build Dependencies
# Source
gendata$(OBJEXT):      gendata.c

# Headers
gendata.c:      $(TPCC_ROOT)/include/tpccrnd.h
$(TPCC_ROOT)/include/lval.h

```

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 "db2tpcc.h"
#include "lval.h"

```

```
static char tbl_cust[CUSTOMERS_PER_DISTRICT];
```

```

static char alnum[] =
"0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ";

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(int x, int y)
{
    srand(x);
    srand(y);
}

```

```

/*
*****
***
*   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);
}

```

```

/*
*****
***
*   NUrund_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 = NURand_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));
}

```

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      /* May be ignored by RTE */
#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);

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

```

lval.h

```

#ifdef __LVAL_H
#define __LVAL_H
#define WAREHOUSES 19800
#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

```

platform.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.
*****/

/*
 * platform.h - Platform Isolation Layer
 */

#ifndef __PLATFORM_H
#define __PLATFORM_H

/*
 *****/
/* Generic Macros
 */
/*
 *****/
#define GEN_ERRCODE    GetLastError()

/*
 *****/
/* Windows I/O Macros
 */
/*
 *****/
#ifndef INVALID_HANDLE_VALUE
#define INVALID_HANDLE_VALUE ((DWORD)-1)
#endif

#ifndef INVALID_SET_FILE_POINTER
#define INVALID_SET_FILE_POINTER ((DWORD)-1)
#endif

#define IOH_INIT(hnd, type, name)
\
    hnd->fd = INVALID_HANDLE_VALUE;
\
    hnd->type = type;
\
    hnd->name = name;

#define IOH_CREATE(hnd)
\
    if (hnd->type == IOH_PIPE)

```

```

{
    DWORD timeout = 1000;
\
    hnd->fd = CreateNamedPipe(hnd->name, PIPE_ACCESS_OUTBOUND,
\
        PIPE_TYPE_BYTE | PIPE_READMODE_BYTE | PIPE_WAIT,
\
        1, 0, 0, timeout, NULL);
\
    rc = (hnd->fd == INVALID_HANDLE_VALUE) ? -1 : 0;
\
} else
{
    rc = 0;
\
}

#define IOH_OPEN(hnd)
\
    if (hnd->type == IOH_PIPE)
    {
        rc = (ConnectNamedPipe(hnd->fd, NULL) != 0) ? 0 : -1;
\
    } else
    {
        hnd->fd = CreateFile(hnd->name, GENERIC_WRITE, FILE_SHARE_WRITE,
\
            NULL, OPEN_ALWAYS, FILE_ATTRIBUTE_NORMAL, NULL);
\
        rc = (hnd->fd == INVALID_HANDLE_VALUE) ? -1 : 0;
\
        if (rc == 0 && hnd->type == IOH_FILE_APPEND)
        {
            rc = SetFilePointer(hnd->fd, 0, 0, FILE_END);
\
            if (rc == INVALID_SET_FILE_POINTER)
            {
                rc = (GetLastError() == NO_ERROR) ? 0 : -1;
\
            } else
            {
                rc = 0;
\
            }
\
        }
\
    }

#define IOH_WRITE(hnd, buff, num, num2)
\
    rc = (WriteFile(hnd->fd, buff, num, &num2, NULL) != 0) ? 0 : -1;

#define IOH_FLUSH(hnd)
\
    if (hnd->type == IOH_PIPE)
    {
        rc = (FlushFileBuffers(hnd->fd) != 0) ? 0 : -1;
\
    }

```

```

    } else
    {
        rc = 0;
    }
}

#define IOH_DELETE(hnd)          rc = 0;

#define IOH_CLOSE(hnd)
\
    if (hnd->type == IOH_PIPE)
    {
        rc = (DisconnectNamedPipe(hnd->fd) != 0) ? 0 : -1;
        IOH_ERRMSG(hnd, "disconnecting");
        rc = (CloseHandle(hnd->fd) != 0) ? 0 : -1;
    }
\

typedef DWORD IOH_NUM;
typedef HANDLE IOH_HND;

/****
**/
/* Windows Semaphore Macros
**/
/****
**/
#define SEM_HANDLE HANDLE

#define SEM_INIT(hnd, x, name)
\
    hnd = CreateSemaphore(NULL, x, 1, NULL);
\
    if (hnd == NULL)
\
        API_ERROR(__LINE__, "CreateSemaphore", (rc=GEN_ERRCODE));

#define SEM_WAIT(hnd)
\
    if ((rc=WaitForSingleObject(hnd, INFINITE)) == WAIT_FAILED)
\
        API_ERROR(__LINE__, "WaitForSingleObject", (rc=GEN_ERRCODE));

#define SEM_FREE(hnd)
\
    ReleaseSemaphore(hnd, 1, NULL)

#define SEM_DESTROY(hnd)
\
    if ((rc=CloseHandle(hnd)) == 0)
\
        API_ERROR(__LINE__, "CloseHandle", (rc=GEN_ERRCODE));

/*
****
**/

```

```

/* Common I/O Macros and Definitions
**/
/*
****
**/
/
#define IOH_FILE 1
#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

```

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(int x, int y);
int rand_integer( int val_lo, int val_hi );
int NURand_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

```

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 tpccctx.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

```

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);
}

```

Appendix C : Tunable Parameters

Database Configuration

dbm.cfg.out

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) = 2.282997e-007

Max number of concurrently active databases      (NUMDB) = 1
Data Links support                             (DATA LINKS) = 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) = 360
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 (KEEPPENCED) = 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

db.cfg.out

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) =
 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 = YES

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) = 16384
 Size of database shared memory (4KB) (DATABASE_MEMORY) = AUTOMATIC
 Catalog cache size (4KB) (CATALOGCACHE_SZ) = (MAXAPPLS*4)
 Log buffer size (4KB) (LOGBUFSZ) = 512
 Utilities heap size (4KB) (UTIL_HEAP_SZ) = 524288
 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) = 2880
 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) = 16
 SQL statement heap (4KB) (STMTHAP) = 16384
 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) = 6
 Number of I/O servers (NUM_IOSERVERS) = 1
 Index sort flag (INDEXSORT) = YES
 Sequential detect flag (SEQDETECT) = NO
 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) = 360
 Average number of active applications (AVG_APPLS) = 1
 Max DB files open per application (MAXFILOP) = 800
 Log file size (4KB) (LOGFILSIZ) = 262144
 Number of primary log files (LOGPRIMARY) = 250
 Number of secondary log files (LOGSECOND) = 0


```

Changed path to log files          (NEWLOGPATH) =
Path to log files                  = \\.\Q:
Overflow log path                  (OVERFLOWLOGPATH) =
Mirror log path                   (MIRRORLOGPATH) =

First active log file              = S0000033.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) = 3
Percent log file reclaimed before soft ckcpt (SOFTMAX) = 1600
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.cfg.out

```

[e] DB2PATH=C:\SQLLIB
[i] DB2_RESOURCE_POLICY=C:\nec8tpcc\tpc-c.ibm\resorceset\aff.cfg
[i] DB2_SELUDI_COMM_BUFFER=Y

```

```

[i] DB2_USE_ALTERNATE_PAGE_CLEANING=YES
[i] DB2_MAX_NON_TABLE_LOCKS=500
[i] DB2_TRUSTED_BINDIN=ON
[i] DB2_KEEPTABLELOCK=ON
[i] DB2_NO_FORK_CHECK=ON
[i] DB2_APM_PERFORMANCE=ALL
[i] DB2_ENABLE_BUFDPD=OFF
[i] DB2_SELECTIVITY=ON
[i] DB2ASSUMEUPDATE=ON
[i] DB2CHECKCLIENTINTERVAL=0
[i] DB2_HASH_JOIN=OFF
[i] DB2CHKSQLDA=OFF
[i] DB2_COLLECT_TS_REC_INFO=false
[i] DB2NTNOCACHE=ON
[i] DB2COMM=tcPIP
[i] DB2CHKPTR=OFF
[g] DB2_EXTSECURITY=YES
[g] DB2SYSTEM=TIGER
[g] DB2PATH=C:\SQLLIB
[g] DB2INSTEDEF=DB2
[g] DB2ADMINSERVER=DB2DAS00

```

aff.cfg

```

<RESOURCE_POLICY>
<DATABASE_RESOURCE_POLICY>
<DBNAME>TPCC</DBNAME>
<METHOD>CPUMASK</METHOD>
<RESOURCE_BINDING>
<RESOURCE>3</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>8</BUFFERPOOL_ID>
<BUFFERPOOL_ID>13</BUFFERPOOL_ID>
<BUFFERPOOL_ID>17</BUFFERPOOL_ID>
<BUFFERPOOL_ID>21</BUFFERPOOL_ID>
<BUFFERPOOL_ID>25</BUFFERPOOL_ID>
<BUFFERPOOL_ID>29</BUFFERPOOL_ID>
<BUFFERPOOL_ID>33</BUFFERPOOL_ID>
<BUFFERPOOL_ID>37</BUFFERPOOL_ID>
<BUFFERPOOL_ID>41</BUFFERPOOL_ID>
</BUFFERPOOL_BINDING>
</RESOURCE_BINDING>
<RESOURCE_BINDING>
<RESOURCE>12</RESOURCE>
<DBMEM_PERCENTAGE>0.0</DBMEM_PERCENTAGE>
<SERVICE_NAME>5550</SERVICE_NAME>
<BUFFERPOOL_BINDING>
<NUM_CLEANERS>12</NUM_CLEANERS>
<BUFFERPOOL_ID>5</BUFFERPOOL_ID>
<BUFFERPOOL_ID>9</BUFFERPOOL_ID>
<BUFFERPOOL_ID>14</BUFFERPOOL_ID>
<BUFFERPOOL_ID>18</BUFFERPOOL_ID>
<BUFFERPOOL_ID>22</BUFFERPOOL_ID>
<BUFFERPOOL_ID>26</BUFFERPOOL_ID>
<BUFFERPOOL_ID>30</BUFFERPOOL_ID>

```

```

<BUFFERPOOL_ID>34</BUFFERPOOL_ID>
<BUFFERPOOL_ID>38</BUFFERPOOL_ID>
<BUFFERPOOL_ID>42</BUFFERPOOL_ID>
</BUFFERPOOL_BINDING>
</RESOURCE_BINDING>
<RESOURCE_BINDING>
<RESOURCE>48</RESOURCE>
<DBMEM_PERCENTAGE>0.0</DBMEM_PERCENTAGE>
<SERVICE_NAME>6550</SERVICE_NAME>
<BUFFERPOOL_BINDING>
<NUM_CLEANERS>12</NUM_CLEANERS>
<BUFFERPOOL_ID>6</BUFFERPOOL_ID>
<BUFFERPOOL_ID>10</BUFFERPOOL_ID>
<BUFFERPOOL_ID>15</BUFFERPOOL_ID>
<BUFFERPOOL_ID>19</BUFFERPOOL_ID>
<BUFFERPOOL_ID>23</BUFFERPOOL_ID>
<BUFFERPOOL_ID>27</BUFFERPOOL_ID>
<BUFFERPOOL_ID>31</BUFFERPOOL_ID>
<BUFFERPOOL_ID>35</BUFFERPOOL_ID>
<BUFFERPOOL_ID>39</BUFFERPOOL_ID>
<BUFFERPOOL_ID>43</BUFFERPOOL_ID>
</BUFFERPOOL_BINDING>
</RESOURCE_BINDING>
<RESOURCE_BINDING>
<RESOURCE>192</RESOURCE>
<DBMEM_PERCENTAGE>0.0</DBMEM_PERCENTAGE>
<SERVICE_NAME>7550</SERVICE_NAME>
<BUFFERPOOL_BINDING>
<NUM_CLEANERS>12</NUM_CLEANERS>
<BUFFERPOOL_ID>7</BUFFERPOOL_ID>
<BUFFERPOOL_ID>11</BUFFERPOOL_ID>
<BUFFERPOOL_ID>16</BUFFERPOOL_ID>
<BUFFERPOOL_ID>20</BUFFERPOOL_ID>
<BUFFERPOOL_ID>24</BUFFERPOOL_ID>
<BUFFERPOOL_ID>28</BUFFERPOOL_ID>
<BUFFERPOOL_ID>32</BUFFERPOOL_ID>
<BUFFERPOOL_ID>36</BUFFERPOOL_ID>
<BUFFERPOOL_ID>40</BUFFERPOOL_ID>
<BUFFERPOOL_ID>44</BUFFERPOOL_ID>
</BUFFERPOOL_BINDING>
</RESOURCE_BINDING>
</DATABASE_RESOURCE_POLICY>
</RESOURCE_POLICY>

```

Server Configuration

Services

These Windows services are started:

- Automatic Updates
- COM+ Event System
- Computer Browser
- Cryptographic Services
- DB2 JDBC Applet Server
- DB2 License Server
- DB2 Registry Reflector

- DB2 Remote Command Server
- DB2 Security Server
- DB2DAS - DB2DAS00
- DHCP Client
- Distributed File System
- Distributed Link Tracking Client
- Distributed Transaction Coordinator
- DNS Client
- Error Reporting Service
- Event Log
- Help and Support
- IPSEC Services
- Logical Disk Manager
- Plug and Play
- Print Spooler
- Protected Storage
- Remote Procedure Call (RPC)
- Remote Registry
- Secondary Logon
- Security Accounts Manager
- Server
- Shell Hardware Detection
- Special Administration Console Helper
- System Event Notification
- Task Scheduler
- TCP/IP NetBIOS Helper
- Terminal Services
- Windows Management Instrumentation
- Windows Time
- Wireless Configuration
- Workstation

System information

System Information report written at: 12/24/04 11:37:10

System Name: TIGER

[System Summary]

Item	Value
OS Name	Microsoft(R) Windows(R) Server 2003, Datacenter Edition
Version	5.2.3790 Build 3790
OS Manufacturer	Microsoft Corporation
System Name	TIGER
System Manufacturer	NEC
System Model	server u32000
System Type	Itanium (TM) -based System
Processor	ia64 Family 31 Model 2 Stepping 1 GenuineIntel ~1600 Mhz
Processor	ia64 Family 31 Model 2 Stepping 1 GenuineIntel ~1600 Mhz
Processor	ia64 Family 31 Model 2 Stepping 1 GenuineIntel ~1600 Mhz
Processor	ia64 Family 31 Model 2 Stepping 1 GenuineIntel ~1600 Mhz
Processor	ia64 Family 31 Model 2 Stepping 1 GenuineIntel ~1600 Mhz

Processor ia64 Family 31 Model 2 Stepping 1 GenuineIntel ~1600 Mhz
 Processor ia64 Family 31 Model 2 Stepping 1 GenuineIntel ~1600 Mhz
 BIOS Version/Date Phoenix Technologies LTD 6.0.0602, 8/12/2004
 SMBIOS Version 2.3
 Windows Directory C:\WINDOWS
 System Directory C:\WINDOWS\system32
 Boot Device \Device\HarddiskVolume53
 Locale United States
 Hardware Abstraction Layer Version = "5.2.3790.0 (srv03_rtm.030324-2048)"
 User Name TIGER\nec
 Time Zone Tokyo Standard Time
 Total Physical Memory 262,113.84 MB
 Available Physical Memory 252.64 GB
 Total Virtual Memory 511.07 GB
 Available Virtual Memory 507.32 GB
 Page File Space 255.10 GB
 Page File C:\pagefile.sys

[Hardware Resources]

[Conflicts/Sharing]

Resource	Device
I/O Port 0x00000000-0x000003AF	PCI bus
I/O Port 0x00000000-0x000003AF	Direct memory access controller
I/O Port 0x00001B00-0x00001BFF	PCI bus
I/O Port 0x00001B00-0x00001BFF	QLogic QLA23xx PCI Fibre Channel Adapter
I/O Port 0x000003C0-0x000003DF	PCI bus
I/O Port 0x000003C0-0x000003DF	Standard VGA Graphics Adapter
I/O Port 0x00001700-0x000017FF	PCI bus
I/O Port 0x00001700-0x000017FF	QLogic QLA23xx PCI Fibre Channel Adapter
I/O Port 0x00001300-0x000014FF	PCI bus
I/O Port 0x00001300-0x000014FF	Adaptec SCSI Card 39160 - Ultra160 SCSI
I/O Port 0x00001A00-0x00001AFF	PCI bus
I/O Port 0x00001A00-0x00001AFF	QLogic QLA23xx PCI Fibre Channel Adapter
I/O Port 0x00001600-0x000016FF	PCI bus
I/O Port 0x00001600-0x000016FF	QLogic QLA23xx PCI Fibre Channel Adapter
I/O Port 0x00001D00-0x00001DFF	PCI bus
I/O Port 0x00001D00-0x00001DFF #2	Intel(R) PRO/1000 XT Server Adapter

I/O Port 0x00001900-0x000019FF	PCI bus
I/O Port 0x00001900-0x000019FF	QLogic QLA23xx PCI Fibre Channel Adapter
I/O Port 0x00001500-0x000015FF	PCI bus
I/O Port 0x00001500-0x000015FF	QLogic QLA23xx PCI Fibre Channel Adapter
I/O Port 0x00001C00-0x00001CFF	PCI bus
I/O Port 0x00001C00-0x00001CFF	QLogic QLA23xx PCI Fibre Channel Adapter
I/O Port 0x00001800-0x000018FF	PCI bus
I/O Port 0x00001800-0x000018FF	QLogic QLA23xx PCI Fibre Channel Adapter
Memory Address 0xA0000-0xBFFFF	PCI bus
Memory Address 0xA0000-0xBFFFF	Standard VGA Graphics Adapter
I/O Port 0x00001200-0x000012FF	PCI bus
I/O Port 0x00001200-0x000012FF	QLogic QLA23xx PCI Fibre Channel Adapter
I/O Port 0x000003B0-0x000003BB	PCI bus
I/O Port 0x000003B0-0x000003BB	Standard VGA Graphics Adapter
I/O Port 0x00001E00-0x0000FFFF	PCI bus
I/O Port 0x00001E00-0x0000FFFF	QLogic QLA23xx PCI Fibre Channel Adapter
I/O Port 0x00000D00-0x000011FF	PCI bus
I/O Port 0x00000D00-0x000011FF	Extended IO Bus

[DMA]

Resource	Device	Status
Channel 4	Direct memory access controller	OK

[Forced Hardware]

Device PNP Device ID

[I/O]

Resource	Device	Status
0x00000000-0x000003AF	PCI bus	OK
0x00000000-0x000003AF	Direct memory access controller	OK
0x000003B0-0x000003BB	PCI bus	OK
0x000003B0-0x000003BB	Standard VGA Graphics Adapter	OK
0x000003BC-0x000003BF	PCI bus	OK
0x000003C0-0x000003DF	PCI bus	OK
0x000003C0-0x000003DF	Standard VGA Graphics Adapter	OK
0x000003E0-0x00000CF7	PCI bus	OK
0x00000D00-0x000011FF	PCI bus	OK
0x00000D00-0x000011FF	Extended IO Bus	OK
0x00000080-0x0000008F	Direct memory access controller	OK
0x000000C0-0x000000DF	Direct memory access controller	OK
0x00000020-0x00000021	Programmable interrupt controller	OK

0x000000A0-0x000000A1	Programmable interrupt controller	OK
0x00000040-0x00000043	System timer	OK
0x00000061-0x00000061	System speaker	OK
0x00000070-0x00000071	System CMOS/real time clock	OK
0x000000F0-0x000000FF	Numeric data processor	OK
0x000000B2-0x000000B3	Generic Bus	OK
0x000000E00-0x000000E3F	Generic Bus	OK
0x00000092-0x00000092	Motherboard resources	OK
0x0000004D0-0x0000004D1	Motherboard resources	OK
0x000000D80-0x000000D80	Motherboard resources	OK
0x000000D81-0x000000D81	Motherboard resources	OK
0x000000CA0-0x000000CA1	Motherboard resources	OK
0x000000CA4-0x000000CA7	Motherboard resources	OK
0x00000002E-0x00000002F	Motherboard resources	OK
0x000000060-0x000000060	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard	OK
0x000000064-0x000000064	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard	OK
0x0000003F8-0x0000003FF	Communications Port (COM1)	OK
0x000000C00-0x000000C03	Extended IO Bus	OK
0x000000F00-0x000000F7F	Extended IO Bus	OK
0x000000F80-0x000000FFF	Extended IO Bus	OK
0x000000CA2-0x000000CA2	OK	
0x000000CA3-0x000000CA3	OK	
0x000001120-0x00000112F	Intel(r) IA64 Bus Master IDE Controller	OK
0x000001F0-0x000001F7	Primary IDE Channel	OK
0x000003F6-0x000003F6	Primary IDE Channel	OK
0x000001100-0x00000111F	Intel(r) 82372FB PCI to USB Universal Host Controller	OK
0x000001000-0x0000010FF	Standard VGA Graphics Adapter	OK
0x000001200-0x0000012FF	PCI bus OK	
0x000001200-0x0000012FF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
0x000001300-0x0000014FF	PCI bus OK	
0x000001300-0x0000014FF	Adaptec SCSI Card 39160 - Ultra160 SCSI	OK
0x000001400-0x0000014FF	Adaptec SCSI Card 39160 - Ultra160 SCSI	OK
0x000001500-0x0000015FF	PCI bus OK	
0x000001500-0x0000015FF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
0x000001600-0x0000016FF	PCI bus OK	
0x000001600-0x0000016FF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
0x000001700-0x0000017FF	PCI bus OK	
0x000001700-0x0000017FF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
0x000001800-0x0000018FF	PCI bus OK	
0x000001800-0x0000018FF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
0x000001900-0x0000019FF	PCI bus OK	
0x000001900-0x0000019FF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
0x000001A00-0x000001AFF	PCI bus OK	
0x000001A00-0x000001AFF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
0x000001B00-0x000001BFF	PCI bus OK	

0x000001B00-0x000001BFF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
0x000001C00-0x000001CFF	PCI bus OK	
0x000001C00-0x000001CFF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
0x000001D00-0x000001DFF	PCI bus OK	
0x000001D00-0x000001DFF	Intel(R) PR0/1000 XT Server Adapter #2	OK
0x000001E00-0x000001FFF	PCI bus OK	
0x000001E00-0x000001FFF	QLogic QLA23xx PCI Fibre Channel Adapter	OK

[IRQs]

Resource	Device	Status
IRQ 9	Microsoft ACPI-Compliant System	OK
IRQ 0	System timer	OK
IRQ 8	System CMOS/real time clock	OK
IRQ 13	Numeric data processor	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 11	OK	
IRQ 14	Primary IDE Channel	OK
IRQ 19	Intel(r) 82372FB PCI to USB Universal Host Controller	OK

IRQ 20	QLogic QLA23xx PCI Fibre Channel Adapter	OK
IRQ 212	Adaptec SCSI Card 39160 - Ultra160 SCSI	OK
IRQ 213	Adaptec SCSI Card 39160 - Ultra160 SCSI	OK
IRQ 272	QLogic QLA23xx PCI Fibre Channel Adapter	OK
IRQ 276	QLogic QLA23xx PCI Fibre Channel Adapter	OK
IRQ 468	QLogic QLA23xx PCI Fibre Channel Adapter	OK
IRQ 596	QLogic QLA23xx PCI Fibre Channel Adapter	OK
IRQ 656	QLogic QLA23xx PCI Fibre Channel Adapter	OK
IRQ 660	QLogic QLA23xx PCI Fibre Channel Adapter	OK
IRQ 784	QLogic QLA23xx PCI Fibre Channel Adapter	OK
IRQ 788	QLogic QLA23xx PCI Fibre Channel Adapter	OK
IRQ 913	Intel(R) PR0/1000 XT Server Adapter #2	OK
IRQ 916	QLogic QLA23xx PCI Fibre Channel Adapter	OK

[Memory]

Resource	Device	Status
0xA0000-0xBFFFF	PCI bus OK	
0xA0000-0xBFFFF	Standard VGA Graphics Adapter	OK
0xC0000-0xC3FFF	PCI bus OK	
0xC4000-0xC7FFF	PCI bus OK	
0xC8000-0xCBFFF	PCI bus OK	
0xCC000-0xCFFFF	PCI bus OK	
0xD0000-0xD3FFF	PCI bus OK	
0xD4000-0xD7FFF	PCI bus OK	
0xD8000-0xDBFFF	PCI bus OK	
0xDC000-0xDFFFF	PCI bus OK	
0xE0000-0xE3FFF	PCI bus OK	
0xE4000-0xE7FFF	PCI bus OK	
0xE8000-0xEBFFF	PCI bus OK	
0xEC000-0xEFFFF	PCI bus OK	
0xF0000-0xFFFFF	PCI bus OK	

0xFBC00000-0xFDFFFFFF	PCI bus OK	
0xFC000000-0xFCFFFFFF	Standard VGA Graphics Adapter OK	
0xFDFFF000-0xFDFFFFFF	Standard VGA Graphics Adapter OK	
0xFB800000-0xFBFFFFFF	PCI bus OK	
0xFBBFF000-0xFBBFFFFFF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
0xFB400000-0xFB7FFFFF	PCI bus OK	
0xFB7FF000-0xFB7FFFFF	Adaptec SCSI Card 39160 - Ultra160 SCSI	OK
0xFB7FE000-0xFB7FEFFF	Adaptec SCSI Card 39160 - Ultra160 SCSI	OK
0xFB000000-0xFB3FFFFF	PCI bus OK	
0xFB3FF000-0xFB3FFFFF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
0xFAC00000-0xFAFFFFFF	PCI bus OK	
0xFAFFF000-0xFAFFFFFF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
0xFA800000-0xFABFFFFF	PCI bus OK	
0xFABFF000-0xFABFFFFF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
0xFA400000-0xFA7FFFFF	PCI bus OK	
0xFA7FF000-0xFA7FFFFF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
0xFA000000-0xFA3FFFFF	PCI bus OK	
0xFA3FF000-0xFA3FFFFF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
0xF9C00000-0xF9FFFFF	PCI bus OK	
0xF9FFF000-0xF9FFFFF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
0xF9800000-0xF9BFFFFF	PCI bus OK	
0xF9BFF000-0xF9BFFFFF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
0xF9400000-0xF97FFFFF	PCI bus OK	
0xF97FF000-0xF97FFFFF	QLogic QLA23xx PCI Fibre Channel Adapter	OK
0xF9000000-0xF93FFFFF	PCI bus OK	
0xF93C0000-0xF93DFFFF	Intel(R) PRO/1000 XT Server Adapter #2	OK
0xF93E0000-0xF93FFFFF	Intel(R) PRO/1000 XT Server Adapter #2	OK
0xE0000000-0xF8FFFFFF	PCI bus OK	
0xF8FFF000-0xF8FFFFFF	QLogic QLA23xx PCI Fibre Channel Adapter	OK

[Components]

[Multimedia]

[Audio Codecs]

CODEC	Manufacturer	Description	Status	File	Version	Size
		Creation Date				
c:\windows\system32\msgsm32.acm	Microsoft Corporation		OK			
	C:\WINDOWS\system32\MSGSM32.ACM			5.2.3790.0		

(srv03_rtm.030324-2048)	66.50 KB (68,096 bytes)	3/25/2003
09:00 PM		
c:\windows\system32\imaadp32.acm	Microsoft Corporation	OK
	C:\WINDOWS\system32\IMAADP32.ACM	5.2.3790.0
(srv03_rtm.030324-2048)	55.00 KB (56,320 bytes)	3/25/2003
09:00 PM		
c:\windows\system32\tssoft32.acm	DSP GROUP, INC.	OK
	C:\WINDOWS\system32\TSSOFT32.ACM	1.01 29.00 KB (29,696 bytes)
		3/25/2003 09:00 PM
c:\windows\system32\msg711.acm	Microsoft Corporation	OK
	C:\WINDOWS\system32\MSG711.ACM	5.2.3790.0
(srv03_rtm.030324-2048)	33.00 KB (33,792 bytes)	3/25/2003
09:00 PM		
c:\windows\system32\msadp32.acm	Microsoft Corporation	OK
	C:\WINDOWS\system32\MSADP32.ACM	5.2.3790.0
(srv03_rtm.030324-2048)	49.00 KB (50,176 bytes)	3/25/2003
09:00 PM		

[Video Codecs]

CODEC	Manufacturer	Description	Status	File	Version	Size
		Creation Date				
c:\windows\system32\msvidc32.dll	Microsoft Corporation		OK			
	C:\WINDOWS\system32\MSVIDC32.DLL			5.2.3790.0		
(srv03_rtm.030324-2048)	67.00 KB (68,608 bytes)	3/25/2003				
09:00 PM						
c:\windows\system32\msrle32.dll	Microsoft Corporation		OK			
	C:\WINDOWS\system32\MSRLE32.DLL			5.2.3790.0		
(srv03_rtm.030324-2048)	24.50 KB (25,088 bytes)	3/25/2003				
09:00 PM						

[CD-ROM]

Item	Value
Drive D:	
Description	CD-ROM Drive
Media Loaded	No
Media Type	CD-ROM
Name	_NEC DVD-ROM DV-5800A
Manufacturer	(Standard CD-ROM drives)
Status	OK
Transfer Rate	Not Available
SCSI Target ID	0
PNP Device ID	IDE\CDROM_NEC_DVD-ROM_DV-5800A_____1.42____\5&8F26A51&0&0.0.0

[Sound Device]

Item	Value
------	-------

[Display]

Item	Value
Name	Standard VGA Graphics Adapter
PNP Device ID	PCI\VEN_1002&DEV_4752&SUBSYS_80EB1033&REV_27\3&267A616A&0&20
Adapter Type	ATI MACH64, (Standard display types) compatible
Adapter Description	Standard VGA Graphics Adapter

```

Adapter RAM      4.00 MB (4,194,304 bytes)
Installed Drivers  vga.dll,framebuf.dll,vga256,vga64k
Driver Version    5.2.3790.0
INF File          display.inf (vga section)
Color Planes      1
Color Table Entries 16777216
Resolution        1024 x 768 x 1 hertz
Bits/Pixel        24
Memory Address     0xFC000000-0xFCFFFFFF
I/O Port          0x00001000-0x000010FF
Memory Address     0xFDFFF000-0xFDFFFFFF
I/O Port          0x000003B0-0x000003BB
I/O Port          0x000003C0-0x000003DF
Memory Address     0xA0000-0xBFFFF

[Infrared]

Item      Value

[Input]

[Keyboard]

Item      Value
Description  Standard 101/102-Key or Microsoft Natural PS/2 Keyboard

Name      Enhanced (101- or 102-key)
Layout     00000409
PNP Device ID  ACPI\PNP0303\2
Number of Function Keys 12
I/O Port     0x00000060-0x00000060
I/O Port     0x00000064-0x00000064
IRQ Channel   IRQ 1

[Pointing Device]

Item      Value
Hardware Type  PS/2 Compatible Mouse
Number of Buttons  2
Status        OK
PNP Device ID  ACPI\PNP0F13\2
Power Management Supported  No
Double Click Threshold 6
Handedness     Right Handed Operation
IRQ Channel     IRQ 12

[Modem]

Item      Value

[Network]

[Adapter]

Item      Value

```

```

Name [00000001] Intel(R) PRO/1000 XT Server Adapter
Adapter Type  Not Available
Product Type  Intel(R) PRO/1000 XT Server Adapter
Installed     Yes
PNP Device ID Not Available
Last Reset    12/24/2004 11:16 AM
Index         1
Service Name   E1000
IP Address     Not Available
IP Subnet      Not Available
Default IP Gateway  Not Available
DHCP Enabled   No
DHCP Server    Not Available
DHCP Lease Expires  Not Available
DHCP Lease Obtained Not Available
MAC Address    Not Available

Name [00000002] RAS Async Adapter
Adapter Type  Not Available
Product Type  RAS Async Adapter
Installed     Yes
PNP Device ID Not Available
Last Reset    12/24/2004 11:16 AM
Index         2
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 [00000003] WAN Miniport (L2TP)
Adapter Type  Not Available
Product Type  WAN Miniport (L2TP)
Installed     Yes
PNP Device ID ROOT\MS_L2TPMINIPORT\0000
Last Reset    12/24/2004 11:16 AM
Index         3
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 [00000004] WAN Miniport (PPTP)
Adapter Type  Wide Area Network (WAN)
Product Type  WAN Miniport (PPTP)
Installed     Yes
PNP Device ID ROOT\MS_PPTPMINIPORT\0000
Last Reset    12/24/2004 11:16 AM
Index         4
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 [00000005] 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 12/24/2004 11:16 AM
Index 5
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 [00000006] Direct Parallel
Adapter Type Not Available
Product Type Direct Parallel
Installed Yes
PNP Device ID ROOT\MS_PTIMINIPOINT\0000
Last Reset 12/24/2004 11:16 AM
Index 6
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 [00000007] WAN Miniport (IP)
Adapter Type Not Available
Product Type WAN Miniport (IP)
Installed Yes
PNP Device ID ROOT\MS_NDISWANIP\0000
Last Reset 12/24/2004 11:16 AM
Index 7
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 [00000008] Intel(R) PRO/1000 XT Server Adapter
Adapter Type Ethernet 802.3
Product Type Intel(R) PRO/1000 XT Server Adapter
Installed Yes
PNP Device ID
PCI\VEN_8086&DEV_1008&SUBSYS_11078086&REV_02\3&27265218&0&18

Last Reset 12/24/2004 11:16 AM
Index 8
Service Name E1000
IP Address 10.1.1.222
IP Subnet 255.255.255.0
Default IP Gateway Not Available
DHCP Enabled No
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address 00:02:B3:CC:50:EA
Memory Address 0xF93C0000-0xF93DFFFF
Memory Address 0xF93E0000-0xF93FFFFFFF
I/O Port 0x00001D00-0x00001DFF
IRQ Channel IRQ 913

[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\wssock32.dll
Size 23.00 KB (23,552 bytes)
Version 5.2.3790.0 (srv03_rtm.030324-2048)

[Ports]

[Serial]

Item Value
Name Communications Port (COM1)
Status OK
PNP Device ID ACPI\PNP0501\2
Maximum Input Buffer Size 0
Maximum Output Buffer Size No

Settable Baud Rate Yes
Settable Data Bits Yes
Settable Flow Control Yes
Settable Parity Yes
Settable Parity Check Yes
Settable Stop Bits Yes
Settable RLSD Yes
Supports RLSD Yes
Supports 16 Bit Mode No
Supports Special Characters No
Baud Rate 9600
Bits/Byte 8
Stop Bits 1
Parity None
Busy No
Abort Read/Write on Error No
Binary Mode Enabled Yes
Continue Xmit on XOff No
CTS Outflow Control No
Discard NULL Bytes No
DSR Outflow Control 0
DSR Sensitivity 0
DTR Flow Control Type Enable
EOF Character 0
Error Replace Character 0
Error Replacement Enabled No
Event Character 0
Parity Check Enabled No
RTS Flow Control Type Enable
XOff Character 19
XOffXmit Threshold 512
XOn Character 17
XOnXmit Threshold 2048
XOnXOff InFlow Control 0
XOnXOff OutFlow Control 0
I/O Port 0x000003F8-0x000003FF
IRQ Channel IRQ 4

[Parallel]

Item Value

[Storage]

[Drives]

Item Value
Drive C:
Description Local Fixed Disk
Compressed No
File System NTFS
Size 33.41 GB (35,878,670,336 bytes)
Free Space 20.70 GB (22,222,811,136 bytes)
Volume Name
Volume Serial Number 685F38B0

Drive D:

Description CD-ROM Disc

Drive G:
Description Local Fixed Disk
Compressed No
File System NTFS
Size 189.93 GB (203,937,591,296 bytes)
Free Space 189.86 GB (203,863,752,704 bytes)
Volume Name New Volume
Volume Serial Number 9CE1F44C

Drive Q:
Description Local Fixed Disk
Compressed Not Available
File System Not Available
Size Not Available
Free Space Not Available
Volume Name Not Available
Volume Serial Number Not Available

Drive S:
Description Local Fixed Disk
Compressed No
File System NTFS
Size 9.77 GB (10,487,197,696 bytes)
Free Space 7.58 GB (8,136,314,880 bytes)
Volume Name db2junc
Volume Serial Number 204B48CC

Drive X:
Description Network Connection
Provider Name \\10.1.1.48\c\$

Drive Y:
Description Network Connection
Provider Name \\10.1.1.47\c\$

[Disks]

Item	Value
Description	Disk drive
Manufacturer	(Standard disk drives)
Model	NEC iStorage 2000 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	498.92 GB (535,712,486,400 bytes)
Total Cylinders	65,130
Total Sectors	1,046,313,450
Total Tracks	16,608,150
Tracks/Cylinder	255
Partition	Disk #4, Partition #0
Partition Size	298.92 GB (320,958,650,880 bytes)

Partition	Starting Offset	214,753,835,520 bytes
Partition	Disk #4, Partition #1	
Partition	Size	2.01 GB (2,154,991,104 bytes)
Partition	Starting Offset	8,257,536 bytes
Partition	Disk #4, Partition #2	
Partition	Size	2.01 GB (2,154,991,104 bytes)
Partition	Starting Offset	2,163,280,896 bytes
Partition	Disk #4, Partition #3	
Partition	Size	2.01 GB (2,154,991,104 bytes)
Partition	Starting Offset	4,318,304,256 bytes
Partition	Disk #4, Partition #4	
Partition	Size	24.01 GB (25,777,995,264 bytes)
Partition	Starting Offset	6,473,327,616 bytes
Partition	Disk #4, Partition #5	
Partition	Size	14.00 GB (15,035,779,584 bytes)
Partition	Starting Offset	32,251,355,136 bytes
Partition	Disk #4, Partition #6	
Partition	Size	2.01 GB (2,154,991,104 bytes)
Partition	Starting Offset	47,287,166,976 bytes
Partition	Disk #4, Partition #7	
Partition	Size	4.01 GB (4,301,789,184 bytes)
Partition	Starting Offset	49,442,190,336 bytes
Partition	Disk #4, Partition #8	
Partition	Size	2.01 GB (2,154,991,104 bytes)
Partition	Starting Offset	53,744,011,776 bytes
Partition	Disk #4, Partition #9	
Partition	Size	2.01 GB (2,154,991,104 bytes)
Partition	Starting Offset	55,899,035,136 bytes
Partition	Disk #4, Partition #10	
Partition	Size	28.01 GB (30,071,591,424 bytes)
Partition	Starting Offset	58,054,058,496 bytes
Partition	Disk #4, Partition #11	
Partition	Size	20.00 GB (21,476,173,824 bytes)
Partition	Starting Offset	88,125,682,176 bytes
Partition	Disk #4, Partition #12	
Partition	Size	4.01 GB (4,301,789,184 bytes)
Partition	Starting Offset	109,601,888,256 bytes

Description	Disk drive	
Manufacturer	(Standard disk drives)	
Model	NEC iStorage 2000 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	2	
SCSI Target ID	0	
Sectors/Track	63	
Size	498.92 GB (535,712,486,400 bytes)	
Total Cylinders	65,130	
Total Sectors	1,046,313,450	
Total Tracks	16,608,150	
Tracks/Cylinder	255	
Partition	Disk #5, Partition #0	
Partition Size	298.92 GB (320,958,650,880 bytes)	
Partition	Starting Offset	214,753,835,520 bytes
Partition	Disk #5, Partition #1	

Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 8,257,536 bytes
 Partition Disk #5, Partition #2
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 2,163,280,896 bytes
 Partition Disk #5, Partition #3
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 4,318,304,256 bytes
 Partition Disk #5, Partition #4
 Partition Size 24.01 GB (25,777,995,264 bytes)
 Partition Starting Offset 6,473,327,616 bytes
 Partition Disk #5, Partition #5
 Partition Size 14.00 GB (15,035,779,584 bytes)
 Partition Starting Offset 32,251,355,136 bytes
 Partition Disk #5, Partition #6
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 47,287,166,976 bytes
 Partition Disk #5, Partition #7
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 49,442,190,336 bytes
 Partition Disk #5, Partition #8
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 53,744,011,776 bytes
 Partition Disk #5, Partition #9
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 55,899,035,136 bytes
 Partition Disk #5, Partition #10
 Partition Size 28.01 GB (30,071,591,424 bytes)
 Partition Starting Offset 58,054,058,496 bytes
 Partition Disk #5, Partition #11
 Partition Size 20.00 GB (21,476,173,824 bytes)
 Partition Starting Offset 88,125,682,176 bytes
 Partition Disk #5, Partition #12
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 109,601,888,256 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model NEC iStorage 2000 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 1
 Sectors/Track 63
 Size 498.92 GB (535,712,486,400 bytes)
 Total Cylinders 65,130
 Total Sectors 1,046,313,450
 Total Tracks 16,608,150
 Tracks/Cylinder 255
 Partition Disk #6, Partition #0
 Partition Size 298.92 GB (320,958,650,880 bytes)
 Partition Starting Offset 214,753,835,520 bytes
 Partition Disk #6, Partition #1
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 8,257,536 bytes

Partition Disk #6, Partition #2
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 2,163,280,896 bytes
 Partition Disk #6, Partition #3
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 4,318,304,256 bytes
 Partition Disk #6, Partition #4
 Partition Size 24.01 GB (25,777,995,264 bytes)
 Partition Starting Offset 6,473,327,616 bytes
 Partition Disk #6, Partition #5
 Partition Size 14.00 GB (15,035,779,584 bytes)
 Partition Starting Offset 32,251,355,136 bytes
 Partition Disk #6, Partition #6
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 47,287,166,976 bytes
 Partition Disk #6, Partition #7
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 49,442,190,336 bytes
 Partition Disk #6, Partition #8
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 53,744,011,776 bytes
 Partition Disk #6, Partition #9
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 55,899,035,136 bytes
 Partition Disk #6, Partition #10
 Partition Size 28.01 GB (30,071,591,424 bytes)
 Partition Starting Offset 58,054,058,496 bytes
 Partition Disk #6, Partition #11
 Partition Size 20.00 GB (21,476,173,824 bytes)
 Partition Starting Offset 88,125,682,176 bytes
 Partition Disk #6, Partition #12
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 109,601,888,256 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model NEC iStorage 2000 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 2
 SCSI Target ID 1
 Sectors/Track 63
 Size 498.92 GB (535,712,486,400 bytes)
 Total Cylinders 65,130
 Total Sectors 1,046,313,450
 Total Tracks 16,608,150
 Tracks/Cylinder 255
 Partition Disk #7, Partition #0
 Partition Size 298.92 GB (320,958,650,880 bytes)
 Partition Starting Offset 214,753,835,520 bytes
 Partition Disk #7, Partition #1
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 8,257,536 bytes
 Partition Disk #7, Partition #2
 Partition Size 2.01 GB (2,154,991,104 bytes)

Partition Starting Offset 2,163,280,896 bytes
 Partition Disk #7, Partition #3
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 4,318,304,256 bytes
 Partition Disk #7, Partition #4
 Partition Size 24.01 GB (25,777,995,264 bytes)
 Partition Starting Offset 6,473,327,616 bytes
 Partition Disk #7, Partition #5
 Partition Size 14.00 GB (15,035,779,584 bytes)
 Partition Starting Offset 32,251,355,136 bytes
 Partition Disk #7, Partition #6
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 47,287,166,976 bytes
 Partition Disk #7, Partition #7
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 49,442,190,336 bytes
 Partition Disk #7, Partition #8
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 53,744,011,776 bytes
 Partition Disk #7, Partition #9
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 55,899,035,136 bytes
 Partition Disk #7, Partition #10
 Partition Size 28.01 GB (30,071,591,424 bytes)
 Partition Starting Offset 58,054,058,496 bytes
 Partition Disk #7, Partition #11
 Partition Size 20.00 GB (21,476,173,824 bytes)
 Partition Starting Offset 88,125,682,176 bytes
 Partition Disk #7, Partition #12
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 109,601,888,256 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model NEC iStorage 2000 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 498.92 GB (535,712,486,400 bytes)
 Total Cylinders 65,130
 Total Sectors 1,046,313,450
 Total Tracks 16,608,150
 Tracks/Cylinder 255
 Partition Disk #30, Partition #0
 Partition Size 298.92 GB (320,958,650,880 bytes)
 Partition Starting Offset 214,753,835,520 bytes
 Partition Disk #30, Partition #1
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 8,257,536 bytes
 Partition Disk #30, Partition #2
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 2,163,280,896 bytes
 Partition Disk #30, Partition #3

Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 4,318,304,256 bytes
 Partition Disk #30, Partition #4
 Partition Size 24.01 GB (25,777,995,264 bytes)
 Partition Starting Offset 6,473,327,616 bytes
 Partition Disk #30, Partition #5
 Partition Size 14.00 GB (15,035,779,584 bytes)
 Partition Starting Offset 32,251,355,136 bytes
 Partition Disk #30, Partition #6
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 47,287,166,976 bytes
 Partition Disk #30, Partition #7
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 49,442,190,336 bytes
 Partition Disk #30, Partition #8
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 53,744,011,776 bytes
 Partition Disk #30, Partition #9
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 55,899,035,136 bytes
 Partition Disk #30, Partition #10
 Partition Size 28.01 GB (30,071,591,424 bytes)
 Partition Starting Offset 58,054,058,496 bytes
 Partition Disk #30, Partition #11
 Partition Size 20.00 GB (21,476,173,824 bytes)
 Partition Starting Offset 88,125,682,176 bytes
 Partition Disk #30, Partition #12
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 109,601,888,256 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model NEC iStorage 2000 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 9
 SCSI Target ID 0
 Sectors/Track 63
 Size 498.92 GB (535,712,486,400 bytes)
 Total Cylinders 65,130
 Total Sectors 1,046,313,450
 Total Tracks 16,608,150
 Tracks/Cylinder 255
 Partition Disk #31, Partition #0
 Partition Size 298.92 GB (320,958,650,880 bytes)
 Partition Starting Offset 214,753,835,520 bytes
 Partition Disk #31, Partition #1
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 8,257,536 bytes
 Partition Disk #31, Partition #2
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 2,163,280,896 bytes
 Partition Disk #31, Partition #3
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 4,318,304,256 bytes

Partition Disk #31, Partition #4
 Partition Size 24.01 GB (25,777,995,264 bytes)
 Partition Starting Offset 6,473,327,616 bytes
 Partition Disk #31, Partition #5
 Partition Size 14.00 GB (15,035,779,584 bytes)
 Partition Starting Offset 32,251,355,136 bytes
 Partition Disk #31, Partition #6
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 47,287,166,976 bytes
 Partition Disk #31, Partition #7
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 49,442,190,336 bytes
 Partition Disk #31, Partition #8
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 53,744,011,776 bytes
 Partition Disk #31, Partition #9
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 55,899,035,136 bytes
 Partition Disk #31, Partition #10
 Partition Size 28.01 GB (30,071,591,424 bytes)
 Partition Starting Offset 58,054,058,496 bytes
 Partition Disk #31, Partition #11
 Partition Size 20.00 GB (21,476,173,824 bytes)
 Partition Starting Offset 88,125,682,176 bytes
 Partition Disk #31, Partition #12
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 109,601,888,256 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model NEC iStorage 2000 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 1
 Sectors/Track 63
 Size 498.92 GB (535,712,486,400 bytes)
 Total Cylinders 65,130
 Total Sectors 1,046,313,450
 Total Tracks 16,608,150
 Tracks/Cylinder 255
 Partition Disk #32, Partition #0
 Partition Size 298.92 GB (320,958,650,880 bytes)
 Partition Starting Offset 214,753,835,520 bytes
 Partition Disk #32, Partition #1
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 8,257,536 bytes
 Partition Disk #32, Partition #2
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 2,163,280,896 bytes
 Partition Disk #32, Partition #3
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 4,318,304,256 bytes
 Partition Disk #32, Partition #4
 Partition Size 24.01 GB (25,777,995,264 bytes)

Partition Starting Offset 6,473,327,616 bytes
 Partition Disk #32, Partition #5
 Partition Size 14.00 GB (15,035,779,584 bytes)
 Partition Starting Offset 32,251,355,136 bytes
 Partition Disk #32, Partition #6
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 47,287,166,976 bytes
 Partition Disk #32, Partition #7
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 49,442,190,336 bytes
 Partition Disk #32, Partition #8
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 53,744,011,776 bytes
 Partition Disk #32, Partition #9
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 55,899,035,136 bytes
 Partition Disk #32, Partition #10
 Partition Size 28.01 GB (30,071,591,424 bytes)
 Partition Starting Offset 58,054,058,496 bytes
 Partition Disk #32, Partition #11
 Partition Size 20.00 GB (21,476,173,824 bytes)
 Partition Starting Offset 88,125,682,176 bytes
 Partition Disk #32, Partition #12
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 109,601,888,256 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model NEC iStorage 2000 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 9
 SCSI Target ID 1
 Sectors/Track 63
 Size 498.92 GB (535,712,486,400 bytes)
 Total Cylinders 65,130
 Total Sectors 1,046,313,450
 Total Tracks 16,608,150
 Tracks/Cylinder 255
 Partition Disk #33, Partition #0
 Partition Size 298.92 GB (320,958,650,880 bytes)
 Partition Starting Offset 214,753,835,520 bytes
 Partition Disk #33, Partition #1
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 8,257,536 bytes
 Partition Disk #33, Partition #2
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 2,163,280,896 bytes
 Partition Disk #33, Partition #3
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 4,318,304,256 bytes
 Partition Disk #33, Partition #4
 Partition Size 24.01 GB (25,777,995,264 bytes)
 Partition Starting Offset 6,473,327,616 bytes
 Partition Disk #33, Partition #5

Partition Size 14.00 GB (15,035,779,584 bytes)
 Partition Starting Offset 32,251,355,136 bytes
 Partition Disk #33, Partition #6
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 47,287,166,976 bytes
 Partition Disk #33, Partition #7
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 49,442,190,336 bytes
 Partition Disk #33, Partition #8
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 53,744,011,776 bytes
 Partition Disk #33, Partition #9
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 55,899,035,136 bytes
 Partition Disk #33, Partition #10
 Partition Size 28.01 GB (30,071,591,424 bytes)
 Partition Starting Offset 58,054,058,496 bytes
 Partition Disk #33, Partition #11
 Partition Size 20.00 GB (21,476,173,824 bytes)
 Partition Starting Offset 88,125,682,176 bytes
 Partition Disk #33, Partition #12
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 109,601,888,256 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model NEC iStorage 2000 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 8
 SCSI Target ID 0
 Sectors/Track 63
 Size 498.92 GB (535,712,486,400 bytes)
 Total Cylinders 65,130
 Total Sectors 1,046,313,450
 Total Tracks 16,608,150
 Tracks/Cylinder 255
 Partition Disk #26, Partition #0
 Partition Size 298.92 GB (320,958,650,880 bytes)
 Partition Starting Offset 214,753,835,520 bytes
 Partition Disk #26, Partition #1
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 8,257,536 bytes
 Partition Disk #26, Partition #2
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 2,163,280,896 bytes
 Partition Disk #26, Partition #3
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 4,318,304,256 bytes
 Partition Disk #26, Partition #4
 Partition Size 24.01 GB (25,777,995,264 bytes)
 Partition Starting Offset 6,473,327,616 bytes
 Partition Disk #26, Partition #5
 Partition Size 14.00 GB (15,035,779,584 bytes)
 Partition Starting Offset 32,251,355,136 bytes

Partition Disk #26, Partition #6
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 47,287,166,976 bytes
 Partition Disk #26, Partition #7
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 49,442,190,336 bytes
 Partition Disk #26, Partition #8
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 53,744,011,776 bytes
 Partition Disk #26, Partition #9
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 55,899,035,136 bytes
 Partition Disk #26, Partition #10
 Partition Size 28.01 GB (30,071,591,424 bytes)
 Partition Starting Offset 58,054,058,496 bytes
 Partition Disk #26, Partition #11
 Partition Size 20.00 GB (21,476,173,824 bytes)
 Partition Starting Offset 88,125,682,176 bytes
 Partition Disk #26, Partition #12
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 109,601,888,256 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model NEC iStorage 2000 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 498.92 GB (535,712,486,400 bytes)
 Total Cylinders 65,130
 Total Sectors 1,046,313,450
 Total Tracks 16,608,150
 Tracks/Cylinder 255
 Partition Disk #27, Partition #0
 Partition Size 298.92 GB (320,958,650,880 bytes)
 Partition Starting Offset 214,753,835,520 bytes
 Partition Disk #27, Partition #1
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 8,257,536 bytes
 Partition Disk #27, Partition #2
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 2,163,280,896 bytes
 Partition Disk #27, Partition #3
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 4,318,304,256 bytes
 Partition Disk #27, Partition #4
 Partition Size 24.01 GB (25,777,995,264 bytes)
 Partition Starting Offset 6,473,327,616 bytes
 Partition Disk #27, Partition #5
 Partition Size 14.00 GB (15,035,779,584 bytes)
 Partition Starting Offset 32,251,355,136 bytes
 Partition Disk #27, Partition #6
 Partition Size 2.01 GB (2,154,991,104 bytes)

Partition Starting Offset 47,287,166,976 bytes
 Partition Disk #27, Partition #7
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 49,442,190,336 bytes
 Partition Disk #27, Partition #8
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 53,744,011,776 bytes
 Partition Disk #27, Partition #9
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 55,899,035,136 bytes
 Partition Disk #27, Partition #10
 Partition Size 28.01 GB (30,071,591,424 bytes)
 Partition Starting Offset 58,054,058,496 bytes
 Partition Disk #27, Partition #11
 Partition Size 20.00 GB (21,476,173,824 bytes)
 Partition Starting Offset 88,125,682,176 bytes
 Partition Disk #27, Partition #12
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 109,601,888,256 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model NEC iStorage 2000 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 8
 SCSI Target ID 1
 Sectors/Track 63
 Size 498.92 GB (535,712,486,400 bytes)
 Total Cylinders 65,130
 Total Sectors 1,046,313,450
 Total Tracks 16,608,150
 Tracks/Cylinder 255
 Partition Disk #28, Partition #0
 Partition Size 298.92 GB (320,958,650,880 bytes)
 Partition Starting Offset 214,753,835,520 bytes
 Partition Disk #28, Partition #1
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 8,257,536 bytes
 Partition Disk #28, Partition #2
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 2,163,280,896 bytes
 Partition Disk #28, Partition #3
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 4,318,304,256 bytes
 Partition Disk #28, Partition #4
 Partition Size 24.01 GB (25,777,995,264 bytes)
 Partition Starting Offset 6,473,327,616 bytes
 Partition Disk #28, Partition #5
 Partition Size 14.00 GB (15,035,779,584 bytes)
 Partition Starting Offset 32,251,355,136 bytes
 Partition Disk #28, Partition #6
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 47,287,166,976 bytes
 Partition Disk #28, Partition #7

Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 49,442,190,336 bytes
 Partition Disk #28, Partition #8
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 53,744,011,776 bytes
 Partition Disk #28, Partition #9
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 55,899,035,136 bytes
 Partition Disk #28, Partition #10
 Partition Size 28.01 GB (30,071,591,424 bytes)
 Partition Starting Offset 58,054,058,496 bytes
 Partition Disk #28, Partition #11
 Partition Size 20.00 GB (21,476,173,824 bytes)
 Partition Starting Offset 88,125,682,176 bytes
 Partition Disk #28, Partition #12
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 109,601,888,256 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model NEC iStorage 2000 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 1
 Sectors/Track 63
 Size 498.92 GB (535,712,486,400 bytes)
 Total Cylinders 65,130
 Total Sectors 1,046,313,450
 Total Tracks 16,608,150
 Tracks/Cylinder 255
 Partition Disk #29, Partition #0
 Partition Size 298.92 GB (320,958,650,880 bytes)
 Partition Starting Offset 214,753,835,520 bytes
 Partition Disk #29, Partition #1
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 8,257,536 bytes
 Partition Disk #29, Partition #2
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 2,163,280,896 bytes
 Partition Disk #29, Partition #3
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 4,318,304,256 bytes
 Partition Disk #29, Partition #4
 Partition Size 24.01 GB (25,777,995,264 bytes)
 Partition Starting Offset 6,473,327,616 bytes
 Partition Disk #29, Partition #5
 Partition Size 14.00 GB (15,035,779,584 bytes)
 Partition Starting Offset 32,251,355,136 bytes
 Partition Disk #29, Partition #6
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 47,287,166,976 bytes
 Partition Disk #29, Partition #7
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 49,442,190,336 bytes

Partition Disk #29, Partition #8
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 53,744,011,776 bytes
 Partition Disk #29, Partition #9
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 55,899,035,136 bytes
 Partition Disk #29, Partition #10
 Partition Size 28.01 GB (30,071,591,424 bytes)
 Partition Starting Offset 58,054,058,496 bytes
 Partition Disk #29, Partition #11
 Partition Size 20.00 GB (21,476,173,824 bytes)
 Partition Starting Offset 88,125,682,176 bytes
 Partition Disk #29, Partition #12
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 109,601,888,256 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model NEC iStorage 2000 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 498.92 GB (535,712,486,400 bytes)
 Total Cylinders 65,130
 Total Sectors 1,046,313,450
 Total Tracks 16,608,150
 Tracks/Cylinder 255
 Partition Disk #16, Partition #0
 Partition Size 298.92 GB (320,958,650,880 bytes)
 Partition Starting Offset 214,753,835,520 bytes
 Partition Disk #16, Partition #1
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 8,257,536 bytes
 Partition Disk #16, Partition #2
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 2,163,280,896 bytes
 Partition Disk #16, Partition #3
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 4,318,304,256 bytes
 Partition Disk #16, Partition #4
 Partition Size 24.01 GB (25,777,995,264 bytes)
 Partition Starting Offset 6,473,327,616 bytes
 Partition Disk #16, Partition #5
 Partition Size 14.00 GB (15,035,779,584 bytes)
 Partition Starting Offset 32,251,355,136 bytes
 Partition Disk #16, Partition #6
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 47,287,166,976 bytes
 Partition Disk #16, Partition #7
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 49,442,190,336 bytes
 Partition Disk #16, Partition #8
 Partition Size 2.01 GB (2,154,991,104 bytes)

Partition Starting Offset 53,744,011,776 bytes
 Partition Disk #16, Partition #9
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 55,899,035,136 bytes
 Partition Disk #16, Partition #10
 Partition Size 28.01 GB (30,071,591,424 bytes)
 Partition Starting Offset 58,054,058,496 bytes
 Partition Disk #16, Partition #11
 Partition Size 20.00 GB (21,476,173,824 bytes)
 Partition Starting Offset 88,125,682,176 bytes
 Partition Disk #16, Partition #12
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 109,601,888,256 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model NEC iStorage 2000 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 5
 SCSI Target ID 0
 Sectors/Track 63
 Size 498.92 GB (535,712,486,400 bytes)
 Total Cylinders 65,130
 Total Sectors 1,046,313,450
 Total Tracks 16,608,150
 Tracks/Cylinder 255
 Partition Disk #17, Partition #0
 Partition Size 298.92 GB (320,958,650,880 bytes)
 Partition Starting Offset 214,753,835,520 bytes
 Partition Disk #17, Partition #1
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 8,257,536 bytes
 Partition Disk #17, Partition #2
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 2,163,280,896 bytes
 Partition Disk #17, Partition #3
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 4,318,304,256 bytes
 Partition Disk #17, Partition #4
 Partition Size 24.01 GB (25,777,995,264 bytes)
 Partition Starting Offset 6,473,327,616 bytes
 Partition Disk #17, Partition #5
 Partition Size 14.00 GB (15,035,779,584 bytes)
 Partition Starting Offset 32,251,355,136 bytes
 Partition Disk #17, Partition #6
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 47,287,166,976 bytes
 Partition Disk #17, Partition #7
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 49,442,190,336 bytes
 Partition Disk #17, Partition #8
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 53,744,011,776 bytes
 Partition Disk #17, Partition #9

Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 55,899,035,136 bytes
 Partition Disk #17, Partition #10
 Partition Size 28.01 GB (30,071,591,424 bytes)
 Partition Starting Offset 58,054,058,496 bytes
 Partition Disk #17, Partition #11
 Partition Size 20.00 GB (21,476,173,824 bytes)
 Partition Starting Offset 88,125,682,176 bytes
 Partition Disk #17, Partition #12
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 109,601,888,256 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model NEC iStorage 2000 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 1
 Sectors/Track 63
 Size 498.92 GB (535,712,486,400 bytes)
 Total Cylinders 65,130
 Total Sectors 1,046,313,450
 Total Tracks 16,608,150
 Tracks/Cylinder 255
 Partition Disk #18, Partition #0
 Partition Size 298.92 GB (320,958,650,880 bytes)
 Partition Starting Offset 214,753,835,520 bytes
 Partition Disk #18, Partition #1
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 8,257,536 bytes
 Partition Disk #18, Partition #2
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 2,163,280,896 bytes
 Partition Disk #18, Partition #3
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 4,318,304,256 bytes
 Partition Disk #18, Partition #4
 Partition Size 24.01 GB (25,777,995,264 bytes)
 Partition Starting Offset 6,473,327,616 bytes
 Partition Disk #18, Partition #5
 Partition Size 14.00 GB (15,035,779,584 bytes)
 Partition Starting Offset 32,251,355,136 bytes
 Partition Disk #18, Partition #6
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 47,287,166,976 bytes
 Partition Disk #18, Partition #7
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 49,442,190,336 bytes
 Partition Disk #18, Partition #8
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 53,744,011,776 bytes
 Partition Disk #18, Partition #9
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 55,899,035,136 bytes

Partition Disk #18, Partition #10
 Partition Size 28.01 GB (30,071,591,424 bytes)
 Partition Starting Offset 58,054,058,496 bytes
 Partition Disk #18, Partition #11
 Partition Size 20.00 GB (21,476,173,824 bytes)
 Partition Starting Offset 88,125,682,176 bytes
 Partition Disk #18, Partition #12
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 109,601,888,256 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model NEC iStorage 2000 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 5
 SCSI Target ID 1
 Sectors/Track 63
 Size 498.92 GB (535,712,486,400 bytes)
 Total Cylinders 65,130
 Total Sectors 1,046,313,450
 Total Tracks 16,608,150
 Tracks/Cylinder 255
 Partition Disk #19, Partition #0
 Partition Size 298.92 GB (320,958,650,880 bytes)
 Partition Starting Offset 214,753,835,520 bytes
 Partition Disk #19, Partition #1
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 8,257,536 bytes
 Partition Disk #19, Partition #2
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 2,163,280,896 bytes
 Partition Disk #19, Partition #3
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 4,318,304,256 bytes
 Partition Disk #19, Partition #4
 Partition Size 24.01 GB (25,777,995,264 bytes)
 Partition Starting Offset 6,473,327,616 bytes
 Partition Disk #19, Partition #5
 Partition Size 14.00 GB (15,035,779,584 bytes)
 Partition Starting Offset 32,251,355,136 bytes
 Partition Disk #19, Partition #6
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 47,287,166,976 bytes
 Partition Disk #19, Partition #7
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 49,442,190,336 bytes
 Partition Disk #19, Partition #8
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 53,744,011,776 bytes
 Partition Disk #19, Partition #9
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 55,899,035,136 bytes
 Partition Disk #19, Partition #10
 Partition Size 28.01 GB (30,071,591,424 bytes)

Partition Starting Offset 58,054,058,496 bytes
 Partition Disk #19, Partition #11
 Partition Size 20.00 GB (21,476,173,824 bytes)
 Partition Starting Offset 88,125,682,176 bytes
 Partition Disk #19, Partition #12
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 109,601,888,256 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model NEC iStorage 2000 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 4
 SCSI Target ID 0
 Sectors/Track 63
 Size 498.92 GB (535,712,486,400 bytes)
 Total Cylinders 65,130
 Total Sectors 1,046,313,450
 Total Tracks 16,608,150
 Tracks/Cylinder 255
 Partition Disk #12, Partition #0
 Partition Size 298.92 GB (320,958,650,880 bytes)
 Partition Starting Offset 214,753,835,520 bytes
 Partition Disk #12, Partition #1
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 8,257,536 bytes
 Partition Disk #12, Partition #2
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 2,163,280,896 bytes
 Partition Disk #12, Partition #3
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 4,318,304,256 bytes
 Partition Disk #12, Partition #4
 Partition Size 24.01 GB (25,777,995,264 bytes)
 Partition Starting Offset 6,473,327,616 bytes
 Partition Disk #12, Partition #5
 Partition Size 14.00 GB (15,035,779,584 bytes)
 Partition Starting Offset 32,251,355,136 bytes
 Partition Disk #12, Partition #6
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 47,287,166,976 bytes
 Partition Disk #12, Partition #7
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 49,442,190,336 bytes
 Partition Disk #12, Partition #8
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 53,744,011,776 bytes
 Partition Disk #12, Partition #9
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 55,899,035,136 bytes
 Partition Disk #12, Partition #10
 Partition Size 28.01 GB (30,071,591,424 bytes)
 Partition Starting Offset 58,054,058,496 bytes
 Partition Disk #12, Partition #11

Partition Size 20.00 GB (21,476,173,824 bytes)
 Partition Starting Offset 88,125,682,176 bytes
 Partition Disk #12, Partition #12
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 109,601,888,256 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model NEC iStorage 2000 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 498.92 GB (535,712,486,400 bytes)
 Total Cylinders 65,130
 Total Sectors 1,046,313,450
 Total Tracks 16,608,150
 Tracks/Cylinder 255
 Partition Disk #13, Partition #0
 Partition Size 298.92 GB (320,958,650,880 bytes)
 Partition Starting Offset 214,753,835,520 bytes
 Partition Disk #13, Partition #1
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 8,257,536 bytes
 Partition Disk #13, Partition #2
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 2,163,280,896 bytes
 Partition Disk #13, Partition #3
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 4,318,304,256 bytes
 Partition Disk #13, Partition #4
 Partition Size 24.01 GB (25,777,995,264 bytes)
 Partition Starting Offset 6,473,327,616 bytes
 Partition Disk #13, Partition #5
 Partition Size 14.00 GB (15,035,779,584 bytes)
 Partition Starting Offset 32,251,355,136 bytes
 Partition Disk #13, Partition #6
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 47,287,166,976 bytes
 Partition Disk #13, Partition #7
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 49,442,190,336 bytes
 Partition Disk #13, Partition #8
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 53,744,011,776 bytes
 Partition Disk #13, Partition #9
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 55,899,035,136 bytes
 Partition Disk #13, Partition #10
 Partition Size 28.01 GB (30,071,591,424 bytes)
 Partition Starting Offset 58,054,058,496 bytes
 Partition Disk #13, Partition #11
 Partition Size 20.00 GB (21,476,173,824 bytes)
 Partition Starting Offset 88,125,682,176 bytes

Partition Disk #13, Partition #12
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 109,601,888,256 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model NEC iStorage 2000 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 4
 SCSI Target ID 1
 Sectors/Track 63
 Size 498.92 GB (535,712,486,400 bytes)
 Total Cylinders 65,130
 Total Sectors 1,046,313,450
 Total Tracks 16,608,150
 Tracks/Cylinder 255
 Partition Disk #14, Partition #0
 Partition Size 298.92 GB (320,958,650,880 bytes)
 Partition Starting Offset 214,753,835,520 bytes
 Partition Disk #14, Partition #1
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 8,257,536 bytes
 Partition Disk #14, Partition #2
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 2,163,280,896 bytes
 Partition Disk #14, Partition #3
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 4,318,304,256 bytes
 Partition Disk #14, Partition #4
 Partition Size 24.01 GB (25,777,995,264 bytes)
 Partition Starting Offset 6,473,327,616 bytes
 Partition Disk #14, Partition #5
 Partition Size 14.00 GB (15,035,779,584 bytes)
 Partition Starting Offset 32,251,355,136 bytes
 Partition Disk #14, Partition #6
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 47,287,166,976 bytes
 Partition Disk #14, Partition #7
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 49,442,190,336 bytes
 Partition Disk #14, Partition #8
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 53,744,011,776 bytes
 Partition Disk #14, Partition #9
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 55,899,035,136 bytes
 Partition Disk #14, Partition #10
 Partition Size 28.01 GB (30,071,591,424 bytes)
 Partition Starting Offset 58,054,058,496 bytes
 Partition Disk #14, Partition #11
 Partition Size 20.00 GB (21,476,173,824 bytes)
 Partition Starting Offset 88,125,682,176 bytes
 Partition Disk #14, Partition #12
 Partition Size 4.01 GB (4,301,789,184 bytes)

Partition Starting Offset 109,601,888,256 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model NEC iStorage 2000 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 1
 Sectors/Track 63
 Size 498.92 GB (535,712,486,400 bytes)
 Total Cylinders 65,130
 Total Sectors 1,046,313,450
 Total Tracks 16,608,150
 Tracks/Cylinder 255
 Partition Disk #15, Partition #0
 Partition Size 298.92 GB (320,958,650,880 bytes)
 Partition Starting Offset 214,753,835,520 bytes
 Partition Disk #15, Partition #1
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 8,257,536 bytes
 Partition Disk #15, Partition #2
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 2,163,280,896 bytes
 Partition Disk #15, Partition #3
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 4,318,304,256 bytes
 Partition Disk #15, Partition #4
 Partition Size 24.01 GB (25,777,995,264 bytes)
 Partition Starting Offset 6,473,327,616 bytes
 Partition Disk #15, Partition #5
 Partition Size 14.00 GB (15,035,779,584 bytes)
 Partition Starting Offset 32,251,355,136 bytes
 Partition Disk #15, Partition #6
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 47,287,166,976 bytes
 Partition Disk #15, Partition #7
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 49,442,190,336 bytes
 Partition Disk #15, Partition #8
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 53,744,011,776 bytes
 Partition Disk #15, Partition #9
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 55,899,035,136 bytes
 Partition Disk #15, Partition #10
 Partition Size 28.01 GB (30,071,591,424 bytes)
 Partition Starting Offset 58,054,058,496 bytes
 Partition Disk #15, Partition #11
 Partition Size 20.00 GB (21,476,173,824 bytes)
 Partition Starting Offset 88,125,682,176 bytes
 Partition Disk #15, Partition #12
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 109,601,888,256 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model NEC iStorage 2000 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 3
 SCSI Target ID 0
 Sectors/Track 63
 Size 498.92 GB (535,712,486,400 bytes)
 Total Cylinders 65,130
 Total Sectors 1,046,313,450
 Total Tracks 16,608,150
 Tracks/Cylinder 255
 Partition Disk #8, Partition #0
 Partition Size 298.92 GB (320,958,650,880 bytes)
 Partition Starting Offset 214,753,835,520 bytes
 Partition Disk #8, Partition #1
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 8,257,536 bytes
 Partition Disk #8, Partition #2
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 2,163,280,896 bytes
 Partition Disk #8, Partition #3
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 4,318,304,256 bytes
 Partition Disk #8, Partition #4
 Partition Size 24.01 GB (25,777,995,264 bytes)
 Partition Starting Offset 6,473,327,616 bytes
 Partition Disk #8, Partition #5
 Partition Size 14.00 GB (15,035,779,584 bytes)
 Partition Starting Offset 32,251,355,136 bytes
 Partition Disk #8, Partition #6
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 47,287,166,976 bytes
 Partition Disk #8, Partition #7
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 49,442,190,336 bytes
 Partition Disk #8, Partition #8
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 53,744,011,776 bytes
 Partition Disk #8, Partition #9
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 55,899,035,136 bytes
 Partition Disk #8, Partition #10
 Partition Size 28.01 GB (30,071,591,424 bytes)
 Partition Starting Offset 58,054,058,496 bytes
 Partition Disk #8, Partition #11
 Partition Size 20.00 GB (21,476,173,824 bytes)
 Partition Starting Offset 88,125,682,176 bytes
 Partition Disk #8, Partition #12
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 109,601,888,256 bytes

Description Disk drive
 Manufacturer (Standard disk drives)

Model NEC iStorage 2000 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 498.92 GB (535,712,486,400 bytes)
 Total Cylinders 65,130
 Total Sectors 1,046,313,450
 Total Tracks 16,608,150
 Tracks/Cylinder 255
 Partition Disk #9, Partition #0
 Partition Size 298.92 GB (320,958,650,880 bytes)
 Partition Starting Offset 214,753,835,520 bytes
 Partition Disk #9, Partition #1
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 8,257,536 bytes
 Partition Disk #9, Partition #2
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 2,163,280,896 bytes
 Partition Disk #9, Partition #3
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 4,318,304,256 bytes
 Partition Disk #9, Partition #4
 Partition Size 24.01 GB (25,777,995,264 bytes)
 Partition Starting Offset 6,473,327,616 bytes
 Partition Disk #9, Partition #5
 Partition Size 14.00 GB (15,035,779,584 bytes)
 Partition Starting Offset 32,251,355,136 bytes
 Partition Disk #9, Partition #6
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 47,287,166,976 bytes
 Partition Disk #9, Partition #7
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 49,442,190,336 bytes
 Partition Disk #9, Partition #8
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 53,744,011,776 bytes
 Partition Disk #9, Partition #9
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 55,899,035,136 bytes
 Partition Disk #9, Partition #10
 Partition Size 28.01 GB (30,071,591,424 bytes)
 Partition Starting Offset 58,054,058,496 bytes
 Partition Disk #9, Partition #11
 Partition Size 20.00 GB (21,476,173,824 bytes)
 Partition Starting Offset 88,125,682,176 bytes
 Partition Disk #9, Partition #12
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 109,601,888,256 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model NEC iStorage 2000 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         3
SCSI Target ID 1
Sectors/Track    63
Size             498.92 GB (535,712,486,400 bytes)
Total Cylinders   65,130
Total Sectors     1,046,313,450
Total Tracks      16,608,150
Tracks/Cylinder   255
Partition         Disk #10, Partition #0
Partition Size    298.92 GB (320,958,650,880 bytes)
Partition Starting Offset 214,753,835,520 bytes
Partition         Disk #10, Partition #1
Partition Size    2.01 GB (2,154,991,104 bytes)
Partition Starting Offset 8,257,536 bytes
Partition         Disk #10, Partition #2
Partition Size    2.01 GB (2,154,991,104 bytes)
Partition Starting Offset 2,163,280,896 bytes
Partition         Disk #10, Partition #3
Partition Size    2.01 GB (2,154,991,104 bytes)
Partition Starting Offset 4,318,304,256 bytes
Partition         Disk #10, Partition #4
Partition Size    24.01 GB (25,777,995,264 bytes)
Partition Starting Offset 6,473,327,616 bytes
Partition         Disk #10, Partition #5
Partition Size    14.00 GB (15,035,779,584 bytes)
Partition Starting Offset 32,251,355,136 bytes
Partition         Disk #10, Partition #6
Partition Size    2.01 GB (2,154,991,104 bytes)
Partition Starting Offset 47,287,166,976 bytes
Partition         Disk #10, Partition #7
Partition Size    4.01 GB (4,301,789,184 bytes)
Partition Starting Offset 49,442,190,336 bytes
Partition         Disk #10, Partition #8
Partition Size    2.01 GB (2,154,991,104 bytes)
Partition Starting Offset 53,744,011,776 bytes
Partition         Disk #10, Partition #9
Partition Size    2.01 GB (2,154,991,104 bytes)
Partition Starting Offset 55,899,035,136 bytes
Partition         Disk #10, Partition #10
Partition Size    28.01 GB (30,071,591,424 bytes)
Partition Starting Offset 58,054,058,496 bytes
Partition         Disk #10, Partition #11
Partition Size    20.00 GB (21,476,173,824 bytes)
Partition Starting Offset 88,125,682,176 bytes
Partition         Disk #10, Partition #12
Partition Size    4.01 GB (4,301,789,184 bytes)
Partition Starting Offset 109,601,888,256 bytes

Description       Disk drive
Manufacturer       (Standard disk drives)
Model              NEC iStorage 2000 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 1
Sectors/Track    63
Size             498.92 GB (535,712,486,400 bytes)
Total Cylinders   65,130
Total Sectors     1,046,313,450
Total Tracks      16,608,150
Tracks/Cylinder   255
Partition         Disk #11, Partition #0
Partition Size    298.92 GB (320,958,650,880 bytes)
Partition Starting Offset 214,753,835,520 bytes
Partition         Disk #11, Partition #1
Partition Size    2.01 GB (2,154,991,104 bytes)
Partition Starting Offset 8,257,536 bytes
Partition         Disk #11, Partition #2
Partition Size    2.01 GB (2,154,991,104 bytes)
Partition Starting Offset 2,163,280,896 bytes
Partition         Disk #11, Partition #3
Partition Size    2.01 GB (2,154,991,104 bytes)
Partition Starting Offset 4,318,304,256 bytes
Partition         Disk #11, Partition #4
Partition Size    24.01 GB (25,777,995,264 bytes)
Partition Starting Offset 6,473,327,616 bytes
Partition         Disk #11, Partition #5
Partition Size    14.00 GB (15,035,779,584 bytes)
Partition Starting Offset 32,251,355,136 bytes
Partition         Disk #11, Partition #6
Partition Size    2.01 GB (2,154,991,104 bytes)
Partition Starting Offset 47,287,166,976 bytes
Partition         Disk #11, Partition #7
Partition Size    4.01 GB (4,301,789,184 bytes)
Partition Starting Offset 49,442,190,336 bytes
Partition         Disk #11, Partition #8
Partition Size    2.01 GB (2,154,991,104 bytes)
Partition Starting Offset 53,744,011,776 bytes
Partition         Disk #11, Partition #9
Partition Size    2.01 GB (2,154,991,104 bytes)
Partition Starting Offset 55,899,035,136 bytes
Partition         Disk #11, Partition #10
Partition Size    28.01 GB (30,071,591,424 bytes)
Partition Starting Offset 58,054,058,496 bytes
Partition         Disk #11, Partition #11
Partition Size    20.00 GB (21,476,173,824 bytes)
Partition Starting Offset 88,125,682,176 bytes
Partition         Disk #11, Partition #12
Partition Size    4.01 GB (4,301,789,184 bytes)
Partition Starting Offset 109,601,888,256 bytes

Description       Disk drive
Manufacturer       (Standard disk drives)
Model              NEC iStorage 2000 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              1
SCSI Target ID        0
Sectors/Track         63
Size      498.92 GB (535,712,486,400 bytes)
Total Cylinders       65,130
Total Sectors      1,046,313,450
Total Tracks       16,608,150
Tracks/Cylinder      255
Partition             Disk #0, Partition #0
Partition Size 298.92 GB (320,958,650,880 bytes)
Partition Starting Offset      214,753,835,520 bytes
Partition             Disk #0, Partition #1
Partition Size 2.01 GB (2,154,991,104 bytes)
Partition Starting Offset      8,257,536 bytes
Partition             Disk #0, Partition #2
Partition Size 2.01 GB (2,154,991,104 bytes)
Partition Starting Offset      2,163,280,896 bytes
Partition             Disk #0, Partition #3
Partition Size 2.01 GB (2,154,991,104 bytes)
Partition Starting Offset      4,318,304,256 bytes
Partition             Disk #0, Partition #4
Partition Size 24.01 GB (25,777,995,264 bytes)
Partition Starting Offset      6,473,327,616 bytes
Partition             Disk #0, Partition #5
Partition Size 14.00 GB (15,035,779,584 bytes)
Partition Starting Offset      32,251,355,136 bytes
Partition             Disk #0, Partition #6
Partition Size 2.01 GB (2,154,991,104 bytes)
Partition Starting Offset      47,287,166,976 bytes
Partition             Disk #0, Partition #7
Partition Size 4.01 GB (4,301,789,184 bytes)
Partition Starting Offset      49,442,190,336 bytes
Partition             Disk #0, Partition #8
Partition Size 2.01 GB (2,154,991,104 bytes)
Partition Starting Offset      53,744,011,776 bytes
Partition             Disk #0, Partition #9
Partition Size 2.01 GB (2,154,991,104 bytes)
Partition Starting Offset      55,899,035,136 bytes
Partition             Disk #0, Partition #10
Partition Size 28.01 GB (30,071,591,424 bytes)
Partition Starting Offset      58,054,058,496 bytes
Partition             Disk #0, Partition #11
Partition Size 20.00 GB (21,476,173,824 bytes)
Partition Starting Offset      88,125,682,176 bytes
Partition             Disk #0, Partition #12
Partition Size 4.01 GB (4,301,789,184 bytes)
Partition Starting Offset      109,601,888,256 bytes

Description      Disk drive
Manufacturer      (Standard disk drives)
Model      NEC iStorage 2000 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          1

```

```

SCSI Target ID 0
Sectors/Track 63
Size      498.92 GB (535,712,486,400 bytes)
Total Cylinders       65,130
Total Sectors      1,046,313,450
Total Tracks       16,608,150
Tracks/Cylinder      255
Partition             Disk #1, Partition #0
Partition Size 298.92 GB (320,958,650,880 bytes)
Partition Starting Offset      214,753,835,520 bytes
Partition             Disk #1, Partition #1
Partition Size 2.01 GB (2,154,991,104 bytes)
Partition Starting Offset      8,257,536 bytes
Partition             Disk #1, Partition #2
Partition Size 2.01 GB (2,154,991,104 bytes)
Partition Starting Offset      2,163,280,896 bytes
Partition             Disk #1, Partition #3
Partition Size 2.01 GB (2,154,991,104 bytes)
Partition Starting Offset      4,318,304,256 bytes
Partition             Disk #1, Partition #4
Partition Size 24.01 GB (25,777,995,264 bytes)
Partition Starting Offset      6,473,327,616 bytes
Partition             Disk #1, Partition #5
Partition Size 14.00 GB (15,035,779,584 bytes)
Partition Starting Offset      32,251,355,136 bytes
Partition             Disk #1, Partition #6
Partition Size 2.01 GB (2,154,991,104 bytes)
Partition Starting Offset      47,287,166,976 bytes
Partition             Disk #1, Partition #7
Partition Size 4.01 GB (4,301,789,184 bytes)
Partition Starting Offset      49,442,190,336 bytes
Partition             Disk #1, Partition #8
Partition Size 2.01 GB (2,154,991,104 bytes)
Partition Starting Offset      53,744,011,776 bytes
Partition             Disk #1, Partition #9
Partition Size 2.01 GB (2,154,991,104 bytes)
Partition Starting Offset      55,899,035,136 bytes
Partition             Disk #1, Partition #10
Partition Size 28.01 GB (30,071,591,424 bytes)
Partition Starting Offset      58,054,058,496 bytes
Partition             Disk #1, Partition #11
Partition Size 20.00 GB (21,476,173,824 bytes)
Partition Starting Offset      88,125,682,176 bytes
Partition             Disk #1, Partition #12
Partition Size 4.01 GB (4,301,789,184 bytes)
Partition Starting Offset      109,601,888,256 bytes

Description      Disk drive
Manufacturer      (Standard disk drives)
Model      NEC iStorage 2000 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          1
SCSI Target ID      1
Sectors/Track      63

```

Size 498.92 GB (535,712,486,400 bytes)
 Total Cylinders 65,130
 Total Sectors 1,046,313,450
 Total Tracks 16,608,150
 Tracks/Cylinder 255
 Partition Disk #2, Partition #0
 Partition Size 298.92 GB (320,958,650,880 bytes)
 Partition Starting Offset 214,753,835,520 bytes
 Partition Disk #2, Partition #1
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 8,257,536 bytes
 Partition Disk #2, Partition #2
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 2,163,280,896 bytes
 Partition Disk #2, Partition #3
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 4,318,304,256 bytes
 Partition Disk #2, Partition #4
 Partition Size 24.01 GB (25,777,995,264 bytes)
 Partition Starting Offset 6,473,327,616 bytes
 Partition Disk #2, Partition #5
 Partition Size 14.00 GB (15,035,779,584 bytes)
 Partition Starting Offset 32,251,355,136 bytes
 Partition Disk #2, Partition #6
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 47,287,166,976 bytes
 Partition Disk #2, Partition #7
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 49,442,190,336 bytes
 Partition Disk #2, Partition #8
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 53,744,011,776 bytes
 Partition Disk #2, Partition #9
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 55,899,035,136 bytes
 Partition Disk #2, Partition #10
 Partition Size 28.01 GB (30,071,591,424 bytes)
 Partition Starting Offset 58,054,058,496 bytes
 Partition Disk #2, Partition #11
 Partition Size 20.00 GB (21,476,173,824 bytes)
 Partition Starting Offset 88,125,682,176 bytes
 Partition Disk #2, Partition #12
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 109,601,888,256 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model NEC iStorage 2000 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 1
 SCSI Target ID 1
 Sectors/Track 63
 Size 498.92 GB (535,712,486,400 bytes)
 Total Cylinders 65,130

Total Sectors 1,046,313,450
 Total Tracks 16,608,150
 Tracks/Cylinder 255
 Partition Disk #3, Partition #0
 Partition Size 298.92 GB (320,958,650,880 bytes)
 Partition Starting Offset 214,753,835,520 bytes
 Partition Disk #3, Partition #1
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 8,257,536 bytes
 Partition Disk #3, Partition #2
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 2,163,280,896 bytes
 Partition Disk #3, Partition #3
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 4,318,304,256 bytes
 Partition Disk #3, Partition #4
 Partition Size 24.01 GB (25,777,995,264 bytes)
 Partition Starting Offset 6,473,327,616 bytes
 Partition Disk #3, Partition #5
 Partition Size 14.00 GB (15,035,779,584 bytes)
 Partition Starting Offset 32,251,355,136 bytes
 Partition Disk #3, Partition #6
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 47,287,166,976 bytes
 Partition Disk #3, Partition #7
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 49,442,190,336 bytes
 Partition Disk #3, Partition #8
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 53,744,011,776 bytes
 Partition Disk #3, Partition #9
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 55,899,035,136 bytes
 Partition Disk #3, Partition #10
 Partition Size 28.01 GB (30,071,591,424 bytes)
 Partition Starting Offset 58,054,058,496 bytes
 Partition Disk #3, Partition #11
 Partition Size 20.00 GB (21,476,173,824 bytes)
 Partition Starting Offset 88,125,682,176 bytes
 Partition Disk #3, Partition #12
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 109,601,888,256 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model NEC iStorage 2000 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 498.92 GB (535,712,486,400 bytes)
 Total Cylinders 65,130
 Total Sectors 1,046,313,450
 Total Tracks 16,608,150

```

Tracks/Cylinder      255
Partition            Disk #20, Partition #0
Partition Size 298.92 GB (320,958,650,880 bytes)
Partition Starting Offset      214,753,835,520 bytes
Partition            Disk #20, Partition #1
Partition Size 2.01 GB (2,154,991,104 bytes)
Partition Starting Offset      8,257,536 bytes
Partition            Disk #20, Partition #2
Partition Size 2.01 GB (2,154,991,104 bytes)
Partition Starting Offset      2,163,280,896 bytes
Partition            Disk #20, Partition #3
Partition Size 2.01 GB (2,154,991,104 bytes)
Partition Starting Offset      4,318,304,256 bytes
Partition            Disk #20, Partition #4
Partition Size 24.01 GB (25,777,995,264 bytes)
Partition Starting Offset      6,473,327,616 bytes
Partition            Disk #20, Partition #5
Partition Size 14.00 GB (15,035,779,584 bytes)
Partition Starting Offset      32,251,355,136 bytes
Partition            Disk #20, Partition #6
Partition Size 2.01 GB (2,154,991,104 bytes)
Partition Starting Offset      47,287,166,976 bytes
Partition            Disk #20, Partition #7
Partition Size 4.01 GB (4,301,789,184 bytes)
Partition Starting Offset      49,442,190,336 bytes
Partition            Disk #20, Partition #8
Partition Size 2.01 GB (2,154,991,104 bytes)
Partition Starting Offset      53,744,011,776 bytes
Partition            Disk #20, Partition #9
Partition Size 2.01 GB (2,154,991,104 bytes)
Partition Starting Offset      55,899,035,136 bytes
Partition            Disk #20, Partition #10
Partition Size 28.01 GB (30,071,591,424 bytes)
Partition Starting Offset      58,054,058,496 bytes
Partition            Disk #20, Partition #11
Partition Size 20.00 GB (21,476,173,824 bytes)
Partition Starting Offset      88,125,682,176 bytes
Partition            Disk #20, Partition #12
Partition Size 4.01 GB (4,301,789,184 bytes)
Partition Starting Offset      109,601,888,256 bytes

```

```

Description      Disk drive
Manufacturer      (Standard disk drives)
Model      NEC iStorage 2000 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      6
SCSI Target ID      0
Sectors/Track      63
Size      498.92 GB (535,712,486,400 bytes)
Total Cylinders      65,130
Total Sectors      1,046,313,450
Total Tracks      16,608,150
Tracks/Cylinder      255
Partition            Disk #21, Partition #0

```

```

Partition Size 298.92 GB (320,958,650,880 bytes)
Partition Starting Offset      214,753,835,520 bytes
Partition            Disk #21, Partition #1
Partition Size 2.01 GB (2,154,991,104 bytes)
Partition Starting Offset      8,257,536 bytes
Partition            Disk #21, Partition #2
Partition Size 2.01 GB (2,154,991,104 bytes)
Partition Starting Offset      2,163,280,896 bytes
Partition            Disk #21, Partition #3
Partition Size 2.01 GB (2,154,991,104 bytes)
Partition Starting Offset      4,318,304,256 bytes
Partition            Disk #21, Partition #4
Partition Size 24.01 GB (25,777,995,264 bytes)
Partition Starting Offset      6,473,327,616 bytes
Partition            Disk #21, Partition #5
Partition Size 14.00 GB (15,035,779,584 bytes)
Partition Starting Offset      32,251,355,136 bytes
Partition            Disk #21, Partition #6
Partition Size 2.01 GB (2,154,991,104 bytes)
Partition Starting Offset      47,287,166,976 bytes
Partition            Disk #21, Partition #7
Partition Size 4.01 GB (4,301,789,184 bytes)
Partition Starting Offset      49,442,190,336 bytes
Partition            Disk #21, Partition #8
Partition Size 2.01 GB (2,154,991,104 bytes)
Partition Starting Offset      53,744,011,776 bytes
Partition            Disk #21, Partition #9
Partition Size 2.01 GB (2,154,991,104 bytes)
Partition Starting Offset      55,899,035,136 bytes
Partition            Disk #21, Partition #10
Partition Size 28.01 GB (30,071,591,424 bytes)
Partition Starting Offset      58,054,058,496 bytes
Partition            Disk #21, Partition #11
Partition Size 20.00 GB (21,476,173,824 bytes)
Partition Starting Offset      88,125,682,176 bytes
Partition            Disk #21, Partition #12
Partition Size 4.01 GB (4,301,789,184 bytes)
Partition Starting Offset      109,601,888,256 bytes

```

```

Description      Disk drive
Manufacturer      (Standard disk drives)
Model      NEC iStorage 2000 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      1
Sectors/Track      63
Size      498.92 GB (535,712,486,400 bytes)
Total Cylinders      65,130
Total Sectors      1,046,313,450
Total Tracks      16,608,150
Tracks/Cylinder      255
Partition            Disk #22, Partition #0
Partition Size 298.92 GB (320,958,650,880 bytes)
Partition Starting Offset      214,753,835,520 bytes

```

Partition Disk #22, Partition #1
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 8,257,536 bytes
 Partition Disk #22, Partition #2
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 2,163,280,896 bytes
 Partition Disk #22, Partition #3
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 4,318,304,256 bytes
 Partition Disk #22, Partition #4
 Partition Size 24.01 GB (25,777,995,264 bytes)
 Partition Starting Offset 6,473,327,616 bytes
 Partition Disk #22, Partition #5
 Partition Size 14.00 GB (15,035,779,584 bytes)
 Partition Starting Offset 32,251,355,136 bytes
 Partition Disk #22, Partition #6
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 47,287,166,976 bytes
 Partition Disk #22, Partition #7
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 49,442,190,336 bytes
 Partition Disk #22, Partition #8
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 53,744,011,776 bytes
 Partition Disk #22, Partition #9
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 55,899,035,136 bytes
 Partition Disk #22, Partition #10
 Partition Size 28.01 GB (30,071,591,424 bytes)
 Partition Starting Offset 58,054,058,496 bytes
 Partition Disk #22, Partition #11
 Partition Size 20.00 GB (21,476,173,824 bytes)
 Partition Starting Offset 88,125,682,176 bytes
 Partition Disk #22, Partition #12
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 109,601,888,256 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model NEC iStorage 2000 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 6
 SCSI Target ID 1
 Sectors/Track 63
 Size 498.92 GB (535,712,486,400 bytes)
 Total Cylinders 65,130
 Total Sectors 1,046,313,450
 Total Tracks 16,608,150
 Tracks/Cylinder 255
 Partition Disk #23, Partition #0
 Partition Size 298.92 GB (320,958,650,880 bytes)
 Partition Starting Offset 214,753,835,520 bytes
 Partition Disk #23, Partition #1
 Partition Size 2.01 GB (2,154,991,104 bytes)

Partition Starting Offset 8,257,536 bytes
 Partition Disk #23, Partition #2
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 2,163,280,896 bytes
 Partition Disk #23, Partition #3
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 4,318,304,256 bytes
 Partition Disk #23, Partition #4
 Partition Size 24.01 GB (25,777,995,264 bytes)
 Partition Starting Offset 6,473,327,616 bytes
 Partition Disk #23, Partition #5
 Partition Size 14.00 GB (15,035,779,584 bytes)
 Partition Starting Offset 32,251,355,136 bytes
 Partition Disk #23, Partition #6
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 47,287,166,976 bytes
 Partition Disk #23, Partition #7
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 49,442,190,336 bytes
 Partition Disk #23, Partition #8
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 53,744,011,776 bytes
 Partition Disk #23, Partition #9
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 55,899,035,136 bytes
 Partition Disk #23, Partition #10
 Partition Size 28.01 GB (30,071,591,424 bytes)
 Partition Starting Offset 58,054,058,496 bytes
 Partition Disk #23, Partition #11
 Partition Size 20.00 GB (21,476,173,824 bytes)
 Partition Starting Offset 88,125,682,176 bytes
 Partition Disk #23, Partition #12
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 109,601,888,256 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model NEC iStorage 2000 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 10
 SCSI Target ID 0
 Sectors/Track 63
 Size 498.92 GB (535,712,486,400 bytes)
 Total Cylinders 65,130
 Total Sectors 1,046,313,450
 Total Tracks 16,608,150
 Tracks/Cylinder 255
 Partition Disk #34, Partition #0
 Partition Size 298.92 GB (320,958,650,880 bytes)
 Partition Starting Offset 214,753,835,520 bytes
 Partition Disk #34, Partition #1
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 8,257,536 bytes
 Partition Disk #34, Partition #2

Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 2,163,280,896 bytes
 Partition Disk #34, Partition #3
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 4,318,304,256 bytes
 Partition Disk #34, Partition #4
 Partition Size 24.01 GB (25,777,995,264 bytes)
 Partition Starting Offset 6,473,327,616 bytes
 Partition Disk #34, Partition #5
 Partition Size 14.00 GB (15,035,779,584 bytes)
 Partition Starting Offset 32,251,355,136 bytes
 Partition Disk #34, Partition #6
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 47,287,166,976 bytes
 Partition Disk #34, Partition #7
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 49,442,190,336 bytes
 Partition Disk #34, Partition #8
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 53,744,011,776 bytes
 Partition Disk #34, Partition #9
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 55,899,035,136 bytes
 Partition Disk #34, Partition #10
 Partition Size 28.01 GB (30,071,591,424 bytes)
 Partition Starting Offset 58,054,058,496 bytes
 Partition Disk #34, Partition #11
 Partition Size 20.00 GB (21,476,173,824 bytes)
 Partition Starting Offset 88,125,682,176 bytes
 Partition Disk #34, Partition #12
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 109,601,888,256 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model NEC iStorage 2000 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 10
 SCSI Target ID 0
 Sectors/Track 63
 Size 498.92 GB (535,712,486,400 bytes)
 Total Cylinders 65,130
 Total Sectors 1,046,313,450
 Total Tracks 16,608,150
 Tracks/Cylinder 255
 Partition Disk #35, Partition #0
 Partition Size 298.92 GB (320,958,650,880 bytes)
 Partition Starting Offset 214,753,835,520 bytes
 Partition Disk #35, Partition #1
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 8,257,536 bytes
 Partition Disk #35, Partition #2
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 2,163,280,896 bytes

Partition Disk #35, Partition #3
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 4,318,304,256 bytes
 Partition Disk #35, Partition #4
 Partition Size 24.01 GB (25,777,995,264 bytes)
 Partition Starting Offset 6,473,327,616 bytes
 Partition Disk #35, Partition #5
 Partition Size 14.00 GB (15,035,779,584 bytes)
 Partition Starting Offset 32,251,355,136 bytes
 Partition Disk #35, Partition #6
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 47,287,166,976 bytes
 Partition Disk #35, Partition #7
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 49,442,190,336 bytes
 Partition Disk #35, Partition #8
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 53,744,011,776 bytes
 Partition Disk #35, Partition #9
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 55,899,035,136 bytes
 Partition Disk #35, Partition #10
 Partition Size 28.01 GB (30,071,591,424 bytes)
 Partition Starting Offset 58,054,058,496 bytes
 Partition Disk #35, Partition #11
 Partition Size 20.00 GB (21,476,173,824 bytes)
 Partition Starting Offset 88,125,682,176 bytes
 Partition Disk #35, Partition #12
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 109,601,888,256 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model NEC iStorage 2000 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 10
 SCSI Target ID 1
 Sectors/Track 63
 Size 498.92 GB (535,712,486,400 bytes)
 Total Cylinders 65,130
 Total Sectors 1,046,313,450
 Total Tracks 16,608,150
 Tracks/Cylinder 255
 Partition Disk #36, Partition #0
 Partition Size 298.92 GB (320,958,650,880 bytes)
 Partition Starting Offset 214,753,835,520 bytes
 Partition Disk #36, Partition #1
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 8,257,536 bytes
 Partition Disk #36, Partition #2
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 2,163,280,896 bytes
 Partition Disk #36, Partition #3
 Partition Size 2.01 GB (2,154,991,104 bytes)

Partition Starting Offset 4,318,304,256 bytes
 Partition Disk #36, Partition #4
 Partition Size 24.01 GB (25,777,995,264 bytes)
 Partition Starting Offset 6,473,327,616 bytes
 Partition Disk #36, Partition #5
 Partition Size 14.00 GB (15,035,779,584 bytes)
 Partition Starting Offset 32,251,355,136 bytes
 Partition Disk #36, Partition #6
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 47,287,166,976 bytes
 Partition Disk #36, Partition #7
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 49,442,190,336 bytes
 Partition Disk #36, Partition #8
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 53,744,011,776 bytes
 Partition Disk #36, Partition #9
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 55,899,035,136 bytes
 Partition Disk #36, Partition #10
 Partition Size 28.01 GB (30,071,591,424 bytes)
 Partition Starting Offset 58,054,058,496 bytes
 Partition Disk #36, Partition #11
 Partition Size 20.00 GB (21,476,173,824 bytes)
 Partition Starting Offset 88,125,682,176 bytes
 Partition Disk #36, Partition #12
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 109,601,888,256 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model NEC iStorage 2000 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 10
 SCSI Target ID 1
 Sectors/Track 63
 Size 498.92 GB (535,712,486,400 bytes)
 Total Cylinders 65,130
 Total Sectors 1,046,313,450
 Total Tracks 16,608,150
 Tracks/Cylinder 255
 Partition Disk #37, Partition #0
 Partition Size 298.92 GB (320,958,650,880 bytes)
 Partition Starting Offset 214,753,835,520 bytes
 Partition Disk #37, Partition #1
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 8,257,536 bytes
 Partition Disk #37, Partition #2
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 2,163,280,896 bytes
 Partition Disk #37, Partition #3
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 4,318,304,256 bytes
 Partition Disk #37, Partition #4

Partition Size 24.01 GB (25,777,995,264 bytes)
 Partition Starting Offset 6,473,327,616 bytes
 Partition Disk #37, Partition #5
 Partition Size 14.00 GB (15,035,779,584 bytes)
 Partition Starting Offset 32,251,355,136 bytes
 Partition Disk #37, Partition #6
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 47,287,166,976 bytes
 Partition Disk #37, Partition #7
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 49,442,190,336 bytes
 Partition Disk #37, Partition #8
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 53,744,011,776 bytes
 Partition Disk #37, Partition #9
 Partition Size 2.01 GB (2,154,991,104 bytes)
 Partition Starting Offset 55,899,035,136 bytes
 Partition Disk #37, Partition #10
 Partition Size 28.01 GB (30,071,591,424 bytes)
 Partition Starting Offset 58,054,058,496 bytes
 Partition Disk #37, Partition #11
 Partition Size 20.00 GB (21,476,173,824 bytes)
 Partition Starting Offset 88,125,682,176 bytes
 Partition Disk #37, Partition #12
 Partition Size 4.01 GB (4,301,789,184 bytes)
 Partition Starting Offset 109,601,888,256 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model NEC iStorage 2000 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 2
 SCSI Bus 0
 SCSI Logical Unit 1
 SCSI Port 7
 SCSI Target ID 0
 Sectors/Track 63
 Size 199.70 GB (214,424,824,320 bytes)
 Total Cylinders 26,069
 Total Sectors 418,798,485
 Total Tracks 6,647,595
 Tracks/Cylinder 255
 Partition Disk #24, Partition #0
 Partition Size 9.77 GB (10,487,199,744 bytes)
 Partition Starting Offset 32,256 bytes
 Partition Disk #24, Partition #1
 Partition Size 189.93 GB (203,937,592,320 bytes)
 Partition Starting Offset 10,487,232,000 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model NEC iStorage 2000 SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 1
 SCSI Bus 0

SCSI Logical Unit 2
 SCSI Port 7
 SCSI Target ID 0
 Sectors/Track 63
 Size 465.66 GB (499,998,320,640 bytes)
 Total Cylinders 60,788
 Total Sectors 976,559,220
 Total Tracks 15,500,940
 Tracks/Cylinder 255
 Partition Disk #25, Partition #0
 Partition Size 465.66 GB (499,998,288,384 bytes)
 Partition Starting Offset 32,256 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model SEAGATE ST336704LC SCSI Disk Device
 Bytes/Sector 512
 Media Loaded No
 Media Type Fixed hard disk
 Partitions Not Available
 SCSI Bus 0
 SCSI Logical Unit 0
 SCSI Port 12
 SCSI Target ID 15
 Sectors/Track 63
 Size 33.87 GB (36,372,188,160 bytes)
 Total Cylinders 4,422
 Total Sectors 71,039,430
 Total Tracks 1,127,610
 Tracks/Cylinder 255
 Partition Disk #38, Partition #0
 Partition Size 345.12 MB (361,880,064 bytes)
 Partition Starting Offset 32,256 bytes
 Partition Disk #38, Partition #1
 Partition Size 33.41 GB (35,878,671,360 bytes)
 Partition Starting Offset 493,516,800 bytes

[SCSI]

Item Value
 Name QLogic QLA23xx PCI Fibre Channel Adapter
 Manufacturer QLogic
 Status OK
 PNP Device ID
 PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&13C0B0C5&0&10

I/O Port 0x00001200-0x000012FF
 Memory Address 0xFBBFF000-0xFBBFFFFF
 IRQ Channel IRQ 20

Name Adaptec SCSI Card 39160 - Ultra160 SCSI
 Manufacturer Adaptec
 Status OK
 PNP Device ID
 PCI\VEN_9005&DEV_00C0&SUBSYS_F6209005&REV_01\3&29E81982&0&10

I/O Port 0x00001300-0x000013FF
 Memory Address 0xFB7FF000-0xFB7FFFFF
 IRQ Channel IRQ 212

Name Adaptec SCSI Card 39160 - Ultra160 SCSI
 Manufacturer Adaptec
 Status OK
 PNP Device ID
 PCI\VEN_9005&DEV_00C0&SUBSYS_F6209005&REV_01\3&29E81982&0&11

I/O Port 0x00001400-0x000014FF
 Memory Address 0xFB7FE000-0xFB7FEFFF
 IRQ Channel IRQ 213

Name QLogic QLA23xx PCI Fibre Channel Adapter
 Manufacturer QLogic
 Status OK
 PNP Device ID
 PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&172E68DD&0&10

I/O Port 0x00001500-0x000015FF
 Memory Address 0xFB3FF000-0xFB3FFFFF
 IRQ Channel IRQ 272

Name QLogic QLA23xx PCI Fibre Channel Adapter
 Manufacturer QLogic
 Status OK
 PNP Device ID
 PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&474B838&0&10
 I/O Port 0x00001600-0x000016FF
 Memory Address 0xFAFFF000-0xFAFFFFF
 IRQ Channel IRQ 276

Name QLogic QLA23xx PCI Fibre Channel Adapter
 Manufacturer QLogic
 Status OK
 PNP Device ID
 PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&20FEA912&0&10

I/O Port 0x00001700-0x000017FF
 Memory Address 0xFABFF000-0xFABFFFFF
 IRQ Channel IRQ 468

Name QLogic QLA23xx PCI Fibre Channel Adapter
 Manufacturer QLogic
 Status OK
 PNP Device ID
 PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&AD74055&0&10
 I/O Port 0x00001800-0x000018FF
 Memory Address 0xFA7FF000-0xFA7FFFFF
 IRQ Channel IRQ 596

Name QLogic QLA23xx PCI Fibre Channel Adapter
 Manufacturer QLogic
 Status OK
 PNP Device ID
 PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&23C0707C&0&10

I/O Port 0x00001900-0x000019FF
 Memory Address 0xFA3FF000-0xFA3FFFFF
 IRQ Channel IRQ 656

Name QLogic QLA23xx PCI Fibre Channel Adapter
Manufacturer QLogic
Status OK
PNP Device ID
PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&1106BFD7&0&10

I/O Port 0x00001A00-0x00001AFF
Memory Address 0xF9FFF000-0xF9FFFFF
IRQ Channel IRQ 660

Name QLogic QLA23xx PCI Fibre Channel Adapter
Manufacturer QLogic
Status OK
PNP Device ID
PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&1B2F0CE&0&10

I/O Port 0x00001B00-0x00001BFF
Memory Address 0xF9BFF000-0xF9BFFFF
IRQ Channel IRQ 784

Name QLogic QLA23xx PCI Fibre Channel Adapter
Manufacturer QLogic
Status OK
PNP Device ID
PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&146CA173&0&10

I/O Port 0x00001C00-0x00001CFF
Memory Address 0xF97FF000-0xF97FFFF
IRQ Channel IRQ 788

Name QLogic QLA23xx PCI Fibre Channel Adapter
Manufacturer QLogic
Status OK
PNP Device ID
PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&39E002BD&0&10

I/O Port 0x00001E00-0x0000FFFF
Memory Address 0xF8FFF000-0xF8FFFFFF
IRQ Channel IRQ 916

[IDE]

Item Value
Name Intel(r) IA64 Bus Master IDE Controller
Manufacturer Intel
Status OK
PNP Device ID
PCI\VEN_8086&DEV_7601&SUBSYS_01061033&REV_01\3&267A616A&0&11

I/O Port 0x00001120-0x0000112F

Name Primary IDE Channel
Manufacturer (Standard IDE ATA/ATAPI controllers)
Status OK
PNP Device ID PCI\IDE\IDECHANNEL\4&31FC9C45&0&0
I/O Port 0x000001F0-0x000001F7
I/O Port 0x000003F6-0x000003F6
IRQ Channel IRQ 14

[Printing]

Name	Driver	Port Name	Server Name
------	--------	-----------	-------------

[Problem Devices]

Device	PNP Device ID	Error Code
System Interrupt Controller	PCI\VEN_1033&DEV_00FE&SUBSYS_00FE1033&REV_00\3&267A616A&0&08	The drivers for this device are not installed.
Not Available	ACPI\NEC4171\0	The drivers for this device are not installed.
System Interrupt Controller	PCI\VEN_1033&DEV_00FD&SUBSYS_00FD1033&REV_00\3&267A616A&0&70	The drivers for this device are not installed.
System Interrupt Controller	PCI\VEN_1033&DEV_00FE&SUBSYS_00FE1033&REV_00\3&29E81982&0&08	The drivers for this device are not installed.
System Interrupt Controller	PCI\VEN_1033&DEV_00FE&SUBSYS_00FE1033&REV_00\3&172E68DD&0&08	The drivers for this device are not installed.
System Interrupt Controller	PCI\VEN_1033&DEV_00FE&SUBSYS_00FE1033&REV_00\3&20FEA912&0&08	The drivers for this device are not installed.
System Interrupt Controller	PCI\VEN_1033&DEV_00FE&SUBSYS_00FE1033&REV_00\3&AD74055&0&08	The drivers for this device are not installed.
System Interrupt Controller	PCI\VEN_1033&DEV_00FE&SUBSYS_00FE1033&REV_00\3&23C0707C&0&08	The drivers for this device are not installed.
System Interrupt Controller	PCI\VEN_1033&DEV_00FE&SUBSYS_00FE1033&REV_00\3&1B2F0CE&0&08	The drivers for this device are not installed.
System Interrupt Controller	PCI\VEN_1033&DEV_00FE&SUBSYS_00FE1033&REV_00\3&27265218&0&08	The drivers for this device are not installed.

[USB]

Device	PNP Device ID
Intel(r) 82372FB PCI to USB Universal Host Controller	PCI\VEN_8086&DEV_7602&SUBSYS_01061033&REV_01\3&267A616A&0&12

USB Root Hub USB\R00T_HUB\4&5C3BD33&0

[Software Environment]

[System Drivers]

Name	Description	File	Type	Started	Start Mode	State
	Status	Error Control	Accept	Pause	Accept	Stop
abiosdsk	Abiosdsk	Not Available	Kernel	Driver	No	
acpi	Disabled	Stopped	OK	Ignore	No	No
	Microsoft ACPI Driver	c:\windows\system32\drivers\acpi.sys				
	Kernel Driver	Yes	Boot	Running	OK	Normal
						No
						Yes
acpiec	ACPIEC	c:\windows\system32\drivers\acpiec.sys				Kernel
Driver	No	Disabled	Stopped	OK	Normal	No

adpu160m	adpu160m	c:\windows\system32\drivers\adpu160m.sys	Kernel Driver	Yes	Boot	Running OK	Normal	No	Yes
adpu320	adpu320	Not Available	Kernel Driver	No	Disabled				
afcnt	afcnt	Not Available	Kernel Driver	No	Disabled				
afd	AFD Networking Support	Environment	Kernel Driver	Yes	Auto				
aic78u2	aic78u2	Not Available	Kernel Driver	No	Disabled				
aic78xx	aic78xx	Not Available	Kernel Driver	No	Disabled				
aliide	AliIde	Not Available	Kernel Driver	No	Disabled				
asynctac	RAS Asynchronous Media Driver		Kernel Driver	No					
atapi	Standard IDE/ESDI Hard Disk Controller		Kernel Driver	Yes	Boot				
atdisk	Atdisk	Not Available	Kernel Driver	No	Disabled				
atmarpc	ATM ARP Client Protocol		Kernel Driver	No					
audstub	Audio Stub Driver		Kernel Driver	Yes	Manual Running OK	Normal	No	Yes	
beep	Beep	c:\windows\system32\drivers\beep.sys	Kernel Driver	Yes					
cbidf2k	cbidf2k	c:\windows\system32\drivers\cbidf2k.sys	Kernel Driver	No	Disabled	Stopped OK	Normal	No	No
cdfs	Cdfs	c:\windows\system32\drivers\cdfs.sys	File System Driver	Yes	Disabled	Running OK	Normal	No	Yes
cdrom	CD-ROM Driver	c:\windows\system32\drivers\cdrom.sys	Kernel Driver	Yes	System Running OK	Normal	No	Yes	
changer	Changer	Not Available	Kernel Driver	No	System Stopped OK				
clusdisk	Cluster Disk Driver	c:\windows\system32\drivers\clusdisk.sys	Kernel Driver	No	Disabled	Stopped OK	Normal	No	No
cmdide	CmdIde	Not Available	Kernel Driver	No	Disabled				
cpqarry2	cpqarry2	Not Available	Kernel Driver	No	Disabled	Stopped OK	Normal	No	No
cpqcissm	cpqcissm	Not Available	Kernel Driver	No	Disabled	Stopped OK	Normal	No	No
cpqfcalm	cpqfcalm	Not Available	Kernel Driver	No	Disabled	Stopped OK	Normal	No	No
crcdisk	CRC Disk Filter Driver	c:\windows\system32\drivers\crcdisk.sys	Kernel Driver	Yes	Boot	Running OK	Normal	No	Yes
dfsdriver	DfsDriver	c:\windows\system32\drivers\dfs.sys	File System Driver	Yes	Boot	Running OK	Normal	No	Yes
disk	Disk Driver	c:\windows\system32\drivers\disk.sys	Kernel Driver	Yes	Boot	Running OK	Normal	No	Yes
dmboot	dmboot	c:\windows\system32\drivers\dmboot.sys	Kernel Driver	No	Disabled	Stopped OK	Normal	No	No
dmio	Logical Disk Manager Driver	c:\windows\system32\drivers\dmio.sys	Kernel Driver	Yes	Boot	Running OK	Normal	No	Yes
dmload	dmload	c:\windows\system32\drivers\dmload.sys	Kernel Driver	Yes	Boot	Running OK	Normal	No	Yes
e1000	Intel(R) PRO/1000 Device Driver	c:\windows\system32\drivers\Intel\1000\645.sys	Kernel Driver	Yes	Manual Running OK	Normal	No	Yes	
em	em	\\??\c:\windows\system32\drivers\em.sys	Kernel Driver	No	Manual Stopped OK	Normal	No	No	
fastfat	Fastfat	c:\windows\system32\drivers\fastfat.sys	File System Driver	Yes	Disabled	Running OK	Normal	No	Yes
fdc	Fdc	c:\windows\system32\drivers\fdc.sys	Kernel Driver	No	System Stopped OK	Ignore	No	No	
fips	Fips	c:\windows\system32\drivers\fips.sys	Kernel Driver	Yes	System Running OK	Normal	No	Yes	
flpydisk	Flpydisk	c:\windows\system32\drivers\flpydisk.sys	Kernel Driver	No	System Stopped OK	Ignore	No	No	
ftdisk	Volume Manager Driver	c:\windows\system32\drivers\ftdisk.sys	Kernel Driver	Yes	Boot	Running OK	Normal	No	Yes
gpc	Generic Packet Classifier	c:\windows\system32\drivers\msgpc.sys	Kernel Driver	Yes	Manual Running OK	Normal	No	Yes	
hpn	hpn	Not Available	Kernel Driver	No	Stopped OK	Normal	No	No	Disabled
http	HTTP	c:\windows\system32\drivers\http.sys	Kernel Driver	No	Manual Stopped OK	Normal	No	No	
i2omgmt	i2omgmt	Not Available	Kernel Driver	No	Normal	No	No	No	System Stopped OK
i8042prt	i8042 Keyboard and PS/2 Mouse Port Driver	c:\windows\system32\drivers\i8042prt.sys	Kernel Driver	Yes	System Running OK	Normal	No	Yes	
imapi	CD-Burning Filter Driver	c:\windows\system32\drivers\imapi.sys	Kernel Driver	No	System Stopped OK	Normal	No	No	
intelide	IntelIde	c:\windows\system32\drivers\intelide.sys	Kernel Driver	Yes	Boot	Running OK	Normal	No	Yes
ipfilterdriver	IP Traffic Filter Driver	c:\windows\system32\drivers\ipfltdrv.sys	Kernel Driver	No	Manual Stopped OK	Normal	No	No	
ipinip	IP in IP Tunnel Driver	c:\windows\system32\drivers\ipinip.sys	Kernel Driver	No	Manual Stopped OK	Normal	No	No	
ipnat	IP Network Address Translator	c:\windows\system32\drivers\ipnat.sys	Kernel Driver	No	Manual Stopped OK	Normal	No	No	
ipsec	IPSEC driver	c:\windows\system32\drivers\ipsec.sys	Kernel Driver	Yes	System Running OK	Normal	No	Yes	
isapnp	PnP ISA/EISA Bus Driver	c:\windows\system32\drivers\isapnp.sys	Kernel Driver	Yes	Boot	Running OK	Critical	No	Yes
kbdclass	Keyboard Class Driver	c:\windows\system32\drivers\kbdclass.sys	Kernel Driver	Yes	System Running OK	Normal	No	Yes	

ksecdd	KSecDD	c:\windows\system32\drivers\ksecdd.sys	Kernel Driver	Yes	Boot	Running OK	Normal	No	Yes
lp6nds35	lp6nds35	Not Available	Kernel Driver	No					
mmdd	mmdd	Not Available	Kernel Driver	No					
modem	Modem	c:\windows\system32\drivers\modem.sys	Kernel Driver	No					
mouclass	Mouse Class Driver								
mountmgr	Mount Point Manager								
mrraid35x	mrraid35x	Not Available	Kernel Driver	No					
mrxdav	WebDav Client Redirector								
mrxsm	MRXSMB	c:\windows\system32\drivers\mrxsm.sys	File System Driver	Yes	System	Running OK	Normal	No	Yes
msfs	Msfs	c:\windows\system32\drivers\msfs.sys	File System Driver	Yes	System	Running OK	Normal	No	Yes
mup	Mup	c:\windows\system32\drivers\mup.sys	File System Driver	Yes	Boot	Running OK	Normal	No	Yes
ndis	NDIS	System Driver	c:\windows\system32\drivers\ndis.sys	Kernel Driver	Yes	Boot	Running OK	Normal	No
ndistapi	Remote Access NDIS TAPI Driver								
ndisuio	NDIS Usermode I/O Protocol								
ndiswan	Remote Access NDIS WAN Driver								
ndproxy	NDIS Proxy	c:\windows\system32\drivers\ndproxy.sys	Kernel Driver	Yes	Manual	Running OK	Normal	No	Yes
netbios	NetBIOS Interface	c:\windows\system32\drivers\netbios.sys	File System Driver	Yes	System	Running OK	Normal	No	Yes
netbt	NetBios over Tcpip	c:\windows\system32\drivers\netbt.sys	Kernel Driver	Yes	System	Running OK	Normal	No	Yes
nfrd960	nfrd960	Not Available	Kernel Driver	No					
npfs	Npfs	c:\windows\system32\drivers\npfs.sys	File System Driver	Yes	System	Running OK	Normal	No	Yes
ntfs	Ntfs	c:\windows\system32\drivers\ntfs.sys	File System Driver	Yes	Disabled	Running OK	Normal	No	Yes
null	Null	c:\windows\system32\drivers\null.sys	Kernel Driver	Yes	System	Running OK	Normal	No	Yes
partmgr	Partition Manager	c:\windows\system32\drivers\partmgr.sys	Kernel Driver	Yes	Boot	Running OK	Normal	No	Yes
pci	PCI Bus Driver	c:\windows\system32\drivers\pci.sys	Kernel Driver	Yes	Boot	Running OK	Critical	No	Yes

pciide	PCIIde	Not Available	Kernel Driver	No					
pcmcia	Pcmcia	c:\windows\system32\drivers\pcmcia.sys	Kernel Driver	No					
pdcomp	PDCOMP	Not Available	Kernel Driver	No					
pdframe	PDFRAME	Not Available	Kernel Driver	No					
pdreli	PDRELI	Not Available	Kernel Driver	No					
pdrframe	PDRFRAME	Not Available	Kernel Driver	No					
pptpminiport	WAN Miniport (PPTP)								
processor	Processor Driver								
ptilink	Direct Parallel Link Driver								
ql1080	ql1080	Not Available	Kernel Driver	No					
ql10wnt	ql10wnt	Not Available	Kernel Driver	No					
ql12160	ql12160	Not Available	Kernel Driver	No					
ql1240	ql1240	Not Available	Kernel Driver	No					
ql1280	ql1280	Not Available	Kernel Driver	No					
ql2100	ql2100	Not Available	Kernel Driver	No					
ql2200	ql2200	Not Available	Kernel Driver	No					
ql2300	ql2300	c:\windows\system32\drivers\ql2300.sys	Kernel Driver	Yes	Boot	Running OK	Normal	No	Yes
qldirect	qldirect	c:\windows\system32\drivers\qldirect.sys	Kernel Driver	Yes	Boot	Running OK	Normal	No	Yes
rasacd	Remote Access Auto Connection Driver								
rasl2tp	WAN Miniport (L2TP)								
rasppoe	Remote Access PPPOE Driver								
raspti	Direct Parallel	c:\windows\system32\drivers\raspti.sys	Kernel Driver	Yes	Manual	Running OK	Normal	No	Yes
rdbss	Rdbss	c:\windows\system32\drivers\rdbss.sys	File System Driver	Yes	System	Running OK	Normal	No	Yes
rdpcdd	RDPCDD	c:\windows\system32\drivers\rdpcdd.sys	Kernel Driver	Yes	System	Running OK	Ignore	No	Yes
rdpdr	Terminal Server Device Redirector Driver								

rdpwd	RDPWD	c:\windows\system32\drivers\rdpwd.sys	Kernel Driver	Yes	
	Manual Running OK	Ignore No	Yes		
redbook	Digital CD Audio Playback Filter Driver	c:\windows\system32\drivers\redbook.sys	Kernel Driver	Yes	
	System Running OK	Normal No	Yes		
serenum	Serenum Filter Driver	c:\windows\system32\drivers\serenum.sys			
	Kernel Driver Yes	Manual Running OK	Normal No	Yes	
serial	Serial port driver	c:\windows\system32\drivers\serial.sys			
	Kernel Driver Yes	System Running OK	Ignore No	Yes	
sfloppy	Sfloppy	c:\windows\system32\drivers\sfloppy.sys	Kernel		
Driver	No	System Stopped OK	Ignore No	No	
simbad	Simbad	Not Available	Kernel Driver	No	Disabled
	Stopped OK	Normal No	No		
srv	Srv	c:\windows\system32\drivers\srv.sys	File System Driver		
	Yes	Manual Running OK	Normal No	Yes	
swenum	Software Bus Driver	c:\windows\system32\drivers\swenum.sys			
	Kernel Driver Yes	Manual Running OK	Normal No	Yes	
symc8xx	symc8xx	Not Available	Kernel Driver	No	Disabled
	Stopped OK	Normal No	No		
symmpi	symmpi	Not Available	Kernel Driver	No	Disabled
	Stopped OK	Normal No	No		
sym_hi	sym_hi	Not Available	Kernel Driver	No	Disabled
	Stopped OK	Normal No	No		
sym_u3	sym_u3	Not Available	Kernel Driver	No	Disabled
	Stopped OK	Normal No	No		
tbs	tbs	\\??c:\windows\system32\drivers\tbs.sys	Kernel		
Driver	No	Manual Stopped OK	Normal No	No	
tcpip	TCP/IP Protocol Driver	c:\windows\system32\drivers\tcpip.sys			
	Kernel Driver Yes	System Running OK	Normal No	Yes	
tdpipe	TDPIPE	c:\windows\system32\drivers\tdpipe.sys	Kernel		
Driver	No	Manual Stopped OK	Ignore No	No	
tdtcp	TDTCP	c:\windows\system32\drivers\tdtcp.sys	Kernel Driver	Yes	
	Manual Running OK	Ignore No	Yes		
termdd	Terminal Device Driver	c:\windows\system32\drivers\termdd.sys			
	Kernel Driver Yes	System Running OK	Normal No	Yes	
toside	TosIde	Not Available	Kernel Driver	No	Disabled
	Stopped OK	Normal No	No		
udfs	Udfs	c:\windows\system32\drivers\udfs.sys	File System Driver		
	No	Disabled	Stopped OK	Normal No	No
usbhub	USB2 Enabled Hub	c:\windows\system32\drivers\usbhub.sys			
	Kernel Driver Yes	Manual Running OK	Normal No	Yes	
usbuhci	Microsoft USB Universal Host Controller Miniport Driver	c:\windows\system32\drivers\usbuhci.sys	Kernel Driver	Yes	
	Manual Running OK	Normal No	Yes		
vapamap	Vapamap	c:\windows\system32\drivers\vapamap.sys	Kernel		
Driver	No	Manual Stopped OK	Normal No	No	
vga	vga	c:\windows\system32\drivers\vgapnp.sys	Kernel		
Driver	Yes	Manual Running OK	Ignore No	Yes	
vgasave	VGA Display Controller.	c:\windows\system32\drivers\vga.sys			
	Kernel Driver No	System Stopped OK	Ignore No	No	
viaide	ViaIde	Not Available	Kernel Driver	No	Disabled
	Stopped OK	Normal No	No		

volsnap	Storage volumes	c:\windows\system32\drivers\volsnap.sys			
	Kernel Driver Yes	Boot Running OK	Normal No	Yes	
wanarp	Remote Access IP ARP Driver	c:\windows\system32\drivers\wanarp.sys	Kernel Driver	Yes	
	Manual Running OK	Normal No	Yes		
wdica	WDICA	Not Available	Kernel Driver	No	Manual Stopped OK
	Ignore No	No			
wlbs	Network Load Balancing	c:\windows\system32\drivers\wlbs.sys			
	Kernel Driver No	Manual Stopped OK	Normal No	No	

[Signed Drivers]

Device Name	Signed	Device Class	Driver	Version	Driver Date
Manufacturer	INF Name	Driver Name	Device ID		
Not Available	Not Available	Not Available	Not Available	Not Available	Not
Available	Not Available	Not Available	Not Available	HTREE\ROOT\0	
ACPI IA64-based PC	Yes	COMPUTER	5.2.3790.0	10/1/2002	
(Standard computers)	hal.inf	Not Available	ROOT\ACPI_HAL\0000		
Microsoft ACPI-Compliant System	Yes	SYSTEM	5.2.3790.0		
10/1/2002	Microsoft	acpi.inf	Not Available		
ACPI_HAL\PNP0C08\0					
Processor	Yes	PROCESSOR	5.2.3790.0	10/1/2002	
(Standard processor types)		cpu.inf	Not Available		
ACPI\GENUINEINTEL_-IA64_FAMILY_31_MODEL_2\0					
Processor	Yes	PROCESSOR	5.2.3790.0	10/1/2002	
(Standard processor types)		cpu.inf	Not Available		
ACPI\GENUINEINTEL_-IA64_FAMILY_31_MODEL_2\1					
Processor	Yes	PROCESSOR	5.2.3790.0	10/1/2002	
(Standard processor types)		cpu.inf	Not Available		
ACPI\GENUINEINTEL_-IA64_FAMILY_31_MODEL_2\2					
Processor	Yes	PROCESSOR	5.2.3790.0	10/1/2002	
(Standard processor types)		cpu.inf	Not Available		
ACPI\GENUINEINTEL_-IA64_FAMILY_31_MODEL_2\3					
Processor	Yes	PROCESSOR	5.2.3790.0	10/1/2002	
(Standard processor types)		cpu.inf	Not Available		
ACPI\GENUINEINTEL_-IA64_FAMILY_31_MODEL_2\4					
Processor	Yes	PROCESSOR	5.2.3790.0	10/1/2002	
(Standard processor types)		cpu.inf	Not Available		
ACPI\GENUINEINTEL_-IA64_FAMILY_31_MODEL_2\5					
Processor	Yes	PROCESSOR	5.2.3790.0	10/1/2002	
(Standard processor types)		cpu.inf	Not Available		
ACPI\GENUINEINTEL_-IA64_FAMILY_31_MODEL_2\6					
Processor	Yes	PROCESSOR	5.2.3790.0	10/1/2002	
(Standard processor types)		cpu.inf	Not Available		
ACPI\GENUINEINTEL_-IA64_FAMILY_31_MODEL_2\7					
PCI bus	Yes	SYSTEM	5.2.3790.0	10/1/2002	(Standard system
devices)	machine.inf	Not Available	ACPI\PNP0A03\0		
System Interrupt Controller	Not Available	UNKNOWN	Not Available	Not	
Available	Not Available	Not Available	Not Available		
	PCI\VEN_1033&DEV_00FE&SUBSYS_00FE1033&REV_00\3&267A616A&0&08				
Intel 82372FB PCI to ISA Bridge	Yes	SYSTEM	5.2.3790.0		
10/1/2002	Intel	machine.inf	Not Available		
PCI\VEN_8086&DEV_7600&SUBSYS_00000000&REV_01\3&267A616A&0&10					

Direct memory access controller	Yes	SYSTEM	5.2.3790.0	
10/1/2002 (Standard system devices)		machine.inf	Not Available	
Available ACPI\PNP0200\4&22C34DB0&0				
Programmable interrupt controller	Yes	SYSTEM	5.2.3790.0	
10/1/2002 (Standard system devices)		machine.inf	Not Available	
Available ACPI\PNP0000\4&22C34DB0&0				
System timer	Yes	SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)
machine.inf	Not Available			
Available ACPI\PNP0100\4&22C34DB0&0				
System speaker	Yes	SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)
machine.inf	Not Available			
Available ACPI\PNP0800\4&22C34DB0&0				
System CMOS/real time clock	Yes	SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)
machine.inf	Not Available			
Available ACPI\PNP0B00\4&22C34DB0&0				
Numeric data processor	Yes	SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)
machine.inf	Not Available			
Available ACPI\PNP0C04\4&22C34DB0&0				
Generic Bus	Yes	SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)
machine.inf	Not Available			
Available ACPI\PNP0A05\4&22C34DB0&0				
Extended IO Bus	Yes	SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)
machine.inf	Not Available			
Available ACPI\PNP0A06\1				
Motherboard resources	Yes	SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)
machine.inf	Not Available			
Available ACPI\PNP0C02\1				
Extended IO Bus	Yes	SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)
machine.inf	Not Available			
Available ACPI\PNP0A06\3				
Motherboard resources	Yes	SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)
machine.inf	Not Available			
Available ACPI\PNP0C02\3				
Standard 101/102-Key or Microsoft Natural PS/2 Keyboard	Yes			
KEYBOARD 5.2.3790.0	10/1/2002 (Standard keyboards)			
keyboard.inf	Not Available	ACPI\PNP0303\2		
PS/2 Compatible Mouse	Yes	MOUSE	5.2.3790.0	10/1/2002 (Standard system devices)
Microsoft mouse.inf	Not Available	ACPI\PNP0F13\2		
Communications Port	Yes	PORTS	5.2.3790.0	10/1/2002 (Standard port types)
msports.inf	Not Available			
Available ACPI\PNP0501\2				
Extended IO Bus	Yes	SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)
machine.inf	Not Available			
Available ACPI\PNP0A06\4				
Not Available	Not Available	Not Available	Not Available	Not Available
Available	Not Available	Not Available	Not Available	Not Available
ACPI\NEC4171\0				
Intel(r) IA64 Bus Master IDE Controller	Yes	HDC	5.2.3790.0	10/1/2002 (Standard IDE ATA/ATAPI controllers)
Intel mshdc.inf	Not Available			
Available PCI\VEN_8086&DEV_7601&SUBSYS_01061033&REV_01\3&267A616A&0&11				
Primary IDE Channel	Yes	HDC	5.2.3790.0	10/1/2002 (Standard IDE ATA/ATAPI controllers)
mshdc.inf	Not Available			
Available PCI\IDE\IDECHANNEL\4&31FC9C45&0&0				
CD-ROM Drive	Yes	CDROM	5.2.3790.0	10/1/2002 (Standard CD-ROM drives)
cdrom.inf	Not Available	IDE\CDROM_NEC_DVD-ROM_DV-5800A	1.42\5&8F26A51&0&0.0.0	
Intel(r) 82372FB PCI to USB Universal Host Controller	Yes	USB	5.2.3790.0	10/1/2002 (Standard system devices)
Intel usbport.inf	Not Available			

Available				
PCI\VEN_8086&DEV_7602&SUBSYS_01061033&REV_01\3&267A616A&0&12				
USB Root Hub	Yes	USB	5.2.3790.0	10/1/2002 (Standard USB Host Controller)
usbport.inf	Not Available			
Available USB\ROOT_HUB\4&5C3BD33&0				
Standard VGA Graphics Adapter	Yes	DISPLAY	5.2.3790.0	10/1/2002 (Standard display types)
display.inf	Not Available			
Available PCI\VEN_1002&DEV_4752&SUBSYS_80EB1033&REV_27\3&267A616A&0&20				
Default Monitor	Yes	MONITOR	5.1.2001.0	6/6/2001 (Standard monitor types)
monitor.inf	Not Available			
Available DISPLAY\DEFAULT_MONITOR\4&20D43725&0&12345678&00&04				
System Interrupt Controller	Not Available	UNKNOWN	Not Available	Not Available
Not Available	Not Available	Not Available	Not Available	Not Available
Available PCI\VEN_1033&DEV_00FD&SUBSYS_00FD1033&REV_00\3&267A616A&0&70				
PCI standard host CPU bridge	Yes	SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)
machine.inf	Not Available			
Available PCI\VEN_1033&DEV_00F9&SUBSYS_00000000&REV_02\3&267A616A&0&78				
PCI standard host CPU bridge	Yes	SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)
machine.inf	Not Available			
Available PCI\VEN_1033&DEV_00F9&SUBSYS_00000000&REV_02\3&267A616A&0&80				
PCI standard host CPU bridge	Yes	SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)
machine.inf	Not Available			
Available PCI\VEN_1033&DEV_00F9&SUBSYS_00000000&REV_02\3&267A616A&0&88				
PCI standard host CPU bridge	Yes	SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)
machine.inf	Not Available			
Available PCI\VEN_1033&DEV_00F9&SUBSYS_00000000&REV_02\3&267A616A&0&90				
PCI standard host CPU bridge	Yes	SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)
machine.inf	Not Available			
Available PCI\VEN_1033&DEV_00F9&SUBSYS_00000000&REV_02\3&267A616A&0&98				
PCI bus	Yes	SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)
machine.inf	Not Available	ACPI\PNP0A03\1		
QLogic QLA23xx PCI Fibre Channel Adapter	Yes	SCSIADAPTER		
8.2.2.10 4/16/2003	QLogic	oem0.inf	Not Available	
Available				
PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&13C0B0C5&0&10				
Disk drive	Yes	DISKDRIVE	5.2.3790.0	10/1/2002 (Standard disk drives)
disk.inf	Not Available			
Available SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2200\4&80CA89A&0&000				
Disk drive	Yes	DISKDRIVE	5.2.3790.0	10/1/2002 (Standard disk drives)
disk.inf	Not Available			
Available SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2200\4&80CA89A&0&001				
Disk drive	Yes	DISKDRIVE	5.2.3790.0	10/1/2002 (Standard disk drives)
disk.inf	Not Available			
Available SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2200\4&80CA89A&0&010				
Disk drive	Yes	DISKDRIVE	5.2.3790.0	10/1/2002 (Standard disk drives)
disk.inf	Not Available			

SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2200\4&80CA89A&0&011

Qlogic processor device Yes SYSTEM 5.2.3790.0 10/1/2002
 QLOGIC scsidev.inf Not Available
 SCSI\PROCESSOR&VEN_QLOGIC&PROD_PSEUDO_DEVICE&REV_\4&80CA89A&0&07F0

PCI bus Yes SYSTEM 5.2.3790.0 10/1/2002 (Standard system devices)
 machine.inf Not Available ACPI\PNP0A03\3
 System Interrupt Controller Not Available UNKNOWN Not Available Not Available
 Not Available Not Available Not Available Not Available
 PCI\VEN_1033&DEV_00FE&SUBSYS_00FE1033&REV_00\3&29E81982&0&08

Adaptec SCSI Card 39160 - Ultra160 SCSI Yes SCSIADAPTER
 5.2.3790.0 10/1/2002 Adaptec pnp SCSI.inf Not Available
 PCI\VEN_9005&DEV_00C0&SUBSYS_F6209005&REV_01\3&29E81982&0&10

Adaptec SCSI Card 39160 - Ultra160 SCSI Yes SCSIADAPTER
 5.2.3790.0 10/1/2002 Adaptec pnp SCSI.inf Not Available
 PCI\VEN_9005&DEV_00C0&SUBSYS_F6209005&REV_01\3&29E81982&0&11

Qlogic processor device Yes SYSTEM 5.2.3790.0 10/1/2002
 QLOGIC scsidev.inf Not Available
 SCSI\PROCESSOR&VEN_QLOGIC&PROD_GEM359&REV_1.06\4&1AF8FFB&0&040

Qlogic processor device Yes SYSTEM 5.2.3790.0 10/1/2002
 QLOGIC scsidev.inf Not Available
 SCSI\PROCESSOR&VEN_QLOGIC&PROD_GEM359&REV_1.06\4&1AF8FFB&0&050

Disk drive Yes DISKDRIVE 5.2.3790.0 10/1/2002
 (Standard disk drives) disk.inf Not Available
 SCSI\DISK&VEN_SEAGATE&PROD_ST336704LC&REV_0004\4&1AF8FFB&0&0F0

PCI bus Yes SYSTEM 5.2.3790.0 10/1/2002 (Standard system devices)
 machine.inf Not Available ACPI\PNP0A03\4
 System Interrupt Controller Not Available UNKNOWN Not Available Not Available
 Not Available Not Available Not Available Not Available
 PCI\VEN_1033&DEV_00FE&SUBSYS_00FE1033&REV_00\3&172E68DD&0&08

QLogic QLA23xx PCI Fibre Channel Adapter Yes SCSIADAPTER
 8.2.2.10 4/16/2003 QLogic oem0.inf Not Available
 PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&172E68DD&0&10

Disk drive Yes DISKDRIVE 5.2.3790.0 10/1/2002
 (Standard disk drives) disk.inf Not Available
 SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2200\4&7C3A82C&0&000

Disk drive Yes DISKDRIVE 5.2.3790.0 10/1/2002
 (Standard disk drives) disk.inf Not Available
 SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2200\4&7C3A82C&0&001

Disk drive Yes DISKDRIVE 5.2.3790.0 10/1/2002
 (Standard disk drives) disk.inf Not Available
 SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2200\4&7C3A82C&0&010

Disk drive Yes DISKDRIVE 5.2.3790.0 10/1/2002
 (Standard disk drives) disk.inf Not Available

SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2200\4&7C3A82C&0&011

Qlogic processor device Yes SYSTEM 5.2.3790.0 10/1/2002
 QLOGIC scsidev.inf Not Available
 SCSI\PROCESSOR&VEN_QLOGIC&PROD_PSEUDO_DEVICE&REV_\4&7C3A82C&0&07F0

PCI bus Yes SYSTEM 5.2.3790.0 10/1/2002 (Standard system devices)
 machine.inf Not Available ACPI\PNP0A03\5
 QLogic QLA23xx PCI Fibre Channel Adapter Yes SCSIADAPTER
 8.2.2.10 4/16/2003 QLogic oem0.inf Not Available
 PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&474B838&0&10

Disk drive Yes DISKDRIVE 5.2.3790.0 10/1/2002
 (Standard disk drives) disk.inf Not Available
 SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2200\4&39ADE309&0&000

Disk drive Yes DISKDRIVE 5.2.3790.0 10/1/2002
 (Standard disk drives) disk.inf Not Available
 SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2200\4&39ADE309&0&001

Disk drive Yes DISKDRIVE 5.2.3790.0 10/1/2002
 (Standard disk drives) disk.inf Not Available
 SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2200\4&39ADE309&0&010

Disk drive Yes DISKDRIVE 5.2.3790.0 10/1/2002
 (Standard disk drives) disk.inf Not Available
 SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2200\4&39ADE309&0&011

Qlogic processor device Yes SYSTEM 5.2.3790.0 10/1/2002
 QLOGIC scsidev.inf Not Available
 SCSI\PROCESSOR&VEN_QLOGIC&PROD_PSEUDO_DEVICE&REV_\4&39ADE309&0&07F0

PCI bus Yes SYSTEM 5.2.3790.0 10/1/2002 (Standard system devices)
 machine.inf Not Available ACPI\PNP0A03\7
 System Interrupt Controller Not Available UNKNOWN Not Available Not Available
 Not Available Not Available Not Available Not Available
 PCI\VEN_1033&DEV_00FE&SUBSYS_00FE1033&REV_00\3&20FEA912&0&08

QLogic QLA23xx PCI Fibre Channel Adapter Yes SCSIADAPTER
 8.2.2.10 4/16/2003 QLogic oem0.inf Not Available
 PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&20FEA912&0&10

Disk drive Yes DISKDRIVE 5.2.3790.0 10/1/2002
 (Standard disk drives) disk.inf Not Available
 SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2300\4&1CA05993&0&001

Disk drive Yes DISKDRIVE 5.2.3790.0 10/1/2002
 (Standard disk drives) disk.inf Not Available
 SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2300\4&1CA05993&0&002

Qlogic processor device Yes SYSTEM 5.2.3790.0 10/1/2002
 QLOGIC scsidev.inf Not Available
 SCSI\PROCESSOR&VEN_QLOGIC&PROD_PSEUDO_DEVICE&REV_\4&1CA05993&0&07F0

PCI bus Yes SYSTEM 5.2.3790.0 10/1/2002 (Standard system devices)
 machine.inf Not Available ACPI\PNP0A03\9

System Interrupt Controller Not Available UNKNOWN Not Available Not Available
 Not Available Not Available Not Available Not Available
 PCI\VEN_1033&DEV_00FE&SUBSYS_00FE1033&REV_00\3&AD74055&0&08
 QLogic QLA23xx PCI Fibre Channel Adapter Yes SCSIADAPTER
 8.2.2.10 4/16/2003 QLogic oem0.inf Not Available
 PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&AD74055&0&10
 Disk drive Yes DISKDRIVE 5.2.3790.0 10/1/2002
 (Standard disk drives) disk.inf Not Available
 SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2200\4&348718C2&0&000
 Disk drive Yes DISKDRIVE 5.2.3790.0 10/1/2002
 (Standard disk drives) disk.inf Not Available
 SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2200\4&348718C2&0&001
 Disk drive Yes DISKDRIVE 5.2.3790.0 10/1/2002
 (Standard disk drives) disk.inf Not Available
 SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2200\4&348718C2&0&010
 Disk drive Yes DISKDRIVE 5.2.3790.0 10/1/2002
 (Standard disk drives) disk.inf Not Available
 SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2200\4&348718C2&0&011
 Qlogic processor device Yes SYSTEM 5.2.3790.0 10/1/2002
 QLOGIC scsdev.inf Not Available
 SCSI\PROCESSOR&VEN_QLOGIC&PROD_PSEUDO_DEVICE&REV_\4&348718C2&0&07F0
 PCI bus Yes SYSTEM 5.2.3790.0 10/1/2002 (Standard system
 devices) machine.inf Not Available ACPI\PNP0A03\A
 System Interrupt Controller Not Available UNKNOWN Not Available Not Available
 Not Available Not Available Not Available Not Available
 PCI\VEN_1033&DEV_00FE&SUBSYS_00FE1033&REV_00\3&23C0707C&0&08
 QLogic QLA23xx PCI Fibre Channel Adapter Yes SCSIADAPTER
 8.2.2.10 4/16/2003 QLogic oem0.inf Not Available
 PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&23C0707C&0&10
 Disk drive Yes DISKDRIVE 5.2.3790.0 10/1/2002
 (Standard disk drives) disk.inf Not Available
 SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2200\4&B5C9978&0&000
 Disk drive Yes DISKDRIVE 5.2.3790.0 10/1/2002
 (Standard disk drives) disk.inf Not Available
 SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2200\4&B5C9978&0&001
 Disk drive Yes DISKDRIVE 5.2.3790.0 10/1/2002
 (Standard disk drives) disk.inf Not Available
 SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2200\4&B5C9978&0&010
 Disk drive Yes DISKDRIVE 5.2.3790.0 10/1/2002
 (Standard disk drives) disk.inf Not Available
 SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2200\4&B5C9978&0&011
 Qlogic processor device Yes SYSTEM 5.2.3790.0 10/1/2002
 QLOGIC scsdev.inf Not Available
 SCSI\PROCESSOR&VEN_QLOGIC&PROD_PSEUDO_DEVICE&REV_\4&B5C9978&0&07F0

PCI bus Yes SYSTEM 5.2.3790.0 10/1/2002 (Standard system
 devices) machine.inf Not Available ACPI\PNP0A03\B
 QLogic QLA23xx PCI Fibre Channel Adapter Yes SCSIADAPTER
 8.2.2.10 4/16/2003 QLogic oem0.inf Not Available
 PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&1106BFD7&0&10
 Disk drive Yes DISKDRIVE 5.2.3790.0 10/1/2002
 (Standard disk drives) disk.inf Not Available
 SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2200\4&35C513CA&0&000
 Disk drive Yes DISKDRIVE 5.2.3790.0 10/1/2002
 (Standard disk drives) disk.inf Not Available
 SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2200\4&35C513CA&0&001
 Disk drive Yes DISKDRIVE 5.2.3790.0 10/1/2002
 (Standard disk drives) disk.inf Not Available
 SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2200\4&35C513CA&0&010
 Disk drive Yes DISKDRIVE 5.2.3790.0 10/1/2002
 (Standard disk drives) disk.inf Not Available
 SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2200\4&35C513CA&0&011
 Qlogic processor device Yes SYSTEM 5.2.3790.0 10/1/2002
 QLOGIC scsdev.inf Not Available
 SCSI\PROCESSOR&VEN_QLOGIC&PROD_PSEUDO_DEVICE&REV_\4&35C513CA&0&07F0
 PCI bus Yes SYSTEM 5.2.3790.0 10/1/2002 (Standard system
 devices) machine.inf Not Available ACPI\PNP0A03\C
 System Interrupt Controller Not Available UNKNOWN Not Available Not Available
 Not Available Not Available Not Available Not Available
 PCI\VEN_1033&DEV_00FE&SUBSYS_00FE1033&REV_00\3&1B2F0CE&0&08
 QLogic QLA23xx PCI Fibre Channel Adapter Yes SCSIADAPTER
 8.2.2.10 4/16/2003 QLogic oem0.inf Not Available
 PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&1B2F0CE&0&10
 Disk drive Yes DISKDRIVE 5.2.3790.0 10/1/2002
 (Standard disk drives) disk.inf Not Available
 SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2200\4&B1193BE&0&000
 Disk drive Yes DISKDRIVE 5.2.3790.0 10/1/2002
 (Standard disk drives) disk.inf Not Available
 SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2200\4&B1193BE&0&001
 Disk drive Yes DISKDRIVE 5.2.3790.0 10/1/2002
 (Standard disk drives) disk.inf Not Available
 SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2200\4&B1193BE&0&010
 Disk drive Yes DISKDRIVE 5.2.3790.0 10/1/2002
 (Standard disk drives) disk.inf Not Available
 SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2200\4&B1193BE&0&011
 Qlogic processor device Yes SYSTEM 5.2.3790.0 10/1/2002
 QLOGIC scsdev.inf Not Available
 SCSI\PROCESSOR&VEN_QLOGIC&PROD_PSEUDO_DEVICE&REV_\4&B1193BE&0&07F0
 PCI bus Yes SYSTEM 5.2.3790.0 10/1/2002 (Standard system
 devices) machine.inf Not Available ACPI\PNP0A03\D

QLogic QLA23xx PCI Fibre Channel Adapter	Yes	SCSIADAPTER
8.2.2.10 4/16/2003 QLogic oem0.inf	Not	
Available		
PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&146CA173&0&10		
Disk drive	Yes	DISKDRIVE 5.2.3790.0 10/1/2002
(Standard disk drives) disk.inf	Not Available	
SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2200\4&4BB1442&0&000		
Disk drive	Yes	DISKDRIVE 5.2.3790.0 10/1/2002
(Standard disk drives) disk.inf	Not Available	
SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2200\4&4BB1442&0&001		
Disk drive	Yes	DISKDRIVE 5.2.3790.0 10/1/2002
(Standard disk drives) disk.inf	Not Available	
SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2200\4&4BB1442&0&010		
Disk drive	Yes	DISKDRIVE 5.2.3790.0 10/1/2002
(Standard disk drives) disk.inf	Not Available	
SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2200\4&4BB1442&0&011		
Qlogic processor device	Yes	SYSTEM 5.2.3790.0 10/1/2002
QLOGIC scsidev.inf	Not Available	
SCSI\PROCESSOR&VEN_QLOGIC&PROD_PSEUDO_DEVICE&REV_\4&4BB1442&0&07F0		
PCI bus	Yes	SYSTEM 5.2.3790.0 10/1/2002 (Standard system devices)
machine.inf	Not Available	ACPI\PNP0A03\E
System Interrupt Controller	Not Available	Not Available
Available	Not Available	Not Available
Available		
PCI\VEN_1033&DEV_00FE&SUBSYS_00FE1033&REV_00\3&27265218&0&08		
Intel(R) PRO/1000 XT Server Adapter	Yes	NET 6.3.6.31
10/1/2002 Intel nete1000.inf	Not Available	
PCI\VEN_8086&DEV_1008&SUBSYS_11078086&REV_02\3&27265218&0&18		
PCI bus	Yes	SYSTEM 5.2.3790.0 10/1/2002 (Standard system devices)
machine.inf	Not Available	ACPI\PNP0A03\F
QLogic QLA23xx PCI Fibre Channel Adapter	Yes	SCSIADAPTER
8.2.2.10 4/16/2003 QLogic oem0.inf	Not	
Available		
PCI\VEN_1077&DEV_2312&SUBSYS_01001077&REV_02\3&39E002BD&0&10		
Disk drive	Yes	DISKDRIVE 5.2.3790.0 10/1/2002
(Standard disk drives) disk.inf	Not Available	
SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2200\4&2FC465E&0&000		
Disk drive	Yes	DISKDRIVE 5.2.3790.0 10/1/2002
(Standard disk drives) disk.inf	Not Available	
SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2200\4&2FC465E&0&001		
Disk drive	Yes	DISKDRIVE 5.2.3790.0 10/1/2002
(Standard disk drives) disk.inf	Not Available	
SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2200\4&2FC465E&0&010		
Disk drive	Yes	DISKDRIVE 5.2.3790.0 10/1/2002
(Standard disk drives) disk.inf	Not Available	
SCSI\DISK&VEN_NEC&PROD_ISTORAGE_2000&REV_2200\4&2FC465E&0&011		

Qlogic processor device	Yes	SYSTEM 5.2.3790.0	10/1/2002
QLOGIC scsidev.inf	Not Available		
SCSI\PROCESSOR&VEN_QLOGIC&PROD_PSEUDO_DEVICE&REV_\4&2FC465E&0&07F0			
ACPI Thermal Zone	Yes	SYSTEM 5.2.3790.0	10/1/2002
(Standard system devices)		machine.inf	Not Available
ACPI\THERMALZONE\PROC			
ACPI Thermal Zone	Yes	SYSTEM 5.2.3790.0	10/1/2002
(Standard system devices)		machine.inf	Not Available
ACPI\THERMALZONE\CELL			
ACPI Thermal Zone	Yes	SYSTEM 5.2.3790.0	10/1/2002
(Standard system devices)		machine.inf	Not Available
ACPI\THERMALZONE\XBC_			
ACPI Thermal Zone	Yes	SYSTEM 5.2.3790.0	10/1/2002
(Standard system devices)		machine.inf	Not Available
ACPI\THERMALZONE\PCXA			
ACPI Fixed Feature Button	Yes	SYSTEM 5.2.3790.0	10/1/2002
(Standard system devices)		machine.inf	Not Available
ACPI\FIXEDBUTTON\2&DABA3FF&0			
Logical Disk Manager	Yes	SYSTEM 5.2.3790.0	10/1/2002
(Standard system devices)		machine.inf	Not Available
ROOT\DMIO\0000			
Volume Manager	Yes	SYSTEM 5.2.3790.0	10/1/2002 (Standard system devices)
machine.inf	Not Available		ROOT\FTDISK\0000
Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002 Microsoft
volume.inf	Not Available		
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5490FFSET3200537A00LENG			
TH4ABAA05A00			
Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002 Microsoft
volume.inf	Not Available		
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5490FFSET7E0000LENGTH80			
728E00			
Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002 Microsoft
volume.inf	Not Available		
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5490FFSET80F10C00LENGTH			
80728E00			
Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002 Microsoft
volume.inf	Not Available		
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5490FFSET101641800LENGT			
H80728E00			
Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002 Microsoft
volume.inf	Not Available		
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5490FFSET181D72400LENGT			
H6007CFE00			
Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002 Microsoft
volume.inf	Not Available		
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5490FFSET78254A000LENGT			
H38033CA00			
Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002 Microsoft
volume.inf	Not Available		
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5490FFSETB0288E800LENGT			
H80728E00			
Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002 Microsoft
volume.inf	Not Available		
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5490FFSETB82FBF400LENGT			
H100681800			
Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002 Microsoft
volume.inf	Not Available		

H700681200	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54A0FFSETD844AA200				LENGTH
TH500146800	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54A0FFSET1484B33200				LENGTH
TH100681800	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54A0FFSET1984C81800				LENGTH
TH4ABAA05A00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54B0FFSET7E0000				LENGTH
80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54B0FFSET80F10C00				LENGTH
H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54B0FFSET101641800				LENGTH
H6007CFE00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54B0FFSET181D72400				LENGTH
H38033CA00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54B0FFSETB0288E800				LENGTH
H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54B0FFSETB82FBF400				LENGTH
H100681800	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54B0FFSETC83648A00				LENGTH
H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54B0FFSETD03D79600				LENGTH
H700681200	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54B0FFSETD844AA200				LENGTH

STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54B0FFSET1484B33200LENG
 TH500146800
 Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54B0FFSET1984C81800LENG
 TH100681800
 Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54C0FFSET3200537A00LENG
 TH4ABAA05A00
 Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54C0FFSET7E0000LENGTH80
 728E00
 Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54C0FFSET80F10C00LENGTH
 80728E00
 Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54C0FFSET101641800LENGT
 H80728E00
 Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54C0FFSET181D72400LENGT
 H6007CFE00
 Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54C0FFSET78254A000LENGT
 H38033CA00
 Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54C0FFSETB0288E800LENGT
 H80728E00
 Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54C0FFSETB82FBF400LENGT
 H100681800
 Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54C0FFSETC83648A00LENGT
 H80728E00
 Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54C0FFSETD03D79600LENGT
 H80728E00
 Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54C0FFSETD844AA200LENGT
 H700681200
 Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54C0FFSET1484B33200LENG
 TH500146800
 Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54C0FFSET1984C81800LENG
 TH100681800

Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&GPTPARTITION{4DDEA260-B23A-01C4-507B-
 9E5F8078F531}
 Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&GPTPARTITION{4E2C7440-B23A-01C4-F1B3-
 12714F758821}
 Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&GPTPARTITION{5315C740-B23A-01C4-D931-
 F8428177D974}
 Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5420FFSET3200537A00LENG
 TH4ABAA05A00
 Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5420FFSET7E0000LENGTH80
 728E00
 Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5420FFSET80F10C00LENGTH
 80728E00
 Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5420FFSET101641800LENGT
 H80728E00
 Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5420FFSET181D72400LENGT
 H6007CFE00
 Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5420FFSET78254A000LENGT
 H38033CA00
 Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5420FFSETB0288E800LENGT
 H80728E00
 Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5420FFSETB82FBF400LENGT
 H100681800
 Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5420FFSETD03D79600LENGT
 H80728E00
 Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available
 STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5420FFSETD844AA200LENGT
 H700681200
 Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
 volume.inf Not Available

Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002	Microsoft
	volume.inf	Not Available			
TH4ABAA05A00		STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5440FFSET3200537A00	LENGTH		
Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002	Microsoft
	volume.inf	Not Available			
728E00		STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5440FFSET7E0000	LENGTH		
Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002	Microsoft
	volume.inf	Not Available			
80728E00		STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5440FFSET80F10C00	LENGTH		
Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002	Microsoft
	volume.inf	Not Available			
H80728E00		STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5440FFSET101641800	LENGTH		
Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002	Microsoft
	volume.inf	Not Available			
H6007CFE00		STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5440FFSET181D72400	LENGTH		
Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002	Microsoft
	volume.inf	Not Available			
H38033CA00		STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5440FFSET78254A000	LENGTH		
Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002	Microsoft
	volume.inf	Not Available			
H80728E00		STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5440FFSETB0288E800	LENGTH		
Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002	Microsoft
	volume.inf	Not Available			
H100681800		STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5440FFSETB82FBF400	LENGTH		
Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002	Microsoft
	volume.inf	Not Available			
H80728E00		STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5440FFSETC83648A00	LENGTH		
Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002	Microsoft
	volume.inf	Not Available			
H80728E00		STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5440FFSETD03D79600	LENGTH		
Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002	Microsoft
	volume.inf	Not Available			
H700681200		STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5440FFSETD844AA200	LENGTH		
Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002	Microsoft
	volume.inf	Not Available			
TH500146800		STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5440FFSET1484B33200	LENGTH		
Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002	Microsoft
	volume.inf	Not Available			
TH100681800		STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5440FFSET1984C81800	LENGTH		
Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002	Microsoft
	volume.inf	Not Available			
TH4ABAA05A00		STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5450FFSET3200537A00	LENGTH		
Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002	Microsoft
	volume.inf	Not Available			

H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5460FFSET101641800LENGTH				
H6007CFE00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5460FFSET181D72400LENGTH				
H38033CA00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5460FFSET78254A000LENGTH				
H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5460FFSETB0288E800LENGTH				
H100681800	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5460FFSETB82FBF400LENGTH				
H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5460FFSETC83648A00LENGTH				
H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5460FFSETD03D79600LENGTH				
H700681200	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5460FFSETD844AA200LENGTH				
TH500146800	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5460FFSET1984C81800LENGTH				
TH100681800	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5470FFSET3200537A00LENGTH				
728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5470FFSET7E00000LENGTHH80				
80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5470FFSET80F10C00LENGTHH				
H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5470FFSET101641800LENGTH				
	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		

H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5580FFSETB0288E800LENGTH				
H100681800	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5580FFSETB82FBF400LENGTH				
H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5580FFSETC83648A00LENGTH				
H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5580FFSETD03D79600LENGTH				
H700681200	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5580FFSETD844AA200LENGTH				
TH500146800	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5580FFSET1484B33200LENGTH				
TH100681800	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5580FFSET1984C81800LENGTH				
TH4ABAA05A00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5590FFSET3200537A00LENGTH				
728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5590FFSET7E0000LENGTHH80				
80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5590FFSET80F10C00LENGTHH				
H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5590FFSET101641800LENGTH				
H6007CFE00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5590FFSET181D72400LENGTH				
H38033CA00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5590FFSETB0288E800LENGTH				
H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		

STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5590FFSETB82FBF400LENGT
H100681800
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5590FFSETC83648A00LENGT
H80728E00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5590FFSETD03D79600LENGT
H80728E00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5590FFSETD844AA200LENGT
H700681200
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5590FFSET1484B33200LENG
TH500146800
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5590FFSET1984C81800LENG
TH100681800
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATUREA5812ED90FFSET7E00LENGTH2711
5F800
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATUREA5812ED90FFSET271167600LENGT
H2F7BA0B400
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURE329D832C0FFSET7E00LENGTH746A
386A00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F56A0FFSET3200537A00LENG
TH4ABAA05A00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F56A0FFSET7E0000LENGTH80
728E00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F56A0FFSET80F10C00LENGTH
80728E00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F56A0FFSET101641800LENGT
H80728E00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F56A0FFSET181D72400LENGT
H6007CFE00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F56A0FFSET78254A000LENGT
H38033CA00

Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F56A0FFSETB0288E800LENGT
H80728E00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F56A0FFSETB82FBF400LENGT
H100681800
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F56A0FFSETC83648A00LENGT
H80728E00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F56A0FFSETD03D79600LENGT
H80728E00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F56A0FFSETD844AA200LENGT
H700681200
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F56A0FFSET1484B33200LENG
TH500146800
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F56A0FFSET1984C81800LENG
TH100681800
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F56B0FFSET3200537A00LENG
TH4ABAA05A00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F56B0FFSET7E0000LENGTH80
728E00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F56B0FFSET80F10C00LENGTH
80728E00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F56B0FFSET101641800LENGT
H80728E00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F56B0FFSET181D72400LENGT
H6007CFE00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F56B0FFSET78254A000LENGT
H38033CA00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F56B0FFSETB0288E800LENGT
H80728E00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available

H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F56C0FFSETD03D79600LENGTH				
H700681200	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F56C0FFSETD844AA200LENGTH				
TH500146800	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F56C0FFSET1484B33200LENGTH				
TH100681800	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F56D0FFSET3200537A00LENGTH				
TH4ABAA05A00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F56D0FFSET7E0000LENGTHH80728E00				
80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F56D0FFSET80F10C00LENGTHH80728E00				
80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F56D0FFSET101641800LENGTHH80728E00				
H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F56D0FFSET181D72400LENGTHH6007CFE00				
H6007CFE00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F56D0FFSET78254A000LENGTHH38033CA00				
H38033CA00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F56D0FFSETB0288E800LENGTHH80728E00				
H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F56D0FFSETB82FBF400LENGTHH100681800				
H100681800	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F56D0FFSETC83648A00LENGTHH80728E00				
H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F56D0FFSETD03D79600LENGTHH80728E00				
H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		

Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5560FFSET1984C81800LENG					
TH100681800					
Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5570FFSET3200537A00LENG					
TH4ABAA05A00					
Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5570FFSET7E0000LENGTH80					
728E00					
Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5570FFSET80F10C00LENGTH					
80728E00					
Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5570FFSET101641800LENGT					
H80728E00					
Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5570FFSET181D72400LENGT					
H6007CFE00					
Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5570FFSET78254A000LENGT					
H38033CA00					
Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5570FFSETB0288E800LENGT					
H80728E00					
Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5570FFSETB82FBF400LENGT					
H100681800					
Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5570FFSETC83648A00LENGT					
H80728E00					
Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5570FFSETD03D79600LENGT					
H80728E00					
Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5570FFSETD844AA200LENGT					
H700681200					
Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5570FFSET1484B33200LENG					
TH500146800					
Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5570FFSET1984C81800LENG					
TH100681800					
Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		

Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf	Not Available		
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5690FFSET80F10C00LENGTH				
80728E00				
Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf	Not Available		
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5690FFSET101641800LENGT				
H80728E00				
Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf	Not Available		
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5690FFSET181D72400LENGT				
H6007CFE00				
Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf	Not Available		
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5690FFSET78254A000LENGT				
H38033CA00				
Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf	Not Available		
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5690FFSETB0288E800LENGT				
H80728E00				
Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf	Not Available		
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5690FFSETB82FBF400LENGT				
H100681800				
Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf	Not Available		
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5690FFSETC83648A00LENGT				
H80728E00				
Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf	Not Available		
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5690FFSETD03D79600LENGT				
H80728E00				
Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf	Not Available		
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5690FFSETD844AA200LENGT				
H700681200				
Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf	Not Available		
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5690FFSET1484B33200LENG				
TH500146800				
Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf	Not Available		
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5690FFSET1984C81800LENG				
TH100681800				
Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf	Not Available		
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5520FFSET3200537A00LENG				
TH4ABAA05A00				
Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf	Not Available		
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5520FFSET7E0000LENGTHH80				
728E00				
Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf	Not Available		
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5520FFSET80F10C00LENGTHH				
80728E00				
Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf	Not Available		

H38033CA00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5530FFSET78254A000				LENGTH
H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5530FFSETB0288E800				LENGTH
H100681800	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5530FFSETB82FBF400				LENGTH
H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5530FFSETD03D79600				LENGTH
H700681200	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5530FFSETD844AA200				LENGTH
TH500146800	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5530FFSET1484B33200				LENGTH
TH100681800	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5530FFSET1984C81800				LENGTH
TH4ABAA05A00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5540FFSET7E0000				LENGTH
80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5540FFSET80F10C00				LENGTH
H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5540FFSET101641800				LENGTH
H6007CFE00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5540FFSET181D72400				LENGTH
H38033CA00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		

H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5550FFSETC83648A00LENGTH				
H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5550FFSETD03D79600LENGTH				
H700681200	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5550FFSETD844AA200LENGTH				
TH500146800	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5550FFSET1984C81800LENGTH				
TH100681800	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55E0FFSET3200537A00LENGTH				
TH4ABAA05A00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55E0FFSET7E0000LENGTHH80				
728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55E0FFSET80F10C00LENGTHH				
80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55E0FFSET101641800LENGTH				
H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55E0FFSET181D72400LENGTH				
H6007CFE00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55E0FFSET78254A000LENGTH				
H38033CA00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55E0FFSETB0288E800LENGTH				
H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55E0FFSETB82FBF400LENGTH				
H100681800	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55E0FFSETC83648A00LENGTH				
H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		

STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55E0FFSETD03D79600LENGT
H80728E00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55E0FFSETD844AA200LENGT
H700681200
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55E0FFSET1484B33200LENG
TH500146800
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55E0FFSET1484B33200LENG
TH100681800
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55E0FFSET1984C81800LENG
TH4ABAA05A00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55F0FFSET3200537A00LENG
728E00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55F0FFSET7E0000LENGTH80
80728E00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55F0FFSET80F10C00LENGTH
H80728E00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55F0FFSET101641800LENGT
H6007CFE00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55F0FFSETB0288E800LENGT
H38033CA00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55F0FFSETB82FBF400LENGT
H100681800
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55F0FFSETC83648A00LENGT
H80728E00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55F0FFSETD03D79600LENGT
H700681200
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55F0FFSETD844AA200LENGT

Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55F0FFSET1484B33200LENG
TH500146800
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55F0FFSET1984C81800LENG
TH100681800
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5500FFSET3200537A00LENG
TH4ABAA05A00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5500FFSET7E0000LENGTH80
728E00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5500FFSET80F10C00LENGTH
80728E00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5500FFSET101641800LENGT
H80728E00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5500FFSET181D72400LENGT
H6007CFE00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5500FFSET78254A000LENGT
H38033CA00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5500FFSETB0288E800LENGT
H80728E00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5500FFSETB82FBF400LENGT
H100681800
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5500FFSETC83648A00LENGT
H80728E00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5500FFSETD03D79600LENGT
H80728E00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5500FFSETD844AA200LENGT
H700681200
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5500FFSET1484B33200LENG
TH500146800
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available

728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55A0FFSET7E0000				LENGTH80
80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55A0FFSET80F10C00				LENGTH
H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55A0FFSET101641800				LENGTH
H6007CFE00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55A0FFSET78254A000				LENGTH
H38033CA00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55A0FFSETB0288E800				LENGTH
H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55A0FFSETB82FBF400				LENGTH
H100681800	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55A0FFSETC83648A00				LENGTH
H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55A0FFSETD03D79600				LENGTH
H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55A0FFSETD844AA200				LENGTH
H700681200	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55A0FFSET1484B33200				LENGTH
TH500146800	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55A0FFSET1984C81800				LENGTH
TH100681800	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55B0FFSET3200537A00				LENGTH
TH4ABAA05A00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55B0FFSET7E0000				LENGTH80
728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		

H6007CFE00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55C0FFSET181D72400				LENGTH
H38033CA00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55C0FFSET78254A000				LENGTH
H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55C0FFSETB0288E800				LENGTH
H100681800	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55C0FFSETC83648A00				LENGTH
H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55C0FFSETD03D79600				LENGTH
H700681200	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55C0FFSETD844AA200				LENGTH
TH500146800	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55C0FFSET1484B33200				LENGTH
TH100681800	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55D0FFSET3200537A00				LENGTH
728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55D0FFSET7E00000				LENGTH
80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55D0FFSET80F10C000				LENGTH
H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55D0FFSET101641800				LENGTH
H6007CFE00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		

STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55D0FFSET78254A000LENGT
H38033CA00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55D0FFSETB0288E800LENGT
H80728E00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55D0FFSETB82FBF400LENGT
H100681800
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55D0FFSETC83648A00LENGT
H80728E00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55D0FFSETD03D79600LENGT
H700681200
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55D0FFSETD844AA200LENGT
TH500146800
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55D0FFSETD844B33200LENG
TH100681800
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F55D0FFSET1984C81800LENG
TH4ABAA05A00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54D0FFSET3200537A00LENG
80728E00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54D0FFSET7E0000LENGTH80
H80728E00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54D0FFSET101641800LENGT
H6007CFE00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54D0FFSET78254A000LENGT
H38033CA00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54D0FFSETB0288E800LENGT
H80728E00

Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54D0FFSETB82FBF400LENGT
H100681800
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54D0FFSETC83648A00LENGT
H80728E00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54D0FFSETD03D79600LENGT
H80728E00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54D0FFSETD844AA200LENGT
H700681200
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54D0FFSET1484B33200LENG
TH500146800
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54D0FFSET1984C81800LENG
TH100681800
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54E0FFSET3200537A00LENG
TH4ABAA05A00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54E0FFSET7E0000LENGTH80
728E00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54E0FFSET80F10C00LENGTH
80728E00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54E0FFSET101641800LENGT
H80728E00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54E0FFSET181D72400LENGT
H6007CFE00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54E0FFSET78254A000LENGT
H38033CA00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54E0FFSETB0288E800LENGT
H80728E00
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54E0FFSETB82FBF400LENGT
H100681800
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available

H700681200	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54F0FFSETD844AA200				LENGTH
TH500146800	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F54F0FFSET1484B33200				LENGTH
TH100681800	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5410FFSET1984C81800				LENGTH
TH4ABAA05A00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5410FFSET7E0000				LENGTH
728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5410FFSET80F10C00				LENGTH
80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5410FFSET101641800				LENGTH
H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5410FFSET181D72400				LENGTH
H6007CFE00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5410FFSET78254A000				LENGTH
H38033CA00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5410FFSETB0288E800				LENGTH
H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5410FFSETB82FBF400				LENGTH
H100681800	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5410FFSETC83648A00				LENGTH
H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5410FFSETD03D79600				LENGTH
H80728E00	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		
	STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5410FFSETD844AA200				LENGTH
H700681200	Generic volume	Yes	VOLUME 5.2.3790.0	10/1/2002	Microsoft
	volume.inf		Not Available		

STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5410FFSET1484B33200LENG
TH500146800
Generic volume Yes VOLUME 5.2.3790.0 10/1/2002 Microsoft
volume.inf Not Available
STORAGE\VOLUME\1&30A96598&0&SIGNATURED9C8F5410FFSET1984C81800LENG
TH100681800
AFD Networking Support Environment Not Available LEGACYDRIVER Not
Available Not Available Not Available Not Available Not
Available ROOT\LEGACY_AFD\0000
Beep Not Available LEGACYDRIVER Not Available Not Available Not
Available Not Available Not Available ROOT\LEGACY_BEEP\0000
CRC Disk Filter Driver Not Available LEGACYDRIVER Not Available Not
Available Not Available Not Available Not Available
ROOT\LEGACY_CRCDISK\0000
dmbboot Not Available LEGACYDRIVER Not Available Not Available Not
Available Not Available Not Available ROOT\LEGACY_DMBOOT\0000

dmload Not Available LEGACYDRIVER Not Available Not Available Not
Available Not Available Not Available ROOT\LEGACY_DMLOAD\0000

em Not Available LEGACYDRIVER Not Available Not Available Not
Available Not Available Not Available ROOT\LEGACY_EM\0000
Fips Not Available LEGACYDRIVER Not Available Not Available Not
Available Not Available Not Available ROOT\LEGACY_FIPS\0000
Generic Packet Classifier Not Available LEGACYDRIVER Not
Available Not Available Not Available Not Available Not
Available ROOT\LEGACY_GPC\0000
IPSEC driver Not Available LEGACYDRIVER Not Available Not
Available Not Available Not Available Not Available
ROOT\LEGACY_IPSEC\0000
ksecdd Not Available LEGACYDRIVER Not Available Not Available Not
Available Not Available Not Available ROOT\LEGACY_KSECDD\0000

mountmgr Not Available LEGACYDRIVER Not Available Not
Available Not Available Not Available Not Available
ROOT\LEGACY_MOUNTMGR\0000
NDIS System Driver Not Available LEGACYDRIVER Not Available Not
Available Not Available Not Available Not Available
ROOT\LEGACY_NDIS\0000
Remote Access NDIS TAPI Driver Not Available LEGACYDRIVER Not
Available Not Available Not Available Not Available Not
Available ROOT\LEGACY_NDISTAPI\0000
NDIS Usermode I/O Protocol Not Available LEGACYDRIVER Not
Available Not Available Not Available Not Available Not
Available ROOT\LEGACY_NDISUIO\0000
NDProxy Not Available LEGACYDRIVER Not Available Not Available Not
Available Not Available Not Available ROOT\LEGACY_NDPROXY\0000

NetBios over Tcpip Not Available LEGACYDRIVER Not Available Not
Available Not Available Not Available Not Available
ROOT\LEGACY_NETBT\0000
Null Not Available LEGACYDRIVER Not Available Not Available Not
Available Not Available Not Available ROOT\LEGACY_NULL\0000
Partition Manager Not Available LEGACYDRIVER Not Available Not
Available Not Available Not Available Not Available
ROOT\LEGACY_PARTMGR\0000
qldirect Not Available LEGACYDRIVER Not Available Not
Available Not Available Not Available Not Available
ROOT\LEGACY_QLDIRECT\0000

Remote Access Auto Connection Driver Not Available LEGACYDRIVER Not
Available Not Available Not Available Not Available Not
Available ROOT\LEGACY_RASACD\0000
RDPDCC Not Available LEGACYDRIVER Not Available Not Available Not
Available Not Available Not Available ROOT\LEGACY_RDPDCC\0000

RDPWD Not Available LEGACYDRIVER Not Available Not Available Not
Available Not Available Not Available ROOT\LEGACY_RDPWD\0000
sacdrv Not Available LEGACYDRIVER Not Available Not Available Not
Available Not Available Not Available ROOT\LEGACY_SACDRV\0000

tbs Not Available LEGACYDRIVER Not Available Not Available Not
Available Not Available Not Available ROOT\LEGACY_TBS\0000
TCP/IP Protocol Driver Not Available LEGACYDRIVER Not Available Not
Available Not Available Not Available Not Available
ROOT\LEGACY_TCPIP\0000
TDTCP Not Available LEGACYDRIVER Not Available Not Available Not
Available Not Available Not Available ROOT\LEGACY_TDTCP\0000
Vapamap Not Available LEGACYDRIVER Not Available Not Available Not
Available Not Available Not Available ROOT\LEGACY_VAPAMAP\0000

volsnap Not Available LEGACYDRIVER Not Available Not Available Not
Available Not Available Not Available ROOT\LEGACY_VOLSNAP\0000

Remote Access IP ARP Driver Not Available LEGACYDRIVER Not
Available Not Available Not Available Not Available Not
Available ROOT\LEGACY_WANARP\0000
Audio Codecs Yes MEDIA 5.2.3790.0 10/1/2002 (Standard
system devices) wave.inf Not Available ROOT\MEDIA\MS_MMCM

Legacy Audio Drivers Yes MEDIA 5.2.3790.0 10/1/2002
(Standard system devices) wave.inf Not Available
ROOT\MEDIA\MS_MMDRV
Media Control Devices Yes MEDIA 5.2.3790.0 10/1/2002
(Standard system devices) wave.inf Not Available
ROOT\MEDIA\MS_MMMCI
Legacy Video Capture Devices Yes MEDIA 5.2.3790.0 10/1/2002
(Standard system devices) wave.inf Not Available
ROOT\MEDIA\MS_MMVCD
Video Codecs Yes MEDIA 5.2.3790.0 10/1/2002 (Standard
system devices) wave.inf Not Available ROOT\MEDIA\MS_MMVID

WAN Miniport (L2TP) Yes NET 5.2.3790.0 10/1/2002
Microsoft netrasa.inf Not Available
ROOT\MS_L2TPMINI\PORT\0000
WAN Miniport (IP) Yes NET 5.2.3790.0 10/1/2002
Microsoft netrasa.inf Not Available
ROOT\MS_NDISWANIP\0000
WAN Miniport (PPPOE) Yes NET 5.2.3790.0 10/1/2002
Microsoft netrasa.inf Not Available
ROOT\MS_PPPOEMINI\PORT\0000
WAN Miniport (PPTP) Yes NET 5.2.3790.0 10/1/2002
Microsoft netrasa.inf Not Available
ROOT\MS_PPTPMINI\PORT\0000
Direct Parallel Yes NET 5.2.3790.0 10/1/2002
Microsoft netrasa.inf Not Available
ROOT\MS_PTMINI\PORT\0000

```

Terminal Server Device Redirector    Yes    SYSTEM  5.2.3790.0
10/1/2002    (Standard system devices)    machine.inf    Not
Available    ROOT\RDPCR\0000
Terminal Server Keyboard Driver      Yes    SYSTEM  5.2.3790.0
10/1/2002    (Standard system devices)    machine.inf    Not
Available    ROOT\RDP_KBD\0000
Terminal Server Mouse Driver Yes      SYSTEM  5.2.3790.0    10/1/2002
(Standard system devices)    machine.inf    Not Available
ROOT\RDP_MOUSE\0000
Plug and Play Software Device Enumerator    Yes    SYSTEM  5.2.3790.0
10/1/2002    (Standard system devices)    machine.inf    Not
Available    ROOT\SYSTEM\0000

```

[Environment Variables]

```

Variable      Value      User Name
Basemake      C:\Program Files\Microsoft SDK\Include\BKOffice.Mak
              <SYSTEM>
Bkoffice      C:\Program Files\Microsoft SDK\    <SYSTEM>
CLASSPATH
              .;C:\SQLLIB\java\db2java.zip;C:\SQLLIB\java\db2jcc.jar;C:\SQLLIB\
java\sqlj.zip;C:\SQLLIB\java\db2jcc_license_cisuz.jar;C:\SQLLIB\java\db2
jcc_license_cu.jar;C:\SQLLIB\bin;C:\SQLLIB\java\common.jar <SYSTEM>

ClusterLog    C:\WINDOWS\Cluster\cluster.log    <SYSTEM>
ComSpec %SystemRoot%\system32\cmd.exe <SYSTEM>
DB2INSTANCE   DB2    <SYSTEM>
DB2TMPDIR     C:\SQLLIB\    <SYSTEM>
INCLUDE C:\SQLLIB\INCLUDE;C:\SQLLIB\LIB;C:\Program Files\Microsoft
SDK\Include\prerelease;C:\Program Files\Microsoft
SDK\Include\Win64\crt;C:\Program Files\Microsoft
SDK\Include\Win64\crt\sys;C:\Program Files\Microsoft
SDK\Include\Win64\mfc;C:\Program Files\Microsoft
SDK\Include\Win64\atl;C:\Program Files\Microsoft SDK\Include
              <SYSTEM>
INETSDK C:\Program Files\Microsoft SDK\    <SYSTEM>
LIB           C:\Program Files\Microsoft SDK\Lib\Prerelease\IA64;C:\Program
Files\Microsoft SDK\Lib\IA64;C:\Program Files\Microsoft
SDK\Lib\IA64\mfc;C:\SQLLIB\LIB    <SYSTEM>
MSSdk C:\Program Files\Microsoft SDK\    <SYSTEM>
Mstools C:\Program Files\Microsoft SDK\    <SYSTEM>
NUMBER_OF_PROCESSORS 8    <SYSTEM>
OS            Windows_NT    <SYSTEM>
Path
              C:\Perl\bin\;%SystemRoot%\system32;%SystemRoot%;%SystemRoot%\Syst
em32\Wbem;C:\SQLLIB\BIN;C:\SQLLIB\FUNCTION;C:\SQLLIB\SAMPLES\REPL;C:\Pro
gram Files\Microsoft SDK\Bin\.;C:\Program Files\Microsoft
SDK\Bin\WinNT\.;C:\Program Files\Microsoft SDK\Bin\Win64    <SYSTEM>

PATHEXT .COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH    <SYSTEM>

PROCESSOR_ARCHITECTURE IA64    <SYSTEM>
PROCESSOR_IDENTIFIER   ia64 Family 31 Model 2 Stepping 1, GenuineIntel
              <SYSTEM>
PROCESSOR_LEVEL        31    <SYSTEM>
PROCESSOR_REVISION     0201    <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
Basemake      C:\Program Files\Microsoft SDK\Include\BKOffice.Mak
              TIGER\nec
Bkoffice      C:\Program Files\Microsoft SDK\    TIGER\nec
INCLUDE C:\SQLLIB\INCLUDE;C:\SQLLIB\LIB;C:\Program Files\Microsoft
SDK\Include\prerelease;C:\Program Files\Microsoft
SDK\Include\Win64\crt;C:\Program Files\Microsoft
SDK\Include\Win64\crt\sys;C:\Program Files\Microsoft
SDK\Include\Win64\mfc;C:\Program Files\Microsoft
SDK\Include\Win64\atl;C:\Program Files\Microsoft SDK\Include
              TIGER\nec
INETSDK C:\Program Files\Microsoft SDK\    TIGER\nec
LIB           C:\Program Files\Microsoft SDK\Lib\Prerelease\IA64;C:\Program
Files\Microsoft SDK\Lib\IA64;C:\Program Files\Microsoft
SDK\Lib\IA64\mfc;C:\SQLLIB\LIB    TIGER\nec
MSSdk C:\Program Files\Microsoft SDK\    TIGER\nec
Mstools C:\Program Files\Microsoft SDK\    TIGER\nec
PATH
              C:\Perl\bin\;%SystemRoot%\system32;%SystemRoot%;%SystemRoot%\Syst
em32\Wbem;C:\SQLLIB\BIN;C:\SQLLIB\FUNCTION;C:\SQLLIB\SAMPLES\REPL;C:\Pro
gram Files\Microsoft SDK\Bin\.;C:\Program Files\Microsoft
SDK\Bin\WinNT\.;C:\Program Files\Microsoft SDK\Bin\Win64    TIGER\nec

TEMP %USERPROFILE%\Local Settings\Temp    TIGER\nec
TMP %USERPROFILE%\Local Settings\Temp    TIGER\nec

```

[Print Jobs]

Document	Size	Owner	Notify	Status	Time Submitted	Start Time
	Until Time	Elapsed Time	Pages	Printed	Job ID	Priority
	Parameters	Driver	Print	Processor	Host	Print Queue
	Data Type	Name				

[Network Connections]

Local Name	Remote Name	Type	Status	User Name
X:	\\10.1.1.48\c\$	Disk	Persistent Connection	TIGER\nec
Y:	\\10.1.1.47\c\$	Disk	Persistent Connection	TIGER\nec
Z:	\\10.1.1.200\c\$	Disk	Persistent Connection	

[Running Tasks]

Name	Path	Process ID	Priority	Min Working Set	Max
Working Set	Start Time	Version Size	File Date		
system idle process	Not Available	0	0	Not Available	Not
Available	Not Available	Not Available	Not Available	Not	
Available	Not Available	4	8	0	
system Not Available	Not Available	Not Available	Not Available	2826240	Not Available
Available	Not Available	Not Available	Not Available		Not

```

smss.exe      Not Available  480      11      409600 2826240 12/24/2004
11:18 AM      Not Available  Not Available  Not Available
csrss.exe     Not Available  528      13      Not Available  Not
Available     12/24/2004 11:18 AM  Not Available  Not Available  Not
Available
winlogon.exe  c:\windows\system32\winlogon.exe  552      13
409600 2826240 12/24/2004 11:18 AM  5.2.3790.0
(srv03_rtm.030324-2048)  618.00 KB (632,832 bytes)  3/25/2003
09:00 PM
services.exe  c:\windows\system32\services.exe  596      9
409600 2826240 12/24/2004 11:18 AM  5.2.3790.0
(srv03_rtm.030324-2048)  286.00 KB (292,864 bytes)  3/25/2003
09:00 PM
lsass.exe     c:\windows\system32\lsass.exe  608      9      409600
2826240 12/24/2004 11:18 AM  5.2.3790.0 (srv03_rtm.030324-2048)
15.00 KB (15,360 bytes)  3/25/2003 09:00 PM
svchost.exe   c:\windows\system32\svchost.exe  780      8
409600 2826240 12/24/2004 11:19 AM  5.2.3790.0
(srv03_rtm.030324-2048)  32.50 KB (33,280 bytes)  3/25/2003
09:00 PM
svchost.exe   c:\windows\system32\svchost.exe  828      8
409600 2826240 12/24/2004 11:19 AM  5.2.3790.0
(srv03_rtm.030324-2048)  32.50 KB (33,280 bytes)  3/25/2003
09:00 PM
svchost.exe   Not Available  980      8      Not Available  Not
Available     12/24/2004 11:19 AM  Not Available  Not Available  Not
svchost.exe   Not Available  1028     8      Not Available  Not
Available     12/24/2004 11:19 AM  Not Available  Not Available  Not
Available
svchost.exe   c:\windows\system32\svchost.exe  1040     8
409600 2826240 12/24/2004 11:19 AM  5.2.3790.0
(srv03_rtm.030324-2048)  32.50 KB (33,280 bytes)  3/25/2003
09:00 PM
spoolsv.exe   c:\windows\system32\spoolsv.exe  1236     8
409600 2826240 12/24/2004 11:19 AM  5.2.3790.0
(srv03_rtm.030324-2048)  155.50 KB (159,232 bytes)  3/25/2003
09:00 PM
msdtc.exe     Not Available  1264     8      Not Available  Not
Available     12/24/2004 11:19 AM  Not Available  Not Available  Not
Available
db2dasrrm.exe Not Available  1520     8      Not Available  Not
Available     12/24/2004 11:19 AM  Not Available  Not Available  Not
Available
db2licd.exe   c:\sqlllib\bin\db2licd.exe  1536     8      409600
2826240 12/24/2004 11:19 AM  8.1.9.715 128.50 KB (131,584
bytes) 11/26/2004 05:19 PM
db2sec.exe    c:\sqlllib\bin\db2sec.exe  1560     8      409600
2826240 12/24/2004 11:19 AM  8.1.9.715 15.00 KB (15,360
bytes) 11/26/2004 05:19 PM
db2rcmd.exe   Not Available  1572     8      Not Available  Not
Available     12/24/2004 11:19 AM  Not Available  Not Available  Not
Available
svchost.exe   c:\windows\system32\svchost.exe  1596     8
409600 2826240 12/24/2004 11:19 AM  5.2.3790.0
(srv03_rtm.030324-2048)  32.50 KB (33,280 bytes)  3/25/2003
09:00 PM

```

```

svchost.exe   Not Available  1640     8      Not Available  Not
Available     12/24/2004 11:19 AM  Not Available  Not Available  Not
Available
db2jds.exe    c:\sqlllib\bin\db2jds.exe  1876     8      409600
2826240 12/24/2004 11:19 AM  8.1.9.715 422.50 KB (432,640
bytes) 11/26/2004 05:19 PM
dfssvc.exe    c:\windows\system32\dfssvc.exe  1900     8
409600 2826240 12/24/2004 11:19 AM  5.2.3790.0
(srv03_rtm.030324-2048)  442.00 KB (452,608 bytes)  3/25/2003
09:00 PM
wmiprvse.exe  Not Available  264      8      Not Available  Not
Available     12/24/2004 11:20 AM  Not Available  Not Available  Not
Available
explorer.exe   c:\windows\explorer.exe  852      8      409600
2826240 12/24/2004 11:22 AM  6.00.3790.0 (srv03_rtm.030324-2048)
1.63 MB (1,704,960 bytes)  3/25/2003 09:00 PM
svchost.exe   c:\windows\system32\svchost.exe  312      8
409600 2826240 12/24/2004 11:22 AM  5.2.3790.0
(srv03_rtm.030324-2048)  32.50 KB (33,280 bytes)  3/25/2003
09:00 PM
mmc.exe        c:\windows\system32\mmc.exe  2080     8      409600 2826240
12/24/2004 11:23 AM  5.2.3790.0 (srv03_rtm.030324-2048)  2.48
MB (2,598,912 bytes)  3/25/2003 09:00 PM
vds.exe        c:\windows\system32\vds.exe  2124     8      409600 2826240
12/24/2004 11:23 AM  5.2.3790.0 (srv03_rtm.030324-2048)
543.50 KB (556,544 bytes)  3/25/2003 09:00 PM
dmadmin.exe    c:\windows\system32\dmadmin.exe  2180     8
409600 2826240 12/24/2004 11:23 AM  5.2.3790.0
(srv03_rtm.030324-2048)  643.50 KB (658,944 bytes)  3/25/2003
09:00 PM
helpctr.exe    c:\windows\pchealth\helpctr\binaries\helpctr.exe  2452
8      409600 2826240 12/24/2004 11:33 AM  5.2.3790.0
(srv03_rtm.030324-2048)  1.97 MB (2,066,432 bytes)  10/15/2004
02:43 PM
wmiprvse.exe  Not Available  2508     8      Not Available  Not
Available     12/24/2004 11:33 AM  Not Available  Not Available  Not
Available
helpsvc.exe    c:\windows\pchealth\helpctr\binaries\helpsvc.exe  2548
8      409600 2826240 12/24/2004 11:33 AM  5.2.3790.0
(srv03_rtm.030324-2048)  2.18 MB (2,289,152 bytes)  10/15/2004
02:43 PM

```

[Loaded Modules]

Name	Version	Size	File Date	Manufacturer	Path
winlogon	5.2.3790.0	(srv03_rtm.030324-2048)	618.00 KB (632,832 bytes)	Microsoft Corporation	
ntdll	5.2.3790.0	(srv03_rtm.030324-2048)	1.45 MB (1,524,224 bytes)	Microsoft Corporation	
kernel32	5.2.3790.0	(srv03_rtm.030324-2048)	1.76 MB (1,850,368 bytes)	Microsoft Corporation	
msvcrt	7.0.3790.0	(srv03_rtm.030324-2048)	873.50 KB (894,464 bytes)	Microsoft Corporation	

advapi32 5.2.3790.0 (srv03_rtm.030324-2048) 1.32 MB (1,383,424 bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\advapi32.dll

rpcrt4 5.2.3790.0 (srv03_rtm.030324-2048) 2.03 MB (2,127,872 bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\rpcrt4.dll

user32 5.2.3790.0 (srv03_rtm.030324-2048) 1.31 MB (1,372,672 bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\user32.dll

gdi32 5.2.3790.0 (srv03_rtm.030324-2048) 783.00 KB (801,792 bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\gdi32.dll

userenv 5.2.3790.0 (srv03_rtm.030324-2048) 1.46 MB (1,536,000 bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\userenv.dll

nddeapi 5.2.3790.0 (srv03_rtm.030324-2048) 39.50 KB (40,448 bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\nddeapi.dll

crypt32 5.131.3790.0 (srv03_rtm.030324-2048) 1.50 MB (1,576,448 bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\crypt32.dll

msasn1 5.2.3790.0 (srv03_rtm.030324-2048) 153.50 KB (157,184 bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\msasn1.dll

secur32 5.2.3790.0 (srv03_rtm.030324-2048) 166.50 KB (170,496 bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\secur32.dll

winsta 5.2.3790.0 (srv03_rtm.030324-2048) 138.50 KB (141,824 bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\winsta.dll

netapi32 5.2.3790.0 (srv03_rtm.030324-2048) 832.00 KB (851,968 bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\netapi32.dll

profmap 5.2.3790.0 (srv03_rtm.030324-2048) 55.50 KB (56,832 bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\profmap.dll

regapi 5.2.3790.0 (srv03_rtm.030324-2048) 124.00 KB (126,976 bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\regapi.dll

ws2_32 5.2.3790.0 (srv03_rtm.030324-2048) 228.50 KB (233,984 bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\ws2_32.dll

ws2help 5.2.3790.0 (srv03_rtm.030324-2048) 49.50 KB (50,688 bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\ws2help.dll

msgina 5.2.3790.0 (srv03_rtm.030324-2048) 1.35 MB (1,417,728 bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\msgina.dll

shsvcs 6.00.3790.0 (srv03_rtm.030324-2048) 321.50 KB (329,216 bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\shsvcs.dll

shlwapi 6.00.3790.0 (srv03_rtm.030324-2048) 722.00 KB (739,328 bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\shlwapi.dll

sfc 5.2.3790.0 (srv03_rtm.030324-2048) 7.50 KB (7,680 bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\sfc.dll

sfc_os 5.2.3790.0 (srv03_rtm.030324-2048) 257.00 KB (263,168 bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\sfc_os.dll

wintrust 5.131.3790.0 (srv03_rtm.030324-2048) 451.50 KB (462,336 bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\wintrust.dll

ole32 5.2.3790.0 (srv03_rtm.030324-2048) 3.38 MB (3,549,184 bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\ole32.dll

imagehlp 5.2.3790.0 (srv03_rtm.030324-2048) 128.50 KB (131,584 bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\imagehlp.dll

comctl32 6.0 (srv03_rtm.030324-2048) 2.18 MB (2,285,056 bytes) 10/15/2004 07:16 AM Microsoft Corporation
c:\windows\winsxs\ia64_microsoft.windows.common-controls_6595b64144ccf1df_6.0.100.0_x-ww_b3722bab\comctl32.dll

version 5.2.3790.0 (srv03_rtm.030324-2048) 44.00 KB (45,056 bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\version.dll

winscard 5.2.3790.0 (srv03_rtm.030324-2048) 291.50 KB (298,496 bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\winscard.dll

wsapi32 5.2.3790.0 (srv03_rtm.030324-2048) 47.50 KB (48,640 bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\wsapi32.dll

sxs 5.2.3790.0 (srv03_rtm.030324-2048) 1.77 MB (1,860,608 bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\sxs.dll

winmm 5.2.3790.0 (srv03_rtm.030324-2048) 404.00 KB (413,696 bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\winmm.dll

shell32 6.00.3790.0 (srv03_rtm.030324-2048) 12.35 MB (12,953,088 bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\shell32.dll

setupapi 5.2.3790.0 (srv03_rtm.030324-2048) 1.90 MB (1,991,168 bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\setupapi.dll

wldap32 5.2.3790.0 (srv03_rtm.030324-2048) 412.50 KB (422,400 bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\wldap32.dll

cscdll 5.2.3790.0 (srv03_rtm.030324-2048) 211.00 KB (216,064 bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\cscdll.dll

wlnotify 5.2.3790.0 (srv03_rtm.030324-2048) 218.00 KB (223,232 bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\wlnotify.dll

winspool 5.2.3790.0 (srv03_rtm.030324-2048) 399.50 KB (409,088 bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\winspool.drv

mpr 5.2.3790.0 (srv03_rtm.030324-2048) 163.00 KB (166,912 bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\mpr.dll

comctl32 5.82 (srv03_rtm.030324-2048) 1.55 MB (1,621,504 bytes) 10/15/2004 07:16 AM Microsoft Corporation
c:\windows\winsxs\ia64_microsoft.windows.common-controls_6595b64144ccf1df_5.82.0.0_x-ww_b9c4a0a5\comctl32.dll

uxtheme 6.00.3790.0 (srv03_rtm.030324-2048) 527.50 KB (540,160 bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\uxtheme.dll

samlib 5.2.3790.0 (srv03_rtm.030324-2048) 106.00 KB (108,544 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\samlib.dll

cscui 5.2.3790.0 (srv03_rtm.030324-2048) 574.00 KB (587,776 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\cscui.dll

drprov 5.2.3790.0 (srv03_rtm.030324-2048) 26.50 KB (27,136 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\drprov.dll

ntlanman 5.2.3790.0 (srv03_rtm.030324-2048) 108.00 KB (110,592 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\ntlanman.dll

netui0 5.2.3790.0 (srv03_rtm.030324-2048) 181.50 KB (185,856 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\netui0.dll

netui1 5.2.3790.0 (srv03_rtm.030324-2048) 482.00 KB (493,568 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\netui1.dll

davclnt 5.2.3790.0 (srv03_rtm.030324-2048) 59.00 KB (60,416 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\davclnt.dll

mprui 5.2.3790.0 (srv03_rtm.030324-2048) 97.00 KB (99,328 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\mprui.dll

netui2 5.2.3790.0 (srv03_rtm.030324-2048) 761.50 KB (779,776 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\netui2.dll

comdlg32 6.00.3790.0 (srv03_rtm.030324-2048) 706.00 KB (722,944 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\comdlg32.dll

netmsg 5.2.3790.0 (srv03_rtm.030324-2048) 177.50 KB (181,760 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\netmsg.dll

oleaut32 5.2.3790.0 3.57 MB (3,739,136 bytes) 3/25/2003 09:00 PM Microsoft Corporation c:\windows\system32\oleaut32.dll

clbcatq 2001.12.4720.0 (srv03_rtm.030324-2048) 1.23 MB (1,292,800 bytes)
 10/15/2004 02:41 PM Microsoft Corporation
 c:\windows\system32\clbcatq.dll

comres 2001.12.4720.0 (srv03_rtm.030324-2048) 779.50 KB (798,208 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\comres.dll

ntmarta 5.2.3790.0 (srv03_rtm.030324-2048) 343.50 KB (351,744 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\ntmarta.dll

services 5.2.3790.0 (srv03_rtm.030324-2048) 286.00 KB (292,864 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\services.exe

scserv 5.2.3790.0 (srv03_rtm.030324-2048) 765.00 KB (783,360 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\scserv.dll

authz 5.2.3790.0 (srv03_rtm.030324-2048) 202.50 KB (207,360 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\authz.dll

umpnpmgr 5.2.3790.0 (srv03_rtm.030324-2048) 314.50 KB (322,048 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\umpnpmgr.dll

ncobjapi 5.2.3790.0 (srv03_rtm.030324-2048) 118.50 KB (121,344 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\ncobjapi.dll

msvcp60 6.10.2240.8 941.50 KB (964,096 bytes) 3/25/2003 09:00 PM Microsoft Corporation c:\windows\system32\msvcp60.dll

eventlog 5.2.3790.0 (srv03_rtm.030324-2048) 157.00 KB (160,768 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\eventlog.dll

psapi 5.2.3790.0 (srv03_rtm.030324-2048) 48.00 KB (49,152 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\psapi.dll

lsass 5.2.3790.0 (srv03_rtm.030324-2048) 15.00 KB (15,360 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\lsass.exe

lsasrv 5.2.3790.0 (srv03_rtm.030324-2048) 1.94 MB (2,033,664 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\lsasrv.dll

samsrv 5.2.3790.0 (srv03_rtm.030324-2048) 1,005.00 KB (1,029,120 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\samsrv.dll

cryptdll 5.2.3790.0 (srv03_rtm.030324-2048) 61.00 KB (62,464 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\cryptdll.dll

dnsapi 5.2.3790.0 (srv03_rtm.030324-2048) 404.00 KB (413,696 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\dnsapi.dll

ntdsapi 5.2.3790.0 (srv03_rtm.030324-2048) 181.50 KB (185,856 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\ntdsapi.dll

msprivs 5.2.3790.0 (srv03_rtm.030324-2048) 46.00 KB (47,104 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\msprivs.dll

kerberos 5.2.3790.0 (srv03_rtm.030324-2048) 876.00 KB (897,024 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\kerberos.dll

msv1_0 5.2.3790.0 (srv03_rtm.030324-2048) 333.50 KB (341,504 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\msv1_0.dll

netlogon 5.2.3790.0 (srv03_rtm.030324-2048) 936.50 KB (958,976 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\netlogon.dll

w32time 5.2.3790.0 (srv03_rtm.030324-2048) 540.50 KB (553,472 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\w32time.dll

iphlpapi 5.2.3790.0 (srv03_rtm.030324-2048) 223.00 KB (228,352 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\iphlpapi.dll

schannel 5.2.3790.0 (srv03_rtm.030324-2048) 468.50 KB (479,744 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\schannel.dll

wdigest 5.2.3790.0 (srv03_rtm.030324-2048) 161.50 KB (165,376 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\wdigest.dll

rsaenh 5.2.3790.0 (srv03_rtm.030324-2048) 371.83 KB (380,752 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\rsaenh.dll

rassfm 5.2.3790.0 (srv03_rtm.030324-2048) 56.00 KB (57,344 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\rassfm.dll

kdcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 571.50 KB (585,216 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\kdcsvc.dll
 ntdsa 5.2.3790.0 (srv03_rtm.030324-2048) 3.82 MB (4,008,448 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\ntdsa.dll
 ntdsatq 5.2.3790.0 (srv03_rtm.030324-2048) 80.50 KB (82,432 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\ntdsatq.dll
 mssock 5.2.3790.0 (srv03_rtm.030324-2048) 671.00 KB (687,104 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\mssock.dll
 esent 5.2.3790.0 (srv03_rtm.030324-2048) 2.48 MB (2,605,056 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\esent.dll
 scecli 5.2.3790.0 (srv03_rtm.030324-2048) 467.50 KB (478,720 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\scecli.dll
 wshtcpip 5.2.3790.0 (srv03_rtm.030324-2048) 38.00 KB (38,912 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\wshtcpip.dll
 ipsecsvc 5.2.3790.0 (srv03_rtm.030324-2048) 410.50 KB (420,352 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\ipsecsvc.dll
 oakley 5.2.3790.0 (srv03_rtm.030324-2048) 493.50 KB (505,344 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\oakley.dll
 winipsec 5.2.3790.0 (srv03_rtm.030324-2048) 78.50 KB (80,384 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\winipsec.dll
 pstorsvc 5.2.3790.0 (srv03_rtm.030324-2048) 56.00 KB (57,344 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\pstorsvc.dll
 psbase 5.2.3790.0 (srv03_rtm.030324-2048) 162.50 KB (166,400 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\psbase.dll
 dssenh 5.2.3790.0 (srv03_rtm.030324-2048) 319.33 KB (326,992 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\dssenh.dll
 wlsctrl 5.2.3790.0 (srv03_rtm.030324-2048) 194.50 KB (199,168 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\wlsctrl.dll
 svchost 5.2.3790.0 (srv03_rtm.030324-2048) 32.50 KB (33,280 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\svchost.exe
 rpcss 5.2.3790.0 (srv03_rtm.030324-2048) 645.00 KB (660,480 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\rpcss.dll
 termsrv 5.2.3790.0 (srv03_rtm.030324-2048) 607.00 KB (621,568 bytes)
 10/15/2004 02:41 PM Microsoft Corporation
 c:\windows\system32\termsrv.dll
 icaapi 5.2.3790.0 (srv03_rtm.030324-2048) 27.00 KB (27,648 bytes)
 10/15/2004 02:41 PM Microsoft Corporation
 c:\windows\system32\icaapi.dll
 mstlsapi 5.2.3790.0 (srv03_rtm.030324-2048) 311.00 KB (318,464 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\mstlsapi.dll

activeds 5.2.3790.0 (srv03_rtm.030324-2048) 543.50 KB (556,544 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\activeds.dll
 adslldpc 5.2.3790.0 (srv03_rtm.030324-2048) 312.00 KB (319,488 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\adslldpc.dll
 credui 5.2.3790.0 (srv03_rtm.030324-2048) 288.00 KB (294,912 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\credui.dll
 atl 3.00.2282 348.00 KB (356,352 bytes) 3/25/2003 09:00 PM
 Microsoft Corporation c:\windows\system32\atl.dll
 rdpwsx 5.2.3790.0 (srv03_rtm.030324-2048) 256.63 KB (262,792 bytes)
 10/15/2004 02:41 PM Microsoft Corporation
 c:\windows\system32\rdpwsx.dll
 wzcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 608.50 KB (623,104 bytes)
 3/25/2003 10:19 PM Microsoft Corporation
 c:\windows\system32\wzcsvc.dll
 rtutils 5.2.3790.0 (srv03_rtm.030324-2048) 81.50 KB (83,456 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\rtutils.dll
 wmi 5.2.3790.0 (srv03_rtm.030324-2048) 5.00 KB (5,120 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\wmi.dll
 dhcpcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 279.50 KB (286,208 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\dhcpcsvc.dll
 rastls 5.2.3790.0 (srv03_rtm.030324-2048) 346.00 KB (354,304 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\rastls.dll
 cryptui 5.131.3790.0 (srv03_rtm.030324-2048) 1.04 MB (1,094,656 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\cryptui.dll
 mprapi 5.2.3790.0 (srv03_rtm.030324-2048) 238.00 KB (243,712 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\mprapi.dll
 rasapi32 5.2.3790.0 (srv03_rtm.030324-2048) 589.50 KB (603,648 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\rasapi32.dll
 rasman 5.2.3790.0 (srv03_rtm.030324-2048) 154.50 KB (158,208 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\rasman.dll
 tapi32 5.2.3790.0 (srv03_rtm.030324-2048) 493.00 KB (504,832 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\tapi32.dll
 raschap 5.2.3790.0 (srv03_rtm.030324-2048) 200.00 KB (204,800 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\raschap.dll
 schedsvc 5.2.3790.0 (srv03_rtm.030324-2048) 527.50 KB (540,160 bytes)
 10/15/2004 02:43 PM Microsoft Corporation
 c:\windows\system32\schedsvc.dll
 msidle 6.00.3790.0 (srv03_rtm.030324-2048) 8.50 KB (8,704 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\msidle.dll
 wkssvc 5.2.3790.0 (srv03_rtm.030324-2048) 304.00 KB (311,296 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\wkssvc.dll
 cryptsvc 5.2.3790.0 (srv03_rtm.030324-2048) 126.00 KB (129,024 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\cryptsvc.dll

```

certcli 5.2.3790.0 (srv03_rtm.030324-2048) 586.00 KB (600,064 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\certcli.dll
vssapi 5.2.3790.0 (srv03_rtm.030324-2048) 1.28 MB (1,339,904 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\vssapi.dll
dmserver 5.2.3790.0 (srv03_rtm.030324-2048) 45.50 KB (46,592
bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\dmserver.dll
es 2001.12.4720.0 (srv03_rtm.030324-2048) 637.50 KB (652,800
bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\es.dll
pchsvc 5.2.3790.0 (srv03_rtm.030324-2048) 94.50 KB (96,768 bytes)
10/15/2004 02:43 PM Microsoft Corporation
c:\windows\pchealth\helpctr\binaries\pchsvc.dll
srvsvc 5.2.3790.0 (srv03_rtm.030324-2048) 188.00 KB (192,512 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\srvsvc.dll
sacsvr 5.2.3790.0 (srv03_rtm.030324-2048) 27.50 KB (28,160 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\sacsvr.dll
seclogon 5.2.3790.0 (srv03_rtm.030324-2048) 41.50 KB (42,496
bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\seclogon.dll
trkwks 5.2.3790.0 (srv03_rtm.030324-2048) 246.00 KB (251,904 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\trkwks.dll
wmisvc 5.2.3790.0 (srv03_rtm.030324-2048) 408.50 KB (418,304 bytes)
10/15/2004 02:40 PM Microsoft Corporation
c:\windows\system32\wbem\wmisvc.dll
wuauserv 5.4.3790.0 (srv03_rtm.030324-2048) 17.50 KB (17,920
bytes) 10/15/2004 02:41 PM Microsoft Corporation
c:\windows\system32\wuauserv.dll
wuaueng 5.4.3790.0 (srv03_rtm.030324-2048) 495.50 KB (507,392 bytes)
10/15/2004 02:41 PM Microsoft Corporation
c:\windows\system32\wuaueng.dll
advpack 6.00.3790.0 (srv03_rtm.030324-2048) 240.00 KB (245,760 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\advpack.dll
wininet 6.00.3790.0 (srv03_rtm.030324-2048) 1.43 MB (1,500,672 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\wininet.dll
sens 5.2.3790.0 (srv03_rtm.030324-2048) 90.50 KB (92,672 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\sens.dll
comsvcs 2001.12.4720.0 (srv03_rtm.030324-2048) 2.96 MB (3,106,816
bytes) 10/15/2004 02:41 PM Microsoft Corporation
c:\windows\system32\comsvcs.dll
browser 5.2.3790.0 (srv03_rtm.030324-2048) 187.00 KB (191,488 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\browser.dll
wiarpc 5.2.3790.0 (srv03_rtm.030324-2048) 67.50 KB (69,120 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\wiarpc.dll
netrap 5.2.3790.0 (srv03_rtm.030324-2048) 30.00 KB (30,720 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\netrap.dll

```

```

wbemcore 5.2.3790.0 (srv03_rtm.030324-2048) 1.64 MB (1,723,904
bytes) 10/15/2004 02:40 PM Microsoft Corporation
c:\windows\system32\wbem\wbemcore.dll
esscli 5.2.3790.0 (srv03_rtm.030324-2048) 919.50 KB (941,568 bytes)
10/15/2004 02:40 PM Microsoft Corporation
c:\windows\system32\wbem\esscli.dll
wbemcomn 5.2.3790.0 (srv03_rtm.030324-2048) 598.50 KB (612,864
bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\wbem\wbemcomn.dll
fastprox 5.2.3790.0 (srv03_rtm.030324-2048) 1.51 MB (1,580,544
bytes) 10/15/2004 02:40 PM Microsoft Corporation
c:\windows\system32\wbem\fastprox.dll
wmiutils 5.2.3790.0 (srv03_rtm.030324-2048) 285.00 KB (291,840
bytes) 10/15/2004 02:41 PM Microsoft Corporation
c:\windows\system32\wbem\wmiutils.dll
repdrvfs 5.2.3790.0 (srv03_rtm.030324-2048) 595.50 KB (609,792
bytes) 10/15/2004 02:40 PM Microsoft Corporation
c:\windows\system32\wbem\repdrvfs.dll
wmiprvsd 5.2.3790.0 (srv03_rtm.030324-2048) 1.38 MB (1,443,328
bytes) 10/15/2004 02:40 PM Microsoft Corporation
c:\windows\system32\wbem\wmiprvsd.dll
wbemess 5.2.3790.0 (srv03_rtm.030324-2048) 976.50 KB (999,936 bytes)
10/15/2004 02:40 PM Microsoft Corporation
c:\windows\system32\wbem\wbemess.dll
ncprov 5.2.3790.0 (srv03_rtm.030324-2048) 133.50 KB (136,704 bytes)
10/15/2004 02:40 PM Microsoft Corporation
c:\windows\system32\wbem\ncprov.dll
wbemsvc 5.2.3790.0 (srv03_rtm.030324-2048) 62.50 KB (64,000 bytes)
10/15/2004 02:40 PM Microsoft Corporation
c:\windows\system32\wbem\wbemsvc.dll
wbemcons 5.2.3790.0 (srv03_rtm.030324-2048) 237.50 KB (243,200
bytes) 10/15/2004 02:40 PM Microsoft Corporation
c:\windows\system32\wbem\wbemcons.dll
winhttp 5.2.3790.0 (srv03_rtm.030324-2048) 882.50 KB (903,680 bytes)
10/15/2004 07:17 AM Microsoft Corporation
c:\windows\winsxs\ia64_microsoft.windows.winhttp_6595b64144ccf1df
_5.1.0.0_x-ww_0fbfffd6\winhttp.dll
netman 5.2.3790.0 (srv03_rtm.030324-2048) 591.00 KB (605,184 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\netman.dll
wzcsapi 5.2.3790.0 (srv03_rtm.030324-2048) 49.50 KB (50,688 bytes)
3/25/2003 10:19 PM Microsoft Corporation
c:\windows\system32\wzcsapi.dll
netshell 5.2.3790.0 (srv03_rtm.030324-2048) 2.65 MB (2,779,648
bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\netshell.dll
clusapi 5.2.3790.0 (srv03_rtm.030324-2048) 165.50 KB (169,472 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\clusapi.dll
netcfgx 5.2.3790.0 (srv03_rtm.030324-2048) 1.94 MB (2,034,688 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\netcfgx.dll
hnetcfg 5.2.3790.0 (srv03_rtm.030324-2048) 764.00 KB (782,336 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\hnetcfg.dll
rasmans 5.2.3790.0 (srv03_rtm.030324-2048) 447.50 KB (458,240 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\rasmans.dll

```

rastapi	5.2.3790.0 (srv03_rtm.030324-2048)	138.50 KB (141,824 bytes)	
	3/25/2003 09:00 PM Microsoft Corporation		
	c:\windows\system32\rastapi.dll		
rasppp	5.2.3790.0 (srv03_rtm.030324-2048)	522.50 KB (535,040 bytes)	
	3/25/2003 09:00 PM Microsoft Corporation		
	c:\windows\system32\rasppp.dll		
ntlsapi	5.2.3790.0 (srv03_rtm.030324-2048)	14.50 KB (14,848 bytes)	
	3/25/2003 09:00 PM Microsoft Corporation		
	c:\windows\system32\ntlsapi.dll		
ipbootp	5.2.3790.0 (srv03_rtm.030324-2048)	72.50 KB (74,240 bytes)	
	3/25/2003 09:00 PM Microsoft Corporation		
	c:\windows\system32\ipbootp.dll		
rasdlg	5.2.3790.0 (srv03_rtm.030324-2048)	1.35 MB (1,420,800 bytes)	
	3/25/2003 09:00 PM Microsoft Corporation		
	c:\windows\system32\rasdlg.dll		
rasadhlp	5.2.3790.0 (srv03_rtm.030324-2048)	13.00 KB (13,312 bytes)	
	3/25/2003 09:00 PM Microsoft Corporation		
	c:\windows\system32\rasadhlp.dll		
spoolsv	5.2.3790.0 (srv03_rtm.030324-2048)	155.50 KB (159,232 bytes)	
	3/25/2003 09:00 PM Microsoft Corporation		
	c:\windows\system32\spoolsv.exe		
spoolss	5.2.3790.0 (srv03_rtm.030324-2048)	227.50 KB (232,960 bytes)	
	3/25/2003 09:00 PM Microsoft Corporation		
	c:\windows\system32\spoolss.dll		
localspl	5.2.3790.0 (srv03_rtm.030324-2048)	839.00 KB (859,136 bytes)	
	3/25/2003 09:00 PM Microsoft Corporation		
	c:\windows\system32\localspl.dll		
cnbjmon	5.2.3680.0 (Lab03_dev(skatar).020509-1043)	99.50 KB (101,888 bytes)	
	3/25/2003 05:50 PM Microsoft Corporation		
	c:\windows\system32\cnbjmon.dll		
pjlmon	5.2.3790.0 (srv03_rtm.030324-2048)	37.50 KB (38,400 bytes)	
	3/25/2003 05:52 PM Microsoft Corporation		
	c:\windows\system32\pjlmon.dll		
tcpmon	5.2.3790.0 (srv03_rtm.030324-2048)	132.50 KB (135,680 bytes)	
	3/25/2003 09:00 PM Microsoft Corporation		
	c:\windows\system32\tcpmon.dll		
mgmtapi	5.2.3790.0 (srv03_rtm.030324-2048)	40.50 KB (41,472 bytes)	
	3/25/2003 09:00 PM Microsoft Corporation		
	c:\windows\system32\mgmtapi.dll		
snmpapi	5.2.3790.0 (srv03_rtm.030324-2048)	52.00 KB (53,248 bytes)	
	3/25/2003 09:00 PM Microsoft Corporation		
	c:\windows\system32\snmpapi.dll		
wsnmp32	5.2.3790.0 (srv03_rtm.030324-2048)	111.50 KB (114,176 bytes)	
	3/25/2003 09:00 PM Microsoft Corporation		
	c:\windows\system32\wsnmp32.dll		
usbmon	5.2.3790.0 (srv03_rtm.030324-2048)	43.50 KB (44,544 bytes)	
	3/25/2003 09:00 PM Microsoft Corporation		
	c:\windows\system32\usbmon.dll		
winrrr	5.2.3790.0 (srv03_rtm.030324-2048)	39.00 KB (39,936 bytes)	
	3/25/2003 09:00 PM Microsoft Corporation		
	c:\windows\system32\winrrr.dll		
wshqos	5.2.3790.0 (srv03_rtm.030324-2048)	65.50 KB (67,072 bytes)	
	3/25/2003 09:00 PM Microsoft Corporation		
	c:\windows\system32\wshqos.dll		
win32spl	5.2.3790.0 (srv03_rtm.030324-2048)	269.00 KB (275,456 bytes)	
	3/25/2003 09:00 PM Microsoft Corporation		
	c:\windows\system32\win32spl.dll		
inetpp	5.2.3790.0 (srv03_rtm.030324-2048)	211.50 KB (216,576 bytes)	
	3/25/2003 09:00 PM Microsoft Corporation		
	c:\windows\system32\inetpp.dll		
icmp	5.2.3790.0 (srv03_rtm.030324-2048)	2.50 KB (2,560 bytes)	
	3/25/2003 09:00 PM Microsoft Corporation		
	c:\windows\system32\icmp.dll		
db2licd	8.1.9.715	128.50 KB (131,584 bytes)	11/26/2004 05:19 PM
	International Business Machines Corporation		
	c:\sql\bin\db2licd.exe		
db2osse64	8.1.9.715	2.29 MB (2,401,792 bytes)	11/26/2004 05:19 PM
	International Business Machines Corporation		
	c:\sql\bin\db2osse64.dll		
db2sys64	8.1.9.715	2.53 MB (2,652,672 bytes)	11/26/2004 05:19 PM
	International Business Machines Corporation		
	c:\sql\bin\db2sys64.dll		
db2wint64	8.1.9.715	21.00 KB (21,504 bytes)	11/26/2004 05:19 PM
	International Business Machines Corporation		
	c:\sql\bin\db2wint64.dll		
db2sysp64	8.1.9.715	248.00 KB (253,952 bytes)	11/26/2004 05:19 PM
	International Business Machines Corporation		
	c:\sql\bin\db2sysp64.dll		
db2app64	8.1.9.715	12.68 MB (13,296,128 bytes)	11/26/2004 05:17 PM
	International Business Machines Corporation		
	c:\sql\bin\db2app64.dll		
db2locale64	8.1.9.715	56.50 KB (57,856 bytes)	11/26/2004 05:19 PM
	International Business Machines Corporation		
	c:\sql\bin\db2locale64.dll		
db2g11n64	8.1.9.715	603.50 KB (617,984 bytes)	11/26/2004 05:19 PM
	International Business Machines Corporation		
	c:\sql\bin\db2g11n64.dll		
db2genreg64	8.1.9.715	322.50 KB (330,240 bytes)	11/26/2004 05:19 PM
	International Business Machines Corporation		
	c:\sql\bin\db2genreg64.dll		
db2install64	8.1.9.715	16.00 KB (16,384 bytes)	11/26/2004 05:19 PM
	International Business Machines Corporation		
	c:\sql\bin\db2install64.dll		
db2dascmn64	8.1.9.715	152.00 KB (155,648 bytes)	11/26/2004 05:18 PM
	International Business Machines Corporation		
	c:\sql\bin\db2dascmn64.dll		
db2osse_db264	8.1.9.715	320.50 KB (328,192 bytes)	11/26/2004 05:19 PM
	International Business Machines Corporation		
	c:\sql\bin\db2osse_db264.dll		
db2sec64	8.1.9.715	20.50 KB (20,992 bytes)	11/26/2004 05:19 PM
	International Business Machines Corporation		
	c:\sql\bin\db2sec64.dll		
db2trcapi64	8.1.9.715	45.00 KB (46,080 bytes)	11/26/2004 05:19 PM
	International Business Machines Corporation		
	c:\sql\bin\db2trcapi64.dll		
db2sec	8.1.9.715	15.00 KB (15,360 bytes)	11/26/2004 05:19 PM
	International Business Machines Corporation		
	c:\sql\bin\db2sec.exe		
ersvc	5.2.3790.0 (srv03_rtm.030324-2048)	61.00 KB (62,464 bytes)	
	3/25/2003 09:00 PM Microsoft Corporation		
	c:\windows\system32\ersvc.dll		
db2jds	8.1.9.715	422.50 KB (432,640 bytes)	11/26/2004 05:19 PM
	International Business Machines Corporation		
	c:\sql\bin\db2jds.exe		

```

dfssvc 5.2.3790.0 (srv03_rtm.030324-2048) 442.00 KB (452,608 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\dfssvc.exe
resutils 5.2.3790.0 (srv03_rtm.030324-2048) 147.50 KB (151,040
bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\resutils.dll
mfc42u 6.00.3014.0 3.34 MB (3,506,176 bytes) 3/25/2003 09:00 PM
Microsoft Corporation c:\windows\system32\mfc42u.dll
wsock32 5.2.3790.0 (srv03_rtm.030324-2048) 23.00 KB (23,552 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\wsock32.dll
explorer 6.00.3790.0 (srv03_rtm.030324-2048) 1.63 MB (1,704,960
bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\explorer.exe
browseui 6.00.3790.0 (srv03_rtm.030324-2048) 2.42 MB (2,536,960
bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\browseui.dll
shdocvw 6.00.3790.0 (srv03_rtm.030324-2048) 3.20 MB (3,359,744 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\shdocvw.dll
apphelp 5.2.3790.0 (srv03_rtm.030324-2048) 262.50 KB (268,800 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\apphelp.dll
themeui 6.00.3790.0 (srv03_rtm.030324-2048) 823.00 KB (842,752 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\themeui.dll
msimg32 5.2.3790.0 (srv03_rtm.030324-2048) 7.00 KB (7,168 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\msimg32.dll
actxprxy 6.00.3790.0 (srv03_rtm.030324-2048) 230.00 KB (235,520
bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\actxprxy.dll
linkinfo 5.2.3790.0 (srv03_rtm.030324-2048) 42.00 KB (43,008
bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\linkinfo.dll
ntshrui 6.00.3790.0 (srv03_rtm.030324-2048) 233.50 KB (239,104 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\ntshrui.dll
webcheck 6.00.3790.0 (srv03_rtm.030324-2048) 664.50 KB (680,448
bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\webcheck.dll
stobject 5.2.3790.0 (srv03_rtm.030324-2048) 178.00 KB (182,272
bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\stobject.dll
batmeter 6.00.3790.0 (srv03_rtm.030324-2048) 55.50 KB (56,832
bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\batmeter.dll
powrprof 6.00.3790.0 (srv03_rtm.030324-2048) 36.00 KB (36,864
bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\powrprof.dll
printui 5.2.3790.0 (srv03_rtm.030324-2048) 1.09 MB (1,142,784 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\printui.dll
cfgmgr32 5.2.3790.0 (srv03_rtm.030324-2048) 16.00 KB (16,384
bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\cfgmgr32.dll
tapisrv 5.2.3790.0 (srv03_rtm.030324-2048) 641.50 KB (656,896 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\tapisrv.dll

```

```

unimdm 5.2.3790.0 (srv03_rtm.030324-2048) 417.50 KB (427,520 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\unimdm.tsp
uniplat 5.2.3790.0 (srv03_rtm.030324-2048) 35.50 KB (36,352 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\uniplat.dll
kmddsp 5.2.3790.0 (srv03_rtm.030324-2048) 84.00 KB (86,016 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\kmddsp.tsp
ndptsp 5.2.3790.0 (srv03_rtm.030324-2048) 128.00 KB (131,072 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\ndptsp.tsp
ipconf 5.2.3790.0 (srv03_rtm.030324-2048) 40.50 KB (41,472 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\ipconf.tsp
h323 5.2.3790.0 (srv03_rtm.030324-2048) 797.00 KB (816,128 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\h323.tsp
hidphone 5.2.3790.0 (srv03_rtm.030324-2048) 87.00 KB (89,088
bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\hidphone.tsp
hid 5.2.3790.0 (srv03_rtm.030324-2048) 44.00 KB (45,056 bytes)
3/25/2003 10:17 PM Microsoft Corporation
c:\windows\system32\hid.dll
mmc 5.2.3790.0 (srv03_rtm.030324-2048) 2.48 MB (2,598,912 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\mmc.exe
oleacc 4.2.5406.0 (srv03_rtm.030324-2048) 485.00 KB (496,640 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\oleacc.dll
mmcbase 5.2.3790.0 (srv03_rtm.030324-2048) 139.00 KB (142,336 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\mmcbase.dll
mmcndmgr 5.2.3790.0 (srv03_rtm.030324-2048) 3.10 MB (3,248,640
bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\mmcndmgr.dll
urlmon 6.00.3790.0 (srv03_rtm.030324-2048) 1.21 MB (1,271,296 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\urlmon.dll
msxml3 8.40.9419.0 3.42 MB (3,590,656 bytes) 3/25/2003 09:00 PM
Microsoft Corporation c:\windows\system32\msxml3.dll
dmdskmgr 5.2.3790.0 (srv03_rtm.030324-2048) 515.00 KB (527,360
bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\dmdskmgr.dll
ntmsapi 5.2.3790.0 (srv03_rtm.030324-2048) 118.00 KB (120,832 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\ntmsapi.dll
dmutil 5.2.3790.0 (srv03_rtm.030324-2048) 71.50 KB (73,216 bytes)
3/25/2003 05:50 PM Microsoft Corporation
c:\windows\system32\dmutil.dll
dmdskres 5.2.3790.0 (srv03_rtm.030324-2048) 115.00 KB (117,760
bytes) 3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\dmdskres.dll
dmdlgs 5.2.3790.0 (srv03_rtm.030324-2048) 670.50 KB (686,592 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\dmdlgs.dll
dmview 5.2.3790.0 (srv03_rtm.030324-2048) 207.00 KB (211,968 bytes)
3/25/2003 09:00 PM Microsoft Corporation
c:\windows\system32\dmview.ocx

```

vds_ps 5.2.3790.0 (srv03_rtm.030324-2048) 31.00 KB (31,744 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\vds_ps.dll

dmvdsitf 5.2.3790.0 (srv03_rtm.030324-2048) 321.00 KB (328,704 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\dmvdsitf.dll

mlang 6.00.3790.0 (srv03_rtm.030324-2048) 799.00 KB (818,176 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\mlang.dll

vds 5.2.3790.0 (srv03_rtm.030324-2048) 543.50 KB (556,544 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\vds.exe

osuninst 5.2.3790.0 (srv03_rtm.030324-2048) 6.50 KB (6,656 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\osuninst.dll

vdsutil 5.2.3790.0 (srv03_rtm.030324-2048) 98.00 KB (100,352 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\vdsutil.dll

vdsbas 5.2.3790.0 (srv03_rtm.030324-2048) 321.50 KB (329,216 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\vdsbas.dll

fmifs 5.2.3790.0 (srv03_rtm.030324-2048) 53.00 KB (54,272 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\fmifs.dll

ulib 5.2.3790.0 (srv03_rtm.030324-2048) 448.50 KB (459,264 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\ulib.dll

ifsutil 5.2.3790.0 (srv03_rtm.030324-2048) 214.00 KB (219,136 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\ifsutil.dll

vdsdyndr 5.2.3790.0 (srv03_rtm.030324-2048) 710.50 KB (727,552 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\vdsdyndr.dll

dmintf 5.2.3790.0 (srv03_rtm.030324-2048) 25.00 KB (25,600 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\dmintf.dll

dmadmin 5.2.3790.0 (srv03_rtm.030324-2048) 643.50 KB (658,944 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\dmadmin.exe

helpctr 5.2.3790.0 (srv03_rtm.030324-2048) 1.97 MB (2,066,432 bytes)
 10/15/2004 02:43 PM Microsoft Corporation
 c:\windows\pchealth\helpctr\binaries\helpctr.exe

hcappres 5.2.3790.0 (srv03_rtm.030324-2048) 6.00 KB (6,144 bytes)
 10/15/2004 02:43 PM Microsoft Corporation
 c:\windows\pchealth\helpctr\binaries\hcappres.dll

itss 5.2.3790.0 (srv03_rtm.030324-2048) 349.00 KB (357,376 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\itss.dll

pchshell 5.2.3790.0 (srv03_rtm.030324-2048) 277.50 KB (284,160 bytes)
 10/15/2004 02:43 PM Microsoft Corporation
 c:\windows\pchealth\helpctr\binaries\pchshell.dll

mshtml 6.00.3790.0 (srv03_rtm.030324-2048) 7.83 MB (8,208,384 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\mshtml.dll

msimtf 5.2.3790.0 (srv03_rtm.030324-2048) 528.00 KB (540,672 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\msimtf.dll

msctf 5.2.3790.0 (srv03_rtm.030324-2048) 924.50 KB (946,688 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\msctf.dll

shdoclc 6.00.3790.0 (srv03_rtm.030324-2048) 588.00 KB (602,112 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\shdoclc.dll

jscript 5.6.0.8515 1.21 MB (1,268,736 bytes) 3/25/2003 09:00 PM
 Microsoft Corporation c:\windows\system32\jscript.dll

msls31 3.10.349.0 448.00 KB (458,752 bytes) 3/25/2003 09:00 PM
 Microsoft Corporation c:\windows\system32\msls31.dll

imm32 5.2.3790.0 (srv03_rtm.030324-2048) 307.50 KB (314,880 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\imm32.dll

mshtml 6.00.3790.0 (srv03_rtm.030324-2048) 1.34 MB (1,408,512 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\mshtml.dll

vbscript 5.6.0.8515 1.06 MB (1,110,016 bytes) 3/25/2003 09:00 PM
 Microsoft Corporation c:\windows\system32\vbscript.dll

mfc42 6.00.3014.0 3.36 MB (3,526,656 bytes) 3/25/2003 09:00 PM
 Microsoft Corporation c:\windows\system32\mfc42.dll

msinfo 5.2.3790.0 (srv03_rtm.030324-2048) 1.20 MB (1,257,984 bytes)
 10/15/2004 02:43 PM Microsoft Corporation
 c:\windows\pchealth\helpctr\binaries\msinfo.dll

riched32 5.2.3790.0 (srv03_rtm.030324-2048) 5.00 KB (5,120 bytes)
 3/25/2003 09:00 PM Microsoft Corporation
 c:\windows\system32\riched32.dll

riched20 5.31.23.1218 1.25 MB (1,313,280 bytes) 3/25/2003 09:00 PM
 Microsoft Corporation c:\windows\system32\riched20.dll

wbemprox 5.2.3790.0 (srv03_rtm.030324-2048) 46.00 KB (47,104 bytes)
 10/15/2004 02:40 PM Microsoft Corporation
 c:\windows\system32\wbem\wbemprox.dll

helpsvc 5.2.3790.0 (srv03_rtm.030324-2048) 2.18 MB (2,289,152 bytes)
 10/15/2004 02:43 PM Microsoft Corporation
 c:\windows\pchealth\helpctr\binaries\helpsvc.exe

[Services]

Display Name	Name	State	Start Mode	Service Type	Path	Error
Control	Start Name	Tag ID				
Alert	Alert	Stopped	Disabled	Share Process		
	c:\windows\system32\svchost.exe -k localservice				Normal	NT
Application Layer Gateway Service		ALG	Stopped	Manual	Own Process	
	c:\windows\system32\alg.exe	Normal	NT	AUTHORITY\LocalService		
Application Management AppMgmt		Stopped	Manual	Share Process		
	c:\windows\system32\svchost.exe -k netsvcs				Normal	LocalSystem
Windows Audio	AudioSrv	Stopped	Disabled	Share Process		
	c:\windows\system32\svchost.exe -k netsvcs				Normal	LocalSystem
Background Intelligent Transfer Service		BITS	Stopped	Manual	Share Process	
	c:\windows\system32\svchost.exe -k netsvcs				Normal	LocalSystem
Computer Browser		Browser	Running	Auto	Share Process	
	c:\windows\system32\svchost.exe -k netsvcs				Normal	LocalSystem

Indexing Service	CiSvc	Stopped	Disabled	Share Process
c:\windows\system32\cisvc.exe Normal LocalSystem 0				
ClipBook	ClipSrv	Stopped	Disabled	Own Process
c:\windows\system32\clipsrv.exe Normal LocalSystem 0				
COM+ System Application	COMSysApp	Stopped	Manual	Own Process
c:\windows\system32\dlhhost.exe /processid:{02d4b3f1-fd88-11d1-960d-00805fc79235} Normal LocalSystem 0				
Cryptographic Services	CryptSvc	Running	Auto	Share Process
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0				
DB2 - DB2-0	DB2-0	Stopped	Manual	Own Process
c:\sqllib\bin\db2syscs.exe Normal .\nec 0				
DB2 - DB2CTLSV-0	DB2CTLSV-0	Stopped	Manual	Own Process
c:\sqllib\bin\db2syscs.exe Normal .\nec 0				
DB2DAS - DB2DAS00	DB2DAS00	Running	Auto	Own Process
"c:\sqllib\bin\db2dasrrm.exe" Normal .\nec 0				
DB2 Governor	DB2GOVERNOR	Stopped	Manual	Own Process
"c:\sqllib\bin\db2govds.exe" Normal .\nec 0				
DB2 JDBC Applet Server	DB2JDS	Running	Auto	Own Process
"c:\sqllib\bin\db2jds.exe" Normal LocalSystem 0				
DB2 License Server	DB2LICD	Running	Auto	Own Process
c:\sqllib\bin\db2licd.exe Ignore LocalSystem 0				
DB2 Registry Reflector	DB2NTREGREFLECTOR	Stopped	Auto	Own Process
"c:\sqllib\bin\db2reg64.exe" Normal .\nec 0				
DB2 Security Server	DB2NTSECSERVER	Running	Auto	Own Process
"c:\sqllib\bin\db2sec.exe" Normal LocalSystem 0				
DB2 Remote Command Server	DB2REMOTECMD	Running	Auto	Own Process
"c:\sqllib\bin\db2rcmd.exe" Ignore .\nec 0				
Distributed File System	Dfs	Running	Auto	Own Process
c:\windows\system32\dfsrv.exe Normal LocalSystem 0				
DHCP Client	Dhcp	Running	Auto	Share Process
c:\windows\system32\svchost.exe -k networkservice Normal NT AUTHORITY\NetworkService 0				
Logical Disk Manager Administrative Service	dmadmin	Running	Manual	Share Process
c:\windows\system32\dmadmin.exe /com Normal LocalSystem 0				
Logical Disk Manager	dmserver	Running	Auto	Share Process
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0				
DNS Client	Dnscache	Running	Auto	Share Process
c:\windows\system32\svchost.exe -k networkservice Normal NT AUTHORITY\NetworkService 0				
Error Reporting Service	ERSvc	Running	Auto	Share Process
c:\windows\system32\svchost.exe -k winerr Ignore LocalSystem 0				
Event Log	Eventlog	Running	Auto	Share Process
c:\windows\system32\services.exe Normal LocalSystem 0				
COM+ Event System	EventSystem	Running	Manual	Share Process
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0				
Help and Support	helpsvc	Running	Auto	Share Process
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0				
Human Interface Device Access	HidServ	Stopped	Disabled	Share Process
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0				

HTTP SSL	HTTPFilter	Stopped	Manual	Share Process
c:\windows\system32\lsass.exe Normal LocalSystem 0				
IAS Jet Database Access	IASJet	Stopped	Manual	Share Process
c:\windows\syswow64\svchost.exe -k iasjet Normal LocalSystem 0				
IMAPI CD-Burning COM Service	ImapiService	Stopped	Disabled	Own Process
c:\windows\system32\imapi.exe Normal LocalSystem 0				
Inter-site Messaging	IsmSrv	Stopped	Disabled	Own Process
c:\windows\system32\ismserv.exe Normal LocalSystem 0				
Kerberos Key Distribution Center	kdc	Stopped	Disabled	Share Process
c:\windows\system32\lsass.exe Normal LocalSystem 0				
Server lanmanserver	Running	Auto	Share Process	
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0				
Workstation	lanmanworkstation	Running	Auto	Share Process
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0				
License Logging	LicenseService	Stopped	Disabled	Own Process
c:\windows\system32\llssrv.exe Normal NT AUTHORITY\NetworkService 0				
TCP/IP NetBIOS Helper	LmHosts	Running	Auto	Share Process
c:\windows\system32\svchost.exe -k localservice Normal NT AUTHORITY\LocalService 0				
Messenger	Messenger	Stopped	Disabled	Share Process
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0				
Distributed Transaction Coordinator	MSDTC	Running	Auto	Own Process
c:\windows\system32\msdtc.exe Normal NT AUTHORITY\NetworkService 0				
Windows Installer	MSIServer	Stopped	Manual	Share Process
c:\windows\system32\msiexec.exe /v Normal LocalSystem 0				
DB2 - NEC	NEC	Stopped	Manual	Own Process
c:\sqllib\bin\db2syscs.exe Normal LocalSystem 0				
Network DDE	NetDDE	Stopped	Disabled	Share Process
c:\windows\system32\netdde.exe Normal LocalSystem 0				
Network DDE DSDM	NetDDEdsdm	Stopped	Disabled	Share Process
c:\windows\system32\netdde.exe Normal LocalSystem 0				
Net Logon	Netlogon	Stopped	Manual	Share Process
c:\windows\system32\lsass.exe Normal LocalSystem 0				
Network Connections	Netman	Running	Manual	Share Process
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0				
Network Location Awareness (NLA)	Nla	Running	Manual	Share Process
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0				
File Replication	NtFrs	Stopped	Manual	Own Process
c:\windows\system32\ntfrs.exe Ignore LocalSystem 0				
NT LM Security Support Provider	NtLmSsp	Stopped	Manual	Share Process
c:\windows\system32\lsass.exe Normal LocalSystem 0				
Removable Storage	NtmsSvc	Stopped	Manual	Share Process
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0				
Plug and Play	PlugPlay	Running	Auto	Share Process
c:\windows\system32\services.exe Normal LocalSystem 0				

IPSEC Services PolicyAgent	Running Auto	Share Process		
c:\windows\system32\lsass.exe	Normal	LocalSystem	0	
Protected Storage ProtectedStorage	Running Auto	Share		
Process c:\windows\system32\lsass.exe	Normal	LocalSystem	0	
Remote Access Auto Connection Manager RasAuto	Stopped Manual	Share		
Process c:\windows\system32\svchost.exe -k netsvcs	Normal	LocalSystem	0	
Remote Access Connection Manager RasMan	Running Manual	Share		
Process c:\windows\system32\svchost.exe -k netsvcs	Normal	LocalSystem	0	
Remote Desktop Help Session Manager RDSessMgr	Stopped Manual	Own		
Process c:\windows\system32\sessmgr.exe	Normal	LocalSystem	0	
Routing and Remote Access RemoteAccess	Stopped Disabled	Share		
Process c:\windows\system32\svchost.exe -k netsvcs	Normal	LocalSystem	0	
Remote Registry RemoteRegistry	Running Auto	Share Process		
c:\windows\system32\svchost.exe -k regsvc	Normal	NT		
AUTHORITY\LocalService 0				
Remote Procedure Call (RPC) Locator RpcLocator	Stopped Manual	Own		
Process c:\windows\system32\locator.exe	Normal	NT		
AUTHORITY\NetworkService 0				
Remote Procedure Call (RPC) RpcSs	Running Auto	Share Process		
c:\windows\system32\svchost -k rpcss	Normal	LocalSystem	0	
Resultant Set of Policy Provider RSOPProv	Stopped Manual	Share		
Process c:\windows\system32\rsopprov.exe	Normal	LocalSystem	0	
Special Administration Console Helper sacsvr	Running Manual	Share		
Process c:\windows\system32\svchost.exe -k netsvcs	Normal	LocalSystem	0	
Security Accounts Manager SamSs	Running Auto	Share Process		
c:\windows\system32\lsass.exe	Normal	LocalSystem	0	
Smart Card SCardSvr	Stopped Manual	Share Process		
c:\windows\system32\scardsvr.exe	Ignore	NT		
AUTHORITY\LocalService 0				
Task Scheduler Schedule	Running Auto	Share Process		
c:\windows\system32\svchost.exe -k netsvcs	Normal	LocalSystem	0	
Secondary Logon seclogon	Running Auto	Share Process		
c:\windows\system32\svchost.exe -k netsvcs	Ignore	LocalSystem	0	
System Event Notification SENS	Running Auto	Share Process		
c:\windows\system32\svchost.exe -k netsvcs	Normal	LocalSystem	0	
Shell Hardware Detection ShellHWDetection	Running Auto	Share		
Process c:\windows\system32\svchost.exe -k netsvcs	Ignore	LocalSystem	0	
Print Spooler Spooler	Running Auto	Own Process		
c:\windows\system32\spoolsv.exe	Normal	LocalSystem	0	
Windows Image Acquisition (WIA) stisvc	Stopped Disabled	Share		
Process c:\windows\system32\svchost.exe -k imgsvc	Normal	NT		
AUTHORITY\LocalService 0				
Microsoft Software Shadow Copy Provider swprv	Stopped Manual	Own		
Process c:\windows\system32\svchost.exe -k swprv	Normal	LocalSystem	0	

Performance Logs and Alerts SysmonLog	Stopped Manual	Own Process
c:\windows\system32\smlogsvc.exe	Normal	NT
Authority\NetworkService 0		
Telephony Tapisrv	Running Manual	Share Process
c:\windows\system32\svchost.exe -k tapisrv	Normal	LocalSystem
Terminal Services TermService	Running Manual	Share Process
c:\windows\system32\svchost.exe -k termsvcs	Normal	LocalSystem
Telnet TlntSvr	Stopped Disabled	Own Process
c:\windows\system32\tlntsvr.exe	Normal	NT
AUTHORITY\LocalService 0		
Distributed Link Tracking Server TrkSvr	Stopped Disabled	Share
Process c:\windows\system32\svchost.exe -k netsvcs	Normal	LocalSystem
Distributed Link Tracking Client TrkWks	Running Auto	Share
Process c:\windows\system32\svchost.exe -k netsvcs	Normal	LocalSystem
Terminal Services Session Directory Tssdis	Stopped Disabled	Own
Process c:\windows\system32\tssdis.exe	Normal	LocalSystem
Upload Manager uploadmgr	Stopped Disabled	Share Process
c:\windows\system32\svchost.exe -k netsvcs	Normal	LocalSystem
Uninterruptible Power Supply UPS	Stopped Manual	Own Process
c:\windows\system32\ups.exe	Normal	NT AUTHORITY\LocalService
Virtual Disk Service vds	Running Manual	Own Process
c:\windows\system32\vds.exe	Normal	LocalSystem
Volume Shadow Copy VSS	Stopped Manual	Own Process
c:\windows\system32\vssvc.exe	Normal	LocalSystem
Windows Time W32Time	Running Auto	Share Process
c:\windows\system32\svchost.exe -k netsvcs	Normal	LocalSystem
WebClient WebClient	Stopped Disabled	Share Process
c:\windows\system32\svchost.exe -k localservice	Normal	NT
AUTHORITY\LocalService 0		
WinHTTP Web Proxy Auto-Discovery Service WinHttpAutoProxySvc	Stopped Manual	Share Process
c:\windows\system32\svchost.exe -k localservice	Normal	NT AUTHORITY\LocalService
Windows Management Instrumentation winmgmt	Running Auto	Share
Process c:\windows\system32\svchost.exe -k netsvcs	Ignore	LocalSystem
Windows Management Instrumentation Driver Extensions Wmi	Stopped	
Manual Share Process c:\windows\system32\svchost.exe -k netsvcs	Normal	LocalSystem
WMI Performance Adapter WmiApSrv	Stopped Manual	Own Process
c:\windows\system32\wbem\wmiapsrv.exe	Normal	LocalSystem
Automatic Updates wuauerv	Running Auto	Share Process
c:\windows\system32\svchost.exe -k netsvcs	Normal	LocalSystem
Wireless Configuration WZCSVC	Running Auto	Share Process
c:\windows\system32\svchost.exe -k netsvcs	Normal	LocalSystem

[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	ActivePerl 5.8 All Users
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 Platform SDK February 2003	All Users:Microsoft Platform SDK February 2003	All Users	
Microsoft Platform SDK February 2003\Open Build Environment Window	All Users:Microsoft Platform SDK February 2003\Open Build Environment Window	All Users	
Microsoft Platform SDK February 2003\Open Build Environment Window\Windows 2000 Build Environment	All Users:Microsoft Platform SDK February 2003\Open Build Environment Window\Windows 2000 Build Environment	All Users	
Microsoft Platform SDK February 2003\Open Build Environment Window\Windows Server 2003 32-bit Build Environment	All Users:Microsoft Platform SDK February 2003\Open Build Environment Window\Windows Server 2003 32-bit Build Environment	All Users	
Microsoft Platform SDK February 2003\Open Build Environment Window\Windows Server 2003 64-bit Build Environment	All Users:Microsoft Platform SDK February 2003\Open Build Environment Window\Windows Server 2003 64-bit Build Environment	All Users	
Microsoft Platform SDK February 2003\Open Build Environment Window\Windows XP 32-bit Build Environment	All Users:Microsoft Platform SDK February 2003\Open Build Environment Window\Windows XP 32-bit Build Environment	All Users	
Microsoft Platform SDK February 2003\Open Build Environment Window\Windows XP 64-bit Build Environment	All Users:Microsoft Platform SDK February 2003\Open Build Environment Window\Windows XP 64-bit Build Environment	All Users	
Microsoft Platform SDK February 2003\Tools	All Users:Microsoft Platform SDK February 2003\Tools	All Users	

Microsoft Platform SDK February 2003\Visual Studio Registration	All Users:Microsoft Platform SDK February 2003\Visual Studio Registration	All Users
Startup	All Users:Startup	All Users
Accessories	NT AUTHORITY\SYSTEM:Accessories	NT AUTHORITY\SYSTEM

Accessories\Accessibility	NT AUTHORITY\SYSTEM:Accessories\Accessibility	NT AUTHORITY\SYSTEM
Accessories\Entertainment	NT AUTHORITY\SYSTEM:Accessories\Entertainment	NT AUTHORITY\SYSTEM
Startup	NT AUTHORITY\SYSTEM:Startup	NT AUTHORITY\SYSTEM
Accessories	TIGER\nec:Accessories	TIGER\nec
Accessories\Accessibility	TIGER\nec:Accessories\Accessibility	TIGER\nec
Accessories\Entertainment	TIGER\nec:Accessories\Entertainment	TIGER\nec
Startup	TIGER\nec:Startup	TIGER\nec

[Startup Programs]

Program	Command	User	Name	Location	Startup
desktop	desktop.ini	NT AUTHORITY\SYSTEM			Startup
desktop	desktop.ini	TIGER\nec			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
12/22/2004 04:46 PM	Application Error	Faulting application taskmgr.exe, version 5.2.3790.0, faulting module rpcrt4.dll, version 5.2.3790.0, fault address 0x000000000000a070.

12/20/2004 07:22 PM	Application Error	Faulting application taskmgr.exe, version 5.2.3790.0, faulting module rpcrt4.dll, version 5.2.3790.0, fault address 0x000000000000a070.

12/19/2004 06:59 PM	Application Error	Faulting application taskmgr.exe, version 5.2.3790.0, faulting module rpcrt4.dll, version 5.2.3790.0, fault address 0x000000000000a070.

12/19/2004 09:31 AM	Application Error	Faulting application taskmgr.exe, version 5.2.3790.0, faulting module rpcrt4.dll, version 5.2.3790.0, fault address 0x000000000000a070.

12/19/2004 07:24 AM	Application Error	Faulting application mmc.exe, version 5.2.3790.0, faulting module sysmon.ocx, version 5.2.3790.0, fault address 0x0000000000006e320.

[illegible]

[illegible]

[illegible]

TPC Benchmark™ C Full Disclosure Report
Copyright © 2005 NEC Corporation

364

Jan 2005

11/2/2004 10:25 AM Application Hang Hanging application mmc.exe, version 5.2.3790.0, hang module kernel32.dll, version 5.2.3790.0, hang address 0x00000000000990d0.

 10/26/2004 12:14 PM Application Hang Hanging application mmc.exe, version 5.2.3790.0, hang module kernel32.dll, version 5.2.3790.0, hang address 0x00000000000990d0.

[Internet Settings]

[Internet Explorer]

[Following are sub-categories of this main category]
 [Summary]

Item	Value
Version	6.0.3790.0
Build	63790
Application Path	C:\Program Files\Internet Explorer
Language	English (United States)
Active Printer	Not Available

Cipher Strength	128-bit
Content Advisor	Disabled
IEAK Install	No

[File Versions]

File	Version	Size	Date	Path	Company
actxprxy.dll	6.0.3790.0	230 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation
advpack.dll	6.0.3790.0	240 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation
asctrls.ocx	6.0.3790.0	219 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation
browseic.dll	6.0.3790.0	62 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation
browseui.dll	6.0.3790.0	2,478 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation
cdfview.dll	6.0.3790.0	292 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation
comctl32.dll	5.82.3790.0	1,584 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation
dxtrans.dll	6.3.3790.0	562 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation
dxtmsft.dll	6.3.3790.0	918 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation
iecont.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
iecontlc.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
iedkcs32.dll	16.0.3790.0	676 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation
iepeers.dll	6.0.3790.0	652 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation
iesetup.dll	6.0.3790.0	89 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation

ieuinit.inf	Not Available	20 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Not Available
ieexplore.exe	6.0.3790.0	102 KB	3/25/2003 09:00:00 PM	C:\Program Files\Internet Explorer	Microsoft Corporation
imgutil.dll	5.2.3790.0	101 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation
inetctl.cpl	6.0.3790.0	589 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation
inetctlc.dll	6.0.3790.0	108 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation
inseng.dll	6.0.3790.0	213 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation
mlang.dll	6.0.3790.0	799 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation
msencode.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
mshta.exe	6.0.3790.0	59 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation
mshtml.dll	6.0.3790.0	8,016 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation
mshtml.tlb	6.0.3790.0	1,319 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation
mshtml.dll	6.0.3790.0	1,376 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation
mshtml.dll	6.0.3790.0	56 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation
msident.dll	6.0.3790.0	128 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation
msident.dll	6.0.3790.0	14 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation
msieftp.dll	6.0.3790.0	536 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation
msrating.dll	6.0.3790.0	379 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation
mstime.dll	6.0.3790.0	1,621 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation
occache.dll	6.0.3790.0	201 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation
proctexe.ocx	<File Missing>	Not Available	Not Available	Not Available	Not Available
sendmail.dll	6.0.3790.0	97 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation
shdoclc.dll	6.0.3790.0	588 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation
shdocvw.dll	6.0.3790.0	3,281 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation
shfolder.dll	6.0.3790.0	37 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation
shlwapi.dll	6.0.3790.0	722 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation
tdc.ocx	1.3.0.3130	177 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation
url.dll	6.0.3790.0	45 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation
urlmon.dll	6.0.3790.0	1,242 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation
webcheck.dll	6.0.3790.0	665 KB	3/25/2003 09:00:00 PM	C:\WINDOWS\system32	Microsoft Corporation

wininet.dll 6.0.3790.0 1,466 KB 3/25/2003 09:00:00 PM
C:\WINDOWS\system32 Microsoft Corporation

[Connectivity]

Item	Value
Connection Preference	Never dial

LAN Settings

AutoConfigProxy	Not Available
AutoProxyDetectMode	Disabled
AutoConfigURL	
Proxy	Disabled
ProxyServer	
ProxyOverride	

[Cache]

[Following are sub-categories of this main category]
[Summary]

Item	Value
Page Refresh Type	Automatic
Temporary Internet Files Folder	C:\Documents and Settings\NetworkService\Local Settings\Temporary Internet Files
Total Disk Space	Not Available
Available Disk Space	Not Available
Maximum Cache Size	Not Available
Available Cache Size	Not Available

[List of Objects]

Program File	Status	CodeBase
SDKInstall Class	Installed	file:///D:\controls\sdkinst.cab

[Content]

[Following are sub-categories of this main category]
[Summary]

Item	Value
Content Advisor	Disabled

[Personal Certificates]

Issued To	Issued By	Validity	Signature Algorithm
No personal certificate information available			

[Other People Certificates]

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

[Publishers]

Name
No publisher information available

[Security]

Zone	Security Level
My Computer	Custom
Local intranet	Medium-low
Trusted sites	Low
Internet	Medium
Restricted sites	High

Client Configuration

COM+ Settings (properties of component TPCC.ALLTxns)

Transactions: not supported

Enable object pooling

- Minimum pool size: 15
- Maximum pool size: 15
- Creation timeout (ms): 1200000

Enable object construction

- Constructor string: "dummy string (do not remove)"

Enable just in time activation

Component supports events and statistics

Concurrency: required

TPCC applicaiton registry settings

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\TPCC]
"Path"="c:\inetpub\wwwroot\"
"NumberOfDeliveryThreads"=dword:00000006
"MaxConnections"=dword:00004e20
"MaxPendingDeliveries"=dword:00002328
"DB_Protocol"="DBLIB"
"TxnMonitor"="COM"
"DbServer"="tiger"
"DbName"="tpcc"
"DbUser"="nec"
"DbPassword"="nec"
"COM_SinglePool"="YES"
```

Microsoft internet information server registry settings

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo]

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo\Parameters]
"ListenBackLog"=dword:000000fa
"DispatchEntries"=hex(7):4c,00,44,00,41,00,50,00,53,00,56,00,43,00,00,00,00,00
"PoolThreadLimit"=dword:00001000
```

```
"ThreadTimeout"=dword:00015180
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo\Performance]
"Library"="infoctrs.dll"
"Open"="OpenINFOPerformanceData"
"Close"="CloseINFOPerformanceData"
"Collect"="CollectINFOPerformanceData"
"Last Counter"=dword:00000b4a
"Last Help"=dword:00000b4b
"First Counter"=dword:00000b0a
"First Help"=dword:00000b0b
"Library Validation
Code"=hex:a2,79,55,c1,2a,b6,c2,01,10,25,00,00,00,00,00,00
"WbemAdapFileTime"=hex:00,c3,bb,02,47,d4,c0,01
"WbemAdapFileSize"=dword:00002510
"WbemAdapStatus"=dword:00000000
```

World Wide Web service registry settings

```
[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\ScriptMap]
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Parameters\VirtualRoots]
"/"="c:\\inetpub\\wwwroot,,205"
"/Scripts"="c:\\inetpub\\scripts,,204"
"/IISHelp"="c:\\winnt\\help\\iishelp,,201"
"/IISAdmin"="C:\\WINNT\\System32\\inetsrv\\iisadmin,,201"
"/IISamples"="c:\\inetpub\\iissamples,,201"
"/MSADC"="c:\\program files\\common files\\system\\msadc,,205"
"/Printers"="C:\\WINNT\\web\\printers,,201"
"/_vti_bin"="C:\\Program Files\\Common Files\\Microsoft Shared\\Web Server Extensions\\40\\isapi,,205"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Performance]
"Library"="w3ctrts.dll"
"Open"="OpenW3PerformanceData"
"Close"="CloseW3PerformanceData"
"Collect"="CollectW3PerformanceData"
"Last Counter"=dword:00000bee
"Last Help"=dword:00000bef
"First Counter"=dword:00000b4c
"First Help"=dword:00000b4d
"Library Validation
Code"=hex:c4,e8,94,aa,2a,b6,c2,01,10,1d,00,00,00,00,00,00
"WbemAdapFileTime"=hex:00,c3,bb,02,47,d4,c0,01
"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,\
05,12,00,00,00,74,00,6f,00,00,00,00,1c,00,ff,01,0f,00,01,02,00,00,00,00,00,\
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,\
00,05,20,00,00,00,23,02,00,00,72,00,73,00,01,01,00,00,00,00,00,05,12,00,\
00,01,01,00,00,00,00,00,05,12,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Enum]
"0"="Root\LEGACY_W3SVC\0000"
"Count"=dword:00000001
"NextInstance"=dword:00000001
```

System information

System Information report written at: 12/24/2004 05:26:35 PM
[System Information]

[Following are sub-categories of this main category]

[System Summary]

Item	Value
OS Name	Microsoft Windows 2000 Server
Version	5.0.2195 Service Pack 2 Build 2195
OS Manufacturer	Microsoft Corporation
System Name	ACL37
System Manufacturer	NEC
System Model	Express5800/120Rf-2 [N8100-878]
System Type	X86-based PC
Processor	x86 Family 15 Model 2 Stepping 7 GenuineIntel ~1994 Mhz
Processor	x86 Family 15 Model 2 Stepping 7 GenuineIntel ~1994 Mhz
Processor	x86 Family 15 Model 2 Stepping 7 GenuineIntel ~1994 Mhz
Processor	x86 Family 15 Model 2 Stepping 7 GenuineIntel ~1994 Mhz
BIOS Version	SWV25 v3.00
Windows Directory	C:\WINNT
System Directory	C:\WINNT\System32
Boot Device	\Device\Harddisk0\Partition1
Locale	United States
User Name	ACL37\nec
Total Physical Memory	2,096,604 KB
Available Physical Memory	1,859,532 KB
Total Virtual Memory	4,429,624 KB
Available Virtual Memory	4,097,224 KB
Page File Space	2,333,020 KB
Page File	C:\pagefile.sys

[Hardware Resources]

[Following are sub-categories of this main category]

[Conflicts/Sharing]

Resource	Device
No conflicted/shared resources	

[DMA]

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

[Forced Hardware]

Device	PNP Device ID
No Forced Hardware	

[I/O]

Address Range	Device	Status
0x0000-0x0CF7	PCI bus	OK
0x0000-0x0CF7	Direct memory access controller	OK
0x0D00-0xFFFF	PCI bus	OK
0x2000-0x4FFF	PCI standard PCI-to-PCI bridge	OK
0x2000-0x4FFF	PCI standard PCI-to-PCI bridge	OK
0x2000-0x4FFF	Intel(R) PRO/100+ PCI Adapter #2	OK
0x3000-0x4FFF	PCI standard PCI-to-PCI bridge	OK
0x3000-0x4FFF	Adaptec AIC-7902 - Ultra320 SCSI	OK
0x3800-0x38FF	Adaptec AIC-7902 - Ultra320 SCSI	OK
0x4000-0x40FF	Adaptec AIC-7902 - Ultra320 SCSI	OK
0x3400-0x34FF	Adaptec AIC-7902 - Ultra320 SCSI	OK
0x20C0-0x20FF	Intel(R) 82546EB Based Dual Port Network Connection #3	OK
0x2080-0x20BF	Intel(R) 82546EB Based Dual Port Network Connection #4	OK
0x2040-0x207F	Intel(R) PRO/100+ PCI Adapter	OK
0x5020-0x503F	Standard Universal PCI to USB Host Controller	OK
0x5000-0x501F	Standard Universal PCI to USB Host Controller	OK
0x1000-0x10FF	ATI Technologies Inc. RAGE XL PCI	OK
0x03B0-0x03BB	ATI Technologies Inc. RAGE XL PCI	OK
0x03C0-0x03DF	ATI Technologies Inc. RAGE XL PCI	OK
0x0A79-0x0A79	ISAPNP Read Data Port	OK
0x0279-0x0279	ISAPNP Read Data Port	OK
0x0274-0x0277	ISAPNP Read Data Port	OK
0x0092-0x0092	Motherboard resources	OK
0x00B2-0x00B3	Motherboard resources	OK
0x03F0-0x03F1	Motherboard resources	OK
0x0400-0x04BF	Motherboard resources	OK
0x04D0-0x04D1	Motherboard resources	OK
0x0010-0x001F	Motherboard resources	OK
0x040B-0x040B	Motherboard resources	OK
0x04D6-0x04D6	Motherboard resources	OK
0x0C14-0x0C14	Motherboard resources	OK
0x0C49-0x0C49	Motherboard resources	OK
0x0C52-0x0C52	Motherboard resources	OK
0x0C6C-0x0C6C	Motherboard resources	OK
0x0C6F-0x0C6F	Motherboard resources	OK
0x0F50-0x0F57	Motherboard resources	OK
0x0C00-0x0C01	Motherboard resources	OK
0x0C98-0x0C98	Motherboard resources	OK
0x0CD6-0x0CD7	Motherboard resources	OK
0x002E-0x002F	Motherboard resources	OK
0x0530-0x0531	Motherboard resources	OK
0x0500-0x052F	Motherboard resources	OK
0x0532-0x053F	Motherboard resources	OK
0x0540-0x055F	Motherboard resources	OK
0x0560-0x057F	Motherboard resources	OK
0x05A0-0x05BF	Motherboard resources	OK
0x0020-0x0021	Programmable interrupt controller	OK
0x00A0-0x00A1	Programmable interrupt controller	OK
0x0024-0x0025	Programmable interrupt controller	OK
0x0028-0x0029	Programmable interrupt controller	OK
0x002C-0x002D	Programmable interrupt controller	OK
0x0030-0x0031	Programmable interrupt controller	OK
0x0034-0x0035	Programmable interrupt controller	OK

0x0038-0x0039	Programmable interrupt controller	OK
0x003C-0x003D	Programmable interrupt controller	OK
0x0050-0x0052	Programmable interrupt controller	OK
0x00A4-0x00A5	Programmable interrupt controller	OK
0x00A8-0x00A9	Programmable interrupt controller	OK
0x00AC-0x00AD	Programmable interrupt controller	OK
0x00B0-0x00B1	Programmable interrupt controller	OK
0x00B4-0x00B5	Programmable interrupt controller	OK
0x00B8-0x00B9	Programmable interrupt controller	OK
0x00BC-0x00BD	Programmable interrupt controller	OK
0x0080-0x0080	Direct memory access controller	OK
0x0081-0x0083	Direct memory access controller	OK
0x0084-0x0086	Direct memory access controller	OK
0x0087-0x0087	Direct memory access controller	OK
0x0088-0x0088	Direct memory access controller	OK
0x0089-0x008B	Direct memory access controller	OK
0x008C-0x008E	Direct memory access controller	OK
0x008F-0x008F	Direct memory access controller	OK
0x0090-0x0091	Direct memory access controller	OK
0x0093-0x009F	Direct memory access controller	OK
0x00C0-0x00DF	Direct memory access controller	OK
0x0040-0x0043	System timer	OK
0x0070-0x0071	System CMOS/real time clock	OK
0x0072-0x0073	System CMOS/real time clock	OK
0x0074-0x0075	System CMOS/real time clock	OK
0x0076-0x0077	System CMOS/real time clock	OK
0x0061-0x0061	System speaker	OK
0x00F0-0x00FF	Numeric data processor	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
0x0CA6-0x0CA6	Microsoft ACPI-Compliant Embedded Controller	OK
0x0CA7-0x0CA7	Microsoft ACPI-Compliant Embedded Controller	OK
0x03F2-0x03F3	Standard floppy disk controller	OK
0x03F4-0x03F5	Standard floppy disk controller	OK
0x03F7-0x03F7	Standard floppy disk controller	OK
0x03F8-0x03FF	Communications Port (COM1)	OK
0x02F8-0x02FF	Communications Port (COM2)	OK
0x03A0-0x03AF	Standard Dual Channel PCI IDE Controller	OK
0x01F0-0x01F7	Primary IDE Channel	OK
0x03F6-0x03F6	Primary IDE Channel	OK
0x0170-0x0177	Secondary IDE Channel	OK
0x0376-0x0376	Secondary IDE Channel	OK
0x0580-0x059F	PCI Device	OK

[IRQs]

IRQ Number	Device
9	Microsoft ACPI-Compliant System
50	Adaptec AIC-7902 - Ultra320 SCSI
49	Adaptec AIC-7902 - Ultra320 SCSI
30	Intel(R) 82546EB Based Dual Port Network Connection #3
31	Intel(R) 82546EB Based Dual Port Network Connection #4
24	Intel(R) PRO/100+ PCI Adapter
27	Intel(R) PRO/100+ PCI Adapter #2
16	Standard Universal PCI to USB Host Controller
19	Standard Universal PCI to USB Host Controller
17	ATI Technologies Inc. RAGE XL PCI

8	System CMOS/real time clock
13	Numeric data processor
12	Logitech PS/2 Port Mouse
1	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard
6	Standard floppy disk controller
4	Communications Port (COM1)
3	Communications Port (COM2)
15	Secondary IDE Channel

[Memory]

Range	Device	Status
0xA0000-0xBFFFF	PCI bus	OK
0xA0000-0xBFFFF	ATI Technologies Inc. RAGE XL PCI	OK
0x80000000-0xFFFFFFFF	PCI bus	OK
0xFDE00000-0xFEFFFFFF	PCI standard PCI-to-PCI bridge	OK
0xFDE00000-0xFEFFFFFF	PCI standard PCI-to-PCI bridge	OK
0xFB900000-0xFBBFFFFFF	PCI standard PCI-to-PCI bridge	OK
0xFB900000-0xFBBFFFFFF	PCI standard PCI-to-PCI bridge	OK
0xFEAE0000-0xFEAE0FFF	System Interrupt Controller	OK
0xFE700000-0xFE9FFFFF	PCI standard PCI-to-PCI bridge	OK
0xFBA00000-0xFBAFFFFFF	PCI standard PCI-to-PCI bridge	OK
0xFE9E0000-0xFE9E1FFF	Adaptec AIC-7902 - Ultra320 SCSI	OK
0xFE9F0000-0xFE9F1FFF	Adaptec AIC-7902 - Ultra320 SCSI	OK
0xFEAF0000-0xFEAF0FFF	System Interrupt Controller	OK
0xFE600000-0xFE67FFFF	Intel(R) 82546EB Based Dual Port Network Connection #3	OK
0xFE680000-0xFE69FFFF	Intel(R) 82546EB Based Dual Port Network Connection #4	OK
0xFE6B0000-0xFE6B0FFF	Intel(R) PRO/100+ PCI Adapter	OK
0xFE300000-0xFE3FFFFF	Intel(R) PRO/100+ PCI Adapter	OK
0xFE6E0000-0xFE6E0FFF	Intel(R) PRO/100+ PCI Adapter #2	OK
0xFE500000-0xFE5FFFFF	Intel(R) PRO/100+ PCI Adapter #2	OK
0xFC000000-0xFCFFFFFF	ATI Technologies Inc. RAGE XL PCI	OK
0xFDCF0000-0xFDCF0FFF	ATI Technologies Inc. RAGE XL PCI	OK
0xFFFF0000-0xFFFFFFFF	Intel(r) 82802 Firmware Hub Device	OK
0xFFE00000-0xFFEFFFFFF	Intel(r) 82802 Firmware Hub Device	OK
0xFFD00000-0xFFDFFFFFF	Intel(r) 82802 Firmware Hub Device	OK
0xFFC00000-0xFFCFFFFFF	Intel(r) 82802 Firmware Hub Device	OK
0xFFBFFC00-0xFFBFFFFFF	Standard Dual Channel PCI IDE Controller	OK

[Components]

[Following are sub-categories of this main category]

[Multimedia]

[Following are sub-categories of this main category]

[Audio Codecs]

Codec	Manufacturer	Description	Status	File	Version	Size
		Creation Date				
c:\winnt\system32\iac25_32.ax	Intel Corporation	Indeo® audio software	OK	C:\WINNT\System32\IAC25_32.AX	2.05.53	195.00 KB
		(199,680 bytes)		12/8/1999 5:00:00 AM		
c:\winnt\system32\msg723.acm	Microsoft Corporation				OK	
		C:\WINNT\System32\MSG723.ACM	4.4.3385		106.77 KB	(109,328 bytes)
		1/7/2003 5:40:48 PM				

```

c:\winnt\system32\lhacm.acm Microsoft Corporation OK
C:\WINNT\System32\LHACM.ACM 4.4.3385 33.27 KB (34,064
bytes) 1/7/2003 5:40:49 PM
c:\winnt\system32\tsssoft32.acm DSP GROUP, INC. OK
C:\WINNT\System32\TSSOFT32.ACM 1.01 9.27 KB (9,488
bytes) 12/8/1999 5:00:00 AM
c:\winnt\system32\msg711.acm Microsoft Corporation OK
C:\WINNT\System32\MSG711.ACM 5.00.2134.1 10.27 KB (10,512
bytes) 12/8/1999 5:00:00 AM
c:\winnt\system32\msgsm32.acm Microsoft Corporation OK
C:\WINNT\System32\MSGSM32.ACM 5.00.2134.1 22.27 KB (22,800
bytes) 12/8/1999 5:00:00 AM
c:\winnt\system32\msadp32.acm Microsoft Corporation OK
C:\WINNT\System32\MSADP32.ACM 5.00.2134.1 14.77 KB (15,120
bytes) 12/8/1999 5:00:00 AM
c:\winnt\system32\imaadp32.acm Microsoft Corporation OK
C:\WINNT\System32\IMAADP32.ACM 5.00.2134.1 16.27 KB
(16,656 bytes) 12/8/1999 5:00:00 AM

```

[Video Codecs]

Codec	Manufacturer	Description	Status	File	Version	Size
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)
c:\winnt\system32\msh261.drv	Microsoft Corporation		OK	C:\WINNT\System32\MSH261.DRV	4.4.3385	163.77 KB (167,696 bytes)
c:\winnt\system32\msh263.drv	Microsoft Corporation		OK	C:\WINNT\System32\MSH263.DRV	4.4.3385	252.27 KB (258,320 bytes)
c:\winnt\system32\ir32_32.dll	Intel(R) Corporation		OK	C:\WINNT\System32\IR32_32.DLL	Not Available	194.50 KB (199,168 bytes)
c:\winnt\system32\msvidc32.dll	Microsoft Corporation		OK	C:\WINNT\System32\MSVIDC32.DLL	5.00.2134.1	27.27 KB (27,920 bytes)
c:\winnt\system32\msrle32.dll	Microsoft Corporation		OK	C:\WINNT\System32\MSRLE32.DLL	5.00.2134.1	10.77 KB (11,024 bytes)
c:\winnt\system32\iccvid.dll	Radius Inc.		OK	C:\WINNT\System32\ICCVID.DLL	1.10.0.6	108.00 KB (110,592 bytes)

[CD-ROM]

```

Item Value
Drive D:
Description CD-ROM Drive
Media Loaded False
Media Type CD-ROM
Name MITSUMI CD-ROM SR244W
Manufacturer (Standard CD-ROM drives)
Status OK
Transfer Rate Not Available
SCSI Target ID 0
PNP Device ID IDE\CDROMMITSUMI_CD-
ROM_SR244W\T01A\5&2C4DDEAA&0&0.0.0

```

[Sound Device]

```

Item Value
No sound devices

```

[Display]

```

Item Value
Name ATI Technologies Inc. RAGE XL PCI
PNP Device ID PCI\VEN_1002&DEV_4752&SUBSYS_81B41033&REV_27\4&27A7C225&0&60F0
Adapter Type ATI RAGE XL PCI, ATI Technologies Inc. compatible
Adapter Description ATI Technologies Inc. RAGE XL PCI
Adapter RAM 8.00 MB (8,388,608 bytes)
Installed Drivers atidrab.dll
Driver Version 5.00.2179.1
INF File display.inf (atirage3 section)
Color Planes 1
Color Table Entries 65536
Resolution 1024 x 768 x 60 hertz
Bits/Pixel 16

```

[Infrared]

```

Item Value
No infrared devices

```

[Input]

[Following are sub-categories of this main category]

[Keyboard]

```

Item Value
Description Standard 101/102-Key or Microsoft Natural PS/2 Keyboard
Name Enhanced (101- or 102-key)
Layout 00000409
PNP Device ID ACPI\PNP0303\4&1E30281&0
NumberOfFunctionKeys 12

```

[Pointing Device]

```

Item Value
Hardware Type Logitech PS/2 Port Mouse
Number of Buttons 5
Status OK
PNP Device ID ACPI\PNP0F12\4&1E30281&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 Not Available

Index 0

Service Name AsyncMac

IP Address Not Available

IP Subnet Not Available

Default IP Gateway Not Available

DHCP Enabled False

DHCP Server Not Available

DHCP Lease Expires Not Available

DHCP Lease Obtained Not Available

MAC Address Not Available

Service Name Not Available

Name [00000001] WAN Miniport (L2TP)

Adapter Type Not Available

Product Name WAN Miniport (L2TP)

Installed True

PNP Device ID ROOT\MS_L2TPMINIPORT\0000

Last Reset Not Available

Index 1

Service Name Rasl2tp

IP Address Not Available

IP Subnet Not Available

Default IP Gateway Not Available

DHCP Enabled False

DHCP Server Not Available

DHCP Lease Expires Not Available

DHCP Lease Obtained Not Available

MAC Address Not Available

Service Name Rasl2tp

Driver c:\winnt\system32\drivers\rasl2tp.sys (50800, 5.00.2179.1)

Name [00000002] WAN Miniport (PPTP)

Adapter Type Wide Area Network (WAN)

Product Name WAN Miniport (PPTP)

Installed True

PNP Device ID ROOT\MS_PPTPMINIPORT\0000

Last Reset Not Available

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\raspttp.sys (47856, 5.00.2160.1)

Name [00000003] Direct Parallel

Adapter Type Not Available

Product Name Direct Parallel

Installed True

PNP Device ID ROOT\MS_PTMINIPORT\0000

Last Reset Not Available

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 Not Available

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 (90096, 5.00.2195.2779)

Name [00000005] Intel(R) 82546EB Based Dual Port Network Connection

Adapter Type Not Available

Product Name Intel(R) 82546EB Based Dual Port Network Connection

Installed True

PNP Device ID Not Available

Last Reset Not Available

Index 5

Service Name E1000

IP Address 10.10.37.250

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 00:07:E9:06:48:1B
Service Name Not Available

Name [00000006] Intel(R) 82546EB Based Dual Port Network Connection
Adapter Type Not Available
Product Name Intel(R) 82546EB Based Dual Port Network Connection
Installed True
PNP Device ID Not Available
Last Reset Not Available
Index 6
Service Name E1000
IP Address 10.30.37.250
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 00:07:E9:0F:CF:E8
Service Name Not Available

Name [00000007] Intel(R) PRO/100 S Server Adapter
Adapter Type Not Available
Product Name Intel(R) PRO/100 S Server Adapter
Installed True
PNP Device ID Not Available
Last Reset Not Available
Index 7
Service Name E100B
IP Address 10.20.37.250
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 00:07:E9:0F:C9:C3
Service Name Not Available

Name [00000008] Intel(R) PRO/100+ PCI Adapter
Adapter Type Ethernet 802.3
Product Name Intel(R) PRO/100+ PCI Adapter
Installed True
PNP Device ID PCI\VEN_8086&DEV_1229&SUBSYS_000B8086&REV_08\5&279870C6&0&40F818
Last Reset Not Available
Index 8
Service Name E100B
IP Address 10.30.37.250
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 00:07:E9:0F:CF:E8
Service Name E100B
IRQ Number 24
I/O Port 0x2040-0x207F

Driver c:\winnt\system32\drivers\ei100bnt5.sys (85776, 4.02.38.0000)

Name [00000009] Intel(R) PRO/100+ PCI Adapter
Adapter Type Ethernet 802.3
Product Name Intel(R) PRO/100+ PCI Adapter
Installed True
PNP Device ID PCI\VEN_8086&DEV_1229&SUBSYS_000B8086&REV_08\5&279870C6&0&48F818
Last Reset Not Available
Index 9
Service Name E100B
IP Address 10.20.37.250
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 00:07:E9:0F:C9:C3
Service Name E100B
IRQ Number 27
I/O Port 0x2000-0x4FFF
Driver c:\winnt\system32\drivers\ei100bnt5.sys (85776, 4.02.38.0000)

Name [00000010] Intel(R) 82546EB Based Dual Port Network Connection
Adapter Type Ethernet 802.3
Product Name Intel(R) 82546EB Based Dual Port Network Connection
Installed True
PNP Device ID PCI\VEN_8086&DEV_1010&SUBSYS_81B41033&REV_01\5&279870C6&0&38F818
Last Reset Not Available
Index 10
Service Name E1000
IP Address 10.1.1.37
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 00:07:E9:06:48:1A
Service Name E1000
IRQ Number 30
I/O Port 0x20C0-0x20FF
Driver c:\winnt\system32\drivers\ei1000nt5.sys (100544, 6.2.11.0)

Name [00000011] Intel(R) 82546EB Based Dual Port Network Connection
Adapter Type Ethernet 802.3
Product Name Intel(R) 82546EB Based Dual Port Network Connection
Installed True
PNP Device ID PCI\VEN_8086&DEV_1010&SUBSYS_81B41033&REV_01\5&279870C6&0&39F818
Last Reset Not Available
Index 11
Service Name E1000
IP Address 10.10.37.250
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 00:07:E9:06:48:1B
 Service Name E1000
 IRQ Number 31
 I/O Port 0x2080-0x20BF
 Driver c:\winnt\system32\drivers\e1000nt5.sys (100544, 6.2.11.0)

[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
 SupportsGracefulClosing True
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD Tcpip [UDP/IP]
 ConnectionlessService True
 GuaranteesDelivery False
 GuaranteesSequencing False
 MaximumAddressSize 16 bytes
 MaximumMessageSize 65467 bytes
 MessageOriented True
 MinimumAddressSize 16 bytes
 PseudoStreamOriented False
 SupportsBroadcasting True
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting True

Name RSVP UDP Service Provider
 ConnectionlessService True
 GuaranteesDelivery False
 GuaranteesSequencing False
 MaximumAddressSize 16 bytes
 MaximumMessageSize 65467 bytes
 MessageOriented True
 MinimumAddressSize 16 bytes
 PseudoStreamOriented False
 SupportsBroadcasting True

SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption True
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting True

Name RSVP TCP Service Provider
 ConnectionlessService False
 GuaranteesDelivery True
 GuaranteesSequencing True
 MaximumAddressSize 16 bytes
 MaximumMessageSize 0 bytes
 MessageOriented False
 MinimumAddressSize 16 bytes
 PseudoStreamOriented False
 SupportsBroadcasting False
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption True
 SupportsExpeditedData True
 SupportsGracefulClosing True
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{7B5E68E7-FFB9-4171-9CE1-70DF24F4E518}] SEQUENCE 8
 ConnectionlessService False
 GuaranteesDelivery True
 GuaranteesSequencing True
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting False
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{7B5E68E7-FFB9-4171-9CE1-70DF24F4E518}] 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_{46926B5F-C35E-41FF-8A02-F91EFC3CFCE4}] SEQUENCE 7
 ConnectionlessService False
 GuaranteesDelivery True
 GuaranteesSequencing True
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting False
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{46926B5F-C35E-41FF-8A02-F91EFC3CFCE4}] 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_{8894E682-5FBF-4DE6-8E40-90D31E4FFC10}] SEQUENCE 6
 ConnectionlessService False
 GuaranteesDelivery True
 GuaranteesSequencing True
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting False
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False

SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{8894E682-5FBF-4DE6-8E40-90D31E4FFC10}] DATAGRAM 6
 ConnectionlessService True
 GuaranteesDelivery False
 GuaranteesSequencing False
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting True
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{2D7243BB-3372-45BC-A771-7CA29A08AD72}] SEQUENCE 5
 ConnectionlessService False
 GuaranteesDelivery True
 GuaranteesSequencing True
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting False
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{2D7243BB-3372-45BC-A771-7CA29A08AD72}] DATAGRAM 5
 ConnectionlessService True
 GuaranteesDelivery False
 GuaranteesSequencing False
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting True
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{270A0FA5-BFD6-4A84-98F7-E8F3BD1B81F3}] SEQUENCE 4
 ConnectionlessService False
 GuaranteesDelivery True
 GuaranteesSequencing True
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting False
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{270A0FA5-BFD6-4A84-98F7-E8F3BD1B81F3}] DATAGRAM 4
 ConnectionlessService True
 GuaranteesDelivery False
 GuaranteesSequencing False
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting True
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{0E28669D-E854-4A06-816C-3A2B19320786}] SEQUENCE 3
 ConnectionlessService False
 GuaranteesDelivery True
 GuaranteesSequencing True
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting False
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{0E28669D-E854-4A06-816C-

3A2B19320786}] DATAGRAM 3
 ConnectionlessService True
 GuaranteesDelivery False
 GuaranteesSequencing False
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting True
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{C0B1C147-666A-427C-86E3-600E14EE91C3}] SEQUENCE 0
 ConnectionlessService False
 GuaranteesDelivery True
 GuaranteesSequencing True
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting False
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{C0B1C147-666A-427C-86E3-600E14EE91C3}] DATAGRAM 0
 ConnectionlessService True
 GuaranteesDelivery False
 GuaranteesSequencing False
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting True
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{0D298F46-E414-4F80-97D6-DF61912CAAB5}] SEQUENCE 1
 ConnectionlessService False

GuaranteesDelivery True
 GuaranteesSequencing True
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting False
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{0D298F46-E414-4F80-97D6-DF61912CAAB5}] DATAGRAM 1
 ConnectionlessService True
 GuaranteesDelivery False
 GuaranteesSequencing False
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting True
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{1118D47B-7ADB-4639-9F2F-2479CC247350}] SEQPACKET 2
 ConnectionlessService False
 GuaranteesDelivery True
 GuaranteesSequencing True
 MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting False
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{1118D47B-7ADB-4639-9F2F-2479CC247350}] DATAGRAM 2
 ConnectionlessService True
 GuaranteesDelivery False
 GuaranteesSequencing False

MaximumAddressSize 20 bytes
 MaximumMessageSize 64000 bytes
 MessageOriented True
 MinimumAddressSize 20 bytes
 PseudoStreamOriented False
 SupportsBroadcasting True
 SupportsConnectData False
 SupportsDisconnectData False
 SupportsEncryption False
 SupportsExpeditedData False
 SupportsGracefulClosing False
 SupportsGuaranteedBandwidth False
 SupportsMulticasting False

[WinSock]

Item Value
 File c:\winnt\system32\winsock.dll
 Version 3.10
 Size 2.80 KB (2,864 bytes)

 File c:\winnt\system32\wsock32.dll
 Version 5.00.2195.2871
 Size 21.27 KB (21,776 bytes)

[Ports]

[Following are sub-categories of this main category]

[Serial]

Item Value
 Name COM1
 Status OK
 PNP Device ID ACPI\PNP0501\1
 Maximum Input Buffer Size 0
 Maximum Output Buffer Size False
 Settable Baud Rate True
 Settable Data Bits True
 Settable Flow Control True
 Settable Parity True
 Settable Parity Check True
 Settable Stop Bits True
 Settable RLSD True
 Supports RLSD True
 Supports 16 Bit Mode False
 Supports Special Characters False
 Baud Rate 9600
 Bits/Byte 8
 Stop Bits 1
 Parity None
 Busy 0
 Abort Read/Write on Error 0
 Binary Mode Enabled -1
 Continue XMit on XOff 0
 CTS Outflow Control 0
 Discard NULL Bytes 0

DSR Outflow Control 0
 DSR Sensitivity 0
 DTR Flow Control Type Enable
 EOF Character 0
 Error Replace Character 0
 Error Replacement Enabled 0
 Event Character 0
 Parity Check Enabled 0
 RTS Flow Control Type Enable
 XOff Character 19
 XOffXMit Threshold 512
 XOn Character 17
 XOnXMit Threshold 2048
 XOnXOff InFlow Control 0
 XOnXOff OutFlow Control 0
 IRQ Number 4
 I/O Port 0x03F8-0x03FF
 Driver c:\winnt\system32\drivers\serial.sys (62416, 5.00.2195.2780)

Name COM2
 Status OK
 PNP Device ID ACPI\PNP0501\2
 Maximum Input Buffer Size 0
 Maximum Output Buffer Size False
 Settable Baud Rate True
 Settable Data Bits True
 Settable Flow Control True
 Settable Parity True
 Settable Parity Check True
 Settable Stop Bits True
 Settable RLSO True
 Supports RLSO True
 Supports 16 Bit Mode False
 Supports Special Characters False
 Baud Rate 9600
 Bits/Byte 8
 Stop Bits 1
 Parity None
 Busy 0
 Abort Read/Write on Error 0
 Binary Mode Enabled -1
 Continue XMit on XOff 0
 CTS Outflow Control 0
 Discard NULL Bytes 0
 DSR Outflow Control 0
 DSR Sensitivity 0
 DTR Flow Control Type Enable
 EOF Character 0
 Error Replace Character 0
 Error Replacement Enabled 0
 Event Character 0
 Parity Check Enabled 0
 RTS Flow Control Type Enable
 XOff Character 19
 XOffXMit Threshold 512
 XOn Character 17
 XOnXMit Threshold 2048
 XOnXOff InFlow Control 0
 XOnXOff OutFlow Control 0

IRQ Number 3
 I/O Port 0x02F8-0x02FF
 Driver c:\winnt\system32\drivers\serial.sys (62416, 5.00.2195.2780)

[Parallel]

Item Value
 No parallel port information

[Storage]

[Following are sub-categories of this main category]

[Drives]

Item Value
 Drive A:
 Description 3 1/2 Inch Floppy Drive

Drive C:
 Description Local Fixed Disk
 Compressed False
 File System NTFS
 Size 16.93 GB (18,177,835,008 bytes)
 Free Space 10.11 GB (10,853,646,336 bytes)
 Volume Name
 Volume Serial Number 80FDAE3C
 Partition Disk #0, Partition #0
 Partition Size 16.93 GB (18,177,836,544 bytes)
 Starting Offset 32256 bytes
 Drive Description Disk drive
 Drive Manufacturer (Standard disk drives)
 Drive Model MAXTOR ATLASU320_36_SCA SCSI Disk Device
 Drive BytesPerSector 512
 Drive MediaLoaded True
 Drive MediaType Fixed hard disk media
 Drive Partitions 1
 Drive SCSI Bus 0
 Drive SCSI Logical Unit 0
 Drive SCSI Port 3
 Drive SCSI Target Id 0
 Drive SectorsPerTrack 63
 Drive Size 36372188160 bytes
 Drive TotalCylinders 4422
 Drive TotalSectors 71039430
 Drive TotalTracks 1127610
 Drive TracksPerCylinder 255

Drive F:
 Description Network Connection
 Provider Name \\10.1.1.48\c\$

[SCSI]

Item Value
 Name Adaptec AIC-7902 - Ultra320 SCSI
 Caption Adaptec AIC-7902 - Ultra320 SCSI

Driver adpu320
 Status OK
 PNP Device ID
 PCI\VEN_9005&DEV_801F&SUBSYS_81B41033&REV_03\5&36D4E837&0&38E818
 Device ID
 PCI\VEN_9005&DEV_801F&SUBSYS_81B41033&REV_03\5&36D4E837&0&38E818
 Device Map Not Available
 Index Not Available
 Max Number Controlled Not Available
 IRQ Number 50
 I/O Port 0x3800-0x38FF
 I/O Port 0x4000-0x40FF
 Driver c:\winnt\system32\drivers\adpu320.sys (120995, 1.1.000.000)

Name Adaptec AIC-7902 - Ultra320 SCSI
 Caption Adaptec AIC-7902 - Ultra320 SCSI
 Driver adpu320
 Status OK
 PNP Device ID
 PCI\VEN_9005&DEV_801F&SUBSYS_81B41033&REV_03\5&36D4E837&0&39E818
 Device ID
 PCI\VEN_9005&DEV_801F&SUBSYS_81B41033&REV_03\5&36D4E837&0&39E818
 Device Map Not Available
 Index Not Available
 Max Number Controlled Not Available
 IRQ Number 49
 I/O Port 0x3000-0x4FFF
 I/O Port 0x3400-0x34FF
 Driver c:\winnt\system32\drivers\adpu320.sys (120995, 1.1.000.000)

[Printing]

Name Port Name Server Name
 No printing information

[Problem Devices]

Device PNP Device ID Error Code
 PCI Device
 PCI\VEN_8086&DEV_2541&SUBSYS_81B41033&REV_01\3&267A616A&0&0128
 System Interrupt Controller
 PCI\VEN_8086&DEV_1461&SUBSYS_81B41033&REV_04\4&38945BE0&0&E01828
 ESG-SHV SCA HSBP M16 SCSI Processor Device SCSI\PROCESSOR&VEN_ESG-SHV&PROD_SCA_HSBP_M16&REV_0.05\6&1B6A8BB1&0&06028
 System Interrupt Controller
 PCI\VEN_8086&DEV_1461&SUBSYS_81B41033&REV_04\4&38945BE0&0&F01828
 PCI Device
 PCI\VEN_8086&DEV_2546&SUBSYS_81B41033&REV_01\3&267A616A&0&01928
 PCI Device
 PCI\VEN_8086&DEV_2483&SUBSYS_81B41033&REV_02\3&267A616A&0&FB28

[USB]

Device PNP Device ID
 Standard Universal PCI to USB Host Controller
 PCI\VEN_8086&DEV_2482&SUBSYS_81B41033&REV_02\3&267A616A&0&E8
 USB Root Hub USB\ROOT_HUB\4&42B666F&0
 Standard Universal PCI to USB Host Controller
 PCI\VEN_8086&DEV_2484&SUBSYS_81B41033&REV_02\3&267A616A&0&E9
 USB Root Hub USB\ROOT_HUB\4&396CDB1C&0

[Software Environment]

[Following are sub-categories of this main category]

[Drivers]

Name	Description	File	Type	Started	Start Mode	State
abiosdsk	Microsoft BIOS Disk Controller Driver	c:\winnt\system32\drivers\abiosdsk.sys	Kernel Driver	True	Normal	False
abp480n5	Abp480n5	abp480n5	Not Available	Kernel Driver	False	False
acpi	Microsoft ACPI Driver	c:\winnt\system32\drivers\acpi.sys	Kernel Driver	True	Running	OK
acpiec	Microsoft Embedded Controller Driver	c:\winnt\system32\drivers\acpiec.sys	Kernel Driver	True	Running	OK
adpu160m	adpu160m	c:\winnt\system32\drivers\adpu160m.sys	Kernel Driver	True	Running	OK
adpu320	adpu320	c:\winnt\system32\drivers\adpu320.sys	Kernel Driver	True	Running	OK
afd	AFD Networking Support Environment	c:\winnt\system32\drivers\afd.sys	Kernel Driver	True	Running	OK
aha154x	Aha154x	Not Available	Kernel Driver	False	Stopped	OK
aic116x	aic116x	Not Available	Kernel Driver	False	Stopped	OK
aic78u2	aic78u2	Not Available	Kernel Driver	False	Stopped	OK
aic78xx	aic78xx	Not Available	Kernel Driver	False	Stopped	OK
ami0nt	ami0nt	Not Available	Kernel Driver	False	Stopped	OK
amsint	amsint	Not Available	Kernel Driver	False	Stopped	OK
asc	asc	Not Available	Kernel Driver	False	Stopped	OK
asc3350p	asc3350p	Not Available	Kernel Driver	False	Stopped	OK
asc3550	asc3550	Not Available	Kernel Driver	False	Stopped	OK
asynmac	RAS Asynchronous Media Driver	c:\winnt\system32\drivers\asynmac.sys	Kernel Driver	False	Stopped	OK
atapi	Standard IDE/ESDI Hard Disk Controller	c:\winnt\system32\drivers\atapi.sys	Kernel Driver	True	Running	OK
atdisk	Atdisk	Not Available	Kernel Driver	False	Stopped	OK
atirage3	atirage3	c:\winnt\system32\drivers\atimpab.sys	Kernel Driver	False	Stopped	OK

atmarpc	Kernel Driver	True	Manual	Running OK	Ignore	False	True
	ATM ARP Client Protocol						
	c:\winnt\system32\drivers\atmarpc.sys	Kernel Driver	False				
audstub	Manual	Stopped OK	Normal	False	False		
	Audio Stub Driver		c:\winnt\system32\drivers\audstub.sys				
	Kernel Driver	True	Manual	Running OK	Normal	False	True
beep	Beep	c:\winnt\system32\drivers\beep.sys	Kernel Driver	True			
	System	Running OK	Normal	False	True		
buslogic	BusLogic	Not Available	Kernel Driver	False			
	Disabled	Stopped OK	Normal	False	False		
cd20xrnt	cd20xrnt	Not Available	Kernel Driver	False			
	Disabled	Stopped OK	Normal	False	False		
cdaudio	Cdaudio	c:\winnt\system32\drivers\cdaudio.sys	Kernel Driver	False			
	System	Stopped OK	Ignore	False	False		
cdfs	Cdfs	c:\winnt\system32\drivers\cdfs.sys	File System Driver				
	True	Disabled	Running OK	Normal	False	True	
cdrom	CD-ROM Driver	c:\winnt\system32\drivers\cdrom.sys	Kernel				
	Driver	True	System	Running OK	Normal	False	True
changer	Changer	Not Available	Kernel Driver	False	System	Stopped OK	
	Ignore	False	False				
cpqarray	Cpqarray	Not Available	Kernel Driver	False			
	Disabled	Stopped OK	Normal	False	False		
cpqarry2	cpqarry2	Not Available	Kernel Driver	False			
	Disabled	Stopped OK	Normal	False	False		
cpqfcalm	cpqfcalm	Not Available	Kernel Driver	False			
	Disabled	Stopped OK	Normal	False	False		
cpqfws2e	cpqfws2e	Not Available	Kernel Driver	False			
	Disabled	Stopped OK	Normal	False	False		
dac960nt	dac960nt	Not Available	Kernel Driver	False			
	Disabled	Stopped OK	Normal	False	False		
deckzpsx	deckzpsx	Not Available	Kernel Driver	False			
	Disabled	Stopped OK	Normal	False	False		
dfsdriver	DfsDriver	c:\winnt\system32\drivers\dfs.sys	File				
	System Driver	True	Boot	Running OK	Normal	False	True
disk	Disk Driver	c:\winnt\system32\drivers\disk.sys	Kernel				
	Driver	True	Boot	Running OK	Normal	False	True
diskperf	Diskperf	c:\winnt\system32\drivers\diskperf.sys					
	Kernel Driver	True	Boot	Running OK	Normal	False	True
dmboot	dmboot	c:\winnt\system32\drivers\dmboot.sys	Kernel Driver	False			
	Disabled	Stopped OK	Normal	False	False		
dmio	Logical Disk Manager Driver	c:\winnt\system32\drivers\dmio.sys					
	Kernel Driver	True	Boot	Running OK	Normal	False	True
dmload	dmload	c:\winnt\system32\drivers\dmload.sys	Kernel Driver	True			
	Boot	Running OK	Normal	False	True		
e1000	Intel(R) PRO/1000 Adapter Driver						
	c:\winnt\system32\drivers\e1000nt5.sys	Kernel Driver	True				
	Manual	Running OK	Normal	False	True		
e100b	Intel PRO Adapter Driver						
	c:\winnt\system32\drivers\e100bnt5.sys	Kernel Driver	True				
	Manual	Running OK	Normal	False	True		
efs	EFS	c:\winnt\system32\drivers\efs.sys	File System Driver				
	True	Disabled	Running OK	Normal	False	True	
fastfat	Fastfat	c:\winnt\system32\drivers\fastfat.sys	File System Driver				
	True	Disabled	Running OK	Normal	False	True	
fd16_700	Fd16_700	Not Available	Kernel Driver	False			
	Disabled	Stopped OK	Normal	False	False		
fdc	Floppy Disk Controller Driver	c:\winnt\system32\drivers\fdc.sys					
	Kernel Driver	True	Manual	Running OK	Normal	False	True
fips	Fips	c:\winnt\system32\drivers\fips.sys	Kernel Driver	True			

fireport	Auto	Running OK	Normal	False	True		
	fireport	Not Available	Kernel Driver	False			
flashpnt	Disabled	Stopped OK	Normal	False	False		
	flashpnt	Not Available	Kernel Driver	False			
flpydisk	Disabled	Stopped OK	Normal	False	False		
	Floppy Disk Driver						
	c:\winnt\system32\drivers\flpydisk.sys	Kernel Driver	True				
	Manual	Running OK	Normal	False	True		
ftdisk	Volume Manager Driver	c:\winnt\system32\drivers\ftdisk.sys					
	Kernel Driver	True	Boot	Running OK	Normal	False	True
gpc	Generic Packet Classifier	c:\winnt\system32\drivers\msgpc.sys					
	Kernel Driver	True	Manual	Running OK	Normal	False	True
i8042prt	i8042 Keyboard and PS/2 Mouse Port Driver						
	c:\winnt\system32\drivers\i8042prt.sys	Kernel Driver	True				
	System	Running OK	Normal	False	True		
ini910u	ini910u	Not Available	Kernel Driver	False	Disabled		
	Stopped OK	Normal	False	False			
intelide	IntelIde	Not Available	Kernel Driver	False			
	Disabled	Stopped OK	Normal	False	False		
ipfilterdriver	IP Traffic Filter Driver						
	c:\winnt\system32\drivers\ipfltdrv.sys	Kernel Driver	False				
	Manual	Stopped OK	Normal	False	False		
ipinip	IP in IP Tunnel Driver	c:\winnt\system32\drivers\ipinip.sys					
	Kernel Driver	False	Manual	Stopped OK	Normal	False	False
ipnat	IP Network Address Translator	c:\winnt\system32\drivers\ipnat.sys					
	Kernel Driver	False	Manual	Stopped OK	Normal	False	False
ipsec	IPSEC driver	c:\winnt\system32\drivers\ipsec.sys	Kernel				
	Driver	True	Manual	Running OK	Normal	False	True
ipsraidn	ipsraidn	Not Available	Kernel Driver	False			
	Disabled	Stopped OK	Normal	False	False		
isapnp	PnP ISA/EISA Bus Driver						
	c:\winnt\system32\drivers\isapnp.sys	Kernel Driver	True	Boot			
	Running OK	Critical	False	True			
kbdclass	Keyboard Class Driver						
	c:\winnt\system32\drivers\kbdclass.sys	Kernel Driver	True				
	System	Running OK	Normal	False	True		
ksecdd	KSecDD	c:\winnt\system32\drivers\ksecdd.sys	Kernel Driver	True			
	Boot	Running OK	Normal	False	True		
lbtfdc	lbtfdc	Not Available	Kernel Driver	False	System	Stopped OK	
	Ignore	False	False				
lp6nds35	lp6nds35	Not Available	Kernel Driver	False			
	Disabled	Stopped OK	Normal	False	False		
mnmd	mnmd	c:\winnt\system32\drivers\mnmd.sys	Kernel Driver	True			
	System	Running OK	Ignore	False	True		
modem	Modem	c:\winnt\system32\drivers\modem.sys	Kernel Driver	False			
	Manual	Stopped OK	Ignore	False	False		
mouclass	Mouse Class Driver						
	c:\winnt\system32\drivers\mouclass.sys	Kernel Driver	True				
	System	Running OK	Normal	False	True		
mountmgr	MountMgr	c:\winnt\system32\drivers\mountmgr.sys					
	Kernel Driver	True	Boot	Running OK	Normal	False	True
mraid35x	mraid35x	Not Available	Kernel Driver	False			
	Disabled	Stopped OK	Normal	False	False		
mrxsm	MRXSMB	c:\winnt\system32\drivers\mrxsm.sys	File System Driver				
	True	System	Running OK	Normal	False	True	
msfs	Msfs	c:\winnt\system32\drivers\msfs.sys	File System Driver				
	True	System	Running OK	Normal	False	True	
mskssrv	Microsoft Streaming Service Proxy						
	c:\winnt\system32\drivers\mskssrv.sys	Kernel Driver	False				

mspclock	Manual	Stopped	OK	Normal	False	False		
	Microsoft Streaming Clock Proxy							
	c:\winnt\system32\drivers\mspclock.sys	Kernel Driver	False					
mspqm	Manual	Stopped	OK	Normal	False	False		
	Microsoft Streaming Quality Manager Proxy							
	c:\winnt\system32\drivers\mspqm.sys	Kernel Driver	False					
mup	Manual	Stopped	OK	Normal	False	False		
	Mup	c:\winnt\system32\drivers\mup.sys	File System Driver					
	True	Boot	Running	OK	Normal	False	True	
ncrc710	NCrc710	Not Available	Kernel Driver	False	Disabled			
	Stopped	OK	Normal	False				
ndis	NDIS	System Driver	c:\winnt\system32\drivers\ndis.sys					
	Kernel Driver	True	Boot	Running	OK	Normal	False	True
ndistapi		Remote Access	NDIS TAPI Driver					
	c:\winnt\system32\drivers\ndistapi.sys	Kernel Driver	True					
	Manual	Running	OK	Normal	False	True		
ndiswan		Remote Access	NDIS WAN Driver					
	c:\winnt\system32\drivers\ndiswan.sys	Kernel Driver	True					
	Manual	Running	OK	Normal	False	True		
ndproxy	NDIS	Proxy	c:\winnt\system32\drivers\ndproxy.sys	Kernel				
	Driver	True	Manual	Running	OK	Normal	False	True
netbios	NetBIOS	Interface	c:\winnt\system32\drivers\netbios.sys	File				
	System Driver	True	System	Running	OK	Normal	False	True
netbt	NetBios	over Tcpip	c:\winnt\system32\drivers\netbt.sys					
	Kernel Driver	True	System	Running	OK	Normal	False	True
netdetect		NetDetect	c:\winnt\system32\drivers\netdect.sys					
	Kernel Driver	False	Manual	Stopped	OK	Normal	False	False
npfs	Npfs	c:\winnt\system32\drivers\npfs.sys	File System Driver					
	True	System	Running	OK	Normal	False	True	
ntfs	Ntfs	c:\winnt\system32\drivers\ntfs.sys	File System Driver					
	True	Disabled	Running	OK	Normal	False	True	
null	Null	c:\winnt\system32\drivers\null.sys	Kernel Driver	True				
	System	Running	OK	Normal	False	True		
nwlkflt		IPX Traffic Filter Driver						
	c:\winnt\system32\drivers\nwlkflt.sys	Kernel Driver	False					
	Manual	Stopped	OK	Normal	False	False		
nwlkfwd		IPX Traffic Forwarder Driver						
	c:\winnt\system32\drivers\nwlkfwd.sys	Kernel Driver	False					
	Manual	Stopped	OK	Normal	False	False		
parallel		Parallel	c:\winnt\system32\drivers\parallel.sys					
	Kernel Driver	False	Auto	Stopped	OK	Ignore	False	False
parport	Parport	c:\winnt\system32\drivers\parport.sys	Kernel Driver	False				
	Auto	Stopped	OK	Ignore	False	False		
partmgr	PartMgr	c:\winnt\system32\drivers\partmgr.sys	Kernel Driver	True				
	Boot	Running	OK	Normal	False	True		
parvdm	ParVdm	c:\winnt\system32\drivers\parvdm.sys	Kernel Driver	False				
	Auto	Stopped	OK	Ignore	False	False		
pci	PCI	Bus Driver	c:\winnt\system32\drivers\pci.sys	Kernel				
	Driver	True	Boot	Running	OK	Critical	False	True
pcidump	PCIDump	Not Available	Kernel Driver	False	System	Stopped	OK	
	Ignore	False	False					
pciide	PCIIde	c:\winnt\system32\drivers\pciide.sys	Kernel Driver	True				
	Boot	Running	OK	Normal	False	True		
pcmcia	Pcmcia	c:\winnt\system32\drivers\pcmcia.sys	Kernel Driver	False				
	Disabled	Stopped	OK	Normal	False	False		
pdcomp	PDCOMP	Not Available	Kernel Driver	False	Manual	Stopped	OK	
	Ignore	False	False					
pdframe	PDFRAME	Not Available	Kernel Driver	False	Manual	Stopped	OK	
	Ignore	False	False					

pdreli	PDRELI	Not Available	Kernel Driver	False	Manual	Stopped	OK	
	Ignore	False	False					
pdrframe		PDFRAME	Not Available	Kernel Driver	False			
	Manual	Stopped	OK	Ignore	False	False		
pptpminiport		WAN Miniport (PPTP)						
	c:\winnt\system32\drivers\rasppptp.sys	Kernel Driver	True					
	Manual	Running	OK	Normal	False	True		
ptilink	Direct	Parallel Link Driver						
	c:\winnt\system32\drivers\ptilink.sys	Kernel Driver	True					
	Manual	Running	OK	Normal	False	True		
ql1080	ql1080	Not Available	Kernel Driver	False	Disabled			
	Stopped	OK	Normal	False	False			
ql10wnt	Ql10wnt	Not Available	Kernel Driver	False	Disabled			
	Stopped	OK	Normal	False	False			
ql1240	ql1240	Not Available	Kernel Driver	False	Disabled			
	Stopped	OK	Normal	False	False			
ql2100	ql2100	Not Available	Kernel Driver	False	Disabled			
	Stopped	OK	Normal	False	False			
ql2300	ql2300	c:\winnt\system32\drivers\ql2300.sys	Kernel Driver	True				
	Boot	Running	OK	Normal	False	True		
rasacd		Remote Access Auto Connection Driver						
	c:\winnt\system32\drivers\rasacd.sys	Kernel Driver	True					
	System	Running	OK	Normal	False	True		
rasl2tp		WAN Miniport (L2TP)	c:\winnt\system32\drivers\rasl2tp.sys					
	Kernel Driver	True	Manual	Running	OK	Normal	False	True
raspti	Direct	Parallel	c:\winnt\system32\drivers\raspti.sys					
	Kernel Driver	True	Manual	Running	OK	Normal	False	True
rca		Microsoft Streaming Network Raw Channel Access						
	c:\winnt\system32\drivers\rca.sys	Kernel Driver	False					
	Manual	Stopped	OK	Normal	False	False		
rdbss	Rdbss	c:\winnt\system32\drivers\rdbss.sys	File System Driver					
	True	System	Running	OK	Normal	False	True	
rdpwd	RDPWD	c:\winnt\system32\drivers\rdpwd.sys	Kernel Driver	False				
	Manual	Stopped	OK	Ignore	False	False		
redbook		Digital CD Audio Playback Filter Driver						
	c:\winnt\system32\drivers\redbook.sys	Kernel Driver	False					
	System	Stopped	OK	Normal	False	False		
serenum	Serenum	Filter Driver	c:\winnt\system32\drivers\serenum.sys					
	Kernel Driver	True	Manual	Running	OK	Normal	False	True
serial		Serial port driver	c:\winnt\system32\drivers\serial.sys					
	Kernel Driver	True	System	Running	OK	Ignore	False	True
sfloppy	Sfloppy	c:\winnt\system32\drivers\sfloppy.sys	Kernel Driver	False				
	System	Stopped	OK	Ignore	False	False		
sglfb	sglfb	Not Available	Kernel Driver	False	System	Stopped	OK	
	Normal	False	False					
simbad	Simbad	Not Available	Kernel Driver	False	Disabled			
	Stopped	OK	Normal	False	False			
sparrow	Sparrow	Not Available	Kernel Driver	False	Disabled			
	Stopped	OK	Normal	False	False			
spud		Special Purpose Utility Driver						
	c:\winnt\system32\drivers\spud.sys	Kernel Driver	True					
	Manual	Running	OK	Normal	False	True		
srv	Srv	c:\winnt\system32\drivers\srv.sys	File System Driver					
	True	Manual	Running	OK	Normal	False	True	
swenum		Software Bus Driver	c:\winnt\system32\drivers\swenum.sys					
	Kernel Driver	True	Manual	Running	OK	Normal	False	True
symc810	symc810	Not Available	Kernel Driver	False	Disabled			
	Stopped	OK	Normal	False	False			
symc8xx	symc8xx	Not Available	Kernel Driver	False	Disabled			

```

sym_hi      Stopped OK      Normal False False
sym_hi      sym_hi Not Available Kernel Driver False Disabled
tcpip       TCP/IP Protocol Driver c:\winnt\system32\drivers\tcpip.sys
Kernel Driver True System Running OK Normal False True
tdasync     TDASYNC c:\winnt\system32\drivers\tdasync.sys Kernel Driver False
Manual Stopped OK Ignore False False
tdipx       TDIPX c:\winnt\system32\drivers\tdipx.sys Kernel Driver False
Manual Stopped OK Ignore False False
tdnetb      TDNETB c:\winnt\system32\drivers\tdnetb.sys Kernel Driver False
Manual Stopped OK Ignore False False
tdpipe      TDPIPE c:\winnt\system32\drivers\tdpipe.sys Kernel Driver False
Manual Stopped OK Ignore False False
tdspx       TDSPX c:\winnt\system32\drivers\tdspx.sys Kernel Driver False
Manual Stopped OK Ignore False False
tdtcp       TDTCP c:\winnt\system32\drivers\tdtcp.sys Kernel Driver False
Manual Stopped OK Ignore False False
termdd      Terminal Device Driver c:\winnt\system32\drivers\termdd.sys
Kernel Driver False Disabled Stopped OK Normal False
False
tga         tga Not Available Kernel Driver False System Stopped OK
Ignore False False
udfs        Udfs c:\winnt\system32\drivers\udfs.sys File System Driver
False Disabled Stopped OK Normal False False
uhcd        Microsoft USB Universal Host Controller Driver
c:\winnt\system32\drivers\uhcd.sys Kernel Driver True
Manual Running OK Normal False True
ultra66     Ultra66 Not Available Kernel Driver False Disabled
Stopped OK Normal False False
update      Microcode Update Driver
c:\winnt\system32\drivers\update.sys Kernel Driver True
Manual Running OK Normal False True
usbhub      Microsoft USB Standard Hub Driver
c:\winnt\system32\drivers\usbhub.sys Kernel Driver True
Manual Running OK Normal False True
usbstor     USB Mass Storage Driver
c:\winnt\system32\drivers\usbstor.sys Kernel Driver False
Manual Stopped OK Normal False False
vgasave     VgaSave c:\winnt\system32\drivers\vga.sys Kernel Driver True
System Running OK Ignore False True
wanarp      Remote Access IP ARP Driver
c:\winnt\system32\drivers\wanarp.sys Kernel Driver True
Manual Running OK Normal False True
wdica       WDICA Not Available Kernel Driver False Manual Stopped OK
Ignore False False

```

[Environment Variables]

```

Variable      Value      User Name
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:\SQL
LIB\java\common.jar <SYSTEM>
ComSpec %SystemRoot%\system32\cmd.exe <SYSTEM>
DB2INSTANCE   nec <SYSTEM>
DB2TEMPDIR    C:\SQLLIB\ <SYSTEM>
INCLUDE C:\SQLLIB\INCLUDE;C:\SQLLIB\LIB <SYSTEM>
LIB           C:\SQLLIB\LIB <SYSTEM>
LOCAL        ASAMA <SYSTEM>

```

```

NUMBER_OF_PROCESSORS 4 <SYSTEM>
OS Windows_NT <SYSTEM>
Os2LibPath %SystemRoot%\system32\os2\dll; <SYSTEM>
Path        C:\Perl\bin\;%SystemRoot%\system32;%SystemRoot%;%SystemRoot%\Syst
em32\Wbem;C:\SQLLIB\BIN;C:\SQLLIB\FUNCTION;C:\SQLLIB\SAMPLES\REPL
<SYSTEM>
PATHEXT .COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH <SYSTEM>
PROCESSOR_ARCHITECTURE x86 <SYSTEM>
PROCESSOR_IDENTIFIER x86 Family 15 Model 2 Stepping 7, GenuineIntel
<SYSTEM>
PROCESSOR_LEVEL 15 <SYSTEM>
PROCESSOR_REVISION 0207 <SYSTEM>
TEMP %SystemRoot%\TEMP <SYSTEM>
TMCONTEXTS 1 <SYSTEM>
TMP %SystemRoot%\TEMP <SYSTEM>
TPCC_DBNAME tpcc <SYSTEM>
windir %SystemRoot% <SYSTEM>
DB2INSTANCE nec ACL37\nec
include C:\Program Files\Microsoft Visual
Studio\VC98\atl\include;C:\Program Files\Microsoft Visual
Studio\VC98\mfcc\include;C:\Program Files\Microsoft Visual
Studio\VC98\include ACL37\nec
lib C:\Program Files\Microsoft Visual Studio\VC98\mfcc\lib;C:\Program
Files\Microsoft Visual Studio\VC98\lib ACL37\nec
MSDevDir C:\Program Files\Microsoft Visual Studio\Common\MSDev98
ACL37\nec
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
ACL37\nec
TEMP %USERPROFILE%\Local Settings\Temp ACL37\nec
TMP %USERPROFILE%\Local Settings\Temp ACL37\nec
TPCC_DBNAME tpcc ACL37\nec
LOCAL ASAMA ACL37\Administrator
PATH C:\%PATH% ACL37\Administrator
TEMP %USERPROFILE%\Local Settings\Temp ACL37\Administrator
TMP %USERPROFILE%\Local Settings\Temp ACL37\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	
Queue	Parameters	Driver Name	Print Processor	Host	Print	
No print jobs	Data Type	Name				

[Network Connections]

Local Name	Remote Name	Type	Status	User Name
F:	\\10.1.1.48\c\$	Disk	OK	ACL37\nec

[Running Tasks]

Name	Path	Process ID	Priority	Min Working Set	Max Working Set
Working Set	Start Time	Version Size	File Date	File Date	File Date
system idle process	Not Available	0	0	Not Available	Not Available
system	Not Available	8	0	1413120	Not Available
smss.exe	c:\winnt\system32\smss.exe	172	11	204800	
1413120	12/24/2004 5:21:19 PM	5.00.2195.2901	44.27 KB	(45,328 bytes)	12/8/1999 5:00:00 AM
csrss.exe	Not Available	200	13	Not Available	Not Available
winlogon.exe	c:\winnt\system32\winlogon.exe	196	13	204800	
204800	1413120 12/24/2004 5:21:29 PM	5.00.2195.2953	173.77 KB	(177,936 bytes)	12/8/1999 5:00:00 AM
services.exe	c:\winnt\system32\services.exe	248	9	204800	
204800	1413120 12/24/2004 5:21:30 PM	5.00.2195.2780	86.77 KB	(88,848 bytes)	12/8/1999 5:00:00 AM
lsass.exe	c:\winnt\system32\lsass.exe	260	9	204800	
1413120	12/24/2004 5:21:30 PM	5.00.2195.2964	32.77 KB	(33,552 bytes)	12/8/1999 5:00:00 AM
svchost.exe	c:\winnt\system32\svchost.exe	424	8	204800	
1413120	12/24/2004 5:21:32 PM	5.00.2134.1	7.77 KB	(7,952 bytes)	12/8/1999 5:00:00 AM
spoolsv.exe	c:\winnt\system32\spoolsv.exe	456	8	204800	
1413120	12/24/2004 5:21:32 PM	5.00.2161.1	43.77 KB	(44,816 bytes)	1/8/2003 2:27:15 AM
msdtc.exe	c:\winnt\system32\msdtc.exe	484	8	204800	
1413120	12/24/2004 5:21:32 PM	1999.9.3421.3	6.77 KB	(6,928 bytes)	1/8/2003 2:35:59 AM
db2dasrrm.exe	Not Available	648	8	Not Available	Not Available
db2jds.exe	c:\sql\bin\db2jds.exe	664	8	204800	
1413120	12/24/2004 5:21:34 PM	8.1.5.449	189.12 KB	(193,656 bytes)	2/21/2004 2:15:38 AM
db2sec.exe	c:\sql\bin\db2sec.exe	712	8	204800	
1413120	12/24/2004 5:21:35 PM	8.1.5.449	29.12 KB	(29,816 bytes)	2/21/2004 2:17:08 AM
db2rcmd.exe	Not Available	724	8	Not Available	Not Available
svchost.exe	c:\winnt\system32\svchost.exe	740	8	204800	
1413120	12/24/2004 5:21:36 PM	5.00.2134.1	7.77 KB	(7,952 bytes)	12/8/1999 5:00:00 AM
llssrv.exe	c:\winnt\system32\llssrv.exe	776	9	204800	
1413120	12/24/2004 5:21:36 PM	5.00.2195.2649	114.27 KB	(117,008 bytes)	5/4/2001 12:05:02 PM
regsvc.exe	c:\winnt\system32\regsvc.exe	820	8	204800	
1413120	12/24/2004 5:21:36 PM	5.00.2195.2104	65.27 KB	(66,832 bytes)	1/7/2003 5:54:51 PM
mstask.exe	c:\winnt\system32\mstask.exe	864	8	204800	
1413120	12/24/2004 5:21:37 PM	4.71.2195.1	115.27 KB	(118,032 bytes)	1/7/2003 5:54:43 PM
winmgmt.exe	c:\winnt\system32\wbem\winmgmt.exe	900	8	204800	
204800	1413120 12/24/2004 5:21:37 PM	1.50.1085.0029	192.08 KB	(196,685 bytes)	1/7/2003 5:55:03 PM
inetinfo.exe	c:\winnt\system32\inet\inetinfo.exe	148	8	204800	
204800	1413120 12/24/2004 5:21:44 PM	5.00.0984	14.27 KB	(14,608 bytes)	1/7/2003 5:57:06 PM
dfssvc.exe	c:\winnt\system32\dfssvc.exe	1092	8	204800	

1413120	12/24/2004 5:21:51 PM	5.00.2195.2841	88.27 KB	(90,384 bytes)	1/7/2003 5:54:30 PM
svchost.exe	c:\winnt\system32\svchost.exe	1356	8	204800	
1413120	12/24/2004 5:22:22 PM	5.00.2134.1	7.77 KB	(7,952 bytes)	12/8/1999 5:00:00 AM
rsrvp.exe	c:\winnt\system32\rsrvp.exe	1136	8	204800	
1413120	12/24/2004 5:23:11 PM	5.00.2167.1	172.77 KB	(176,912 bytes)	12/8/1999 5:00:00 AM
explorer.exe	c:\winnt\explorer.exe	1500	8	204800	
12/24/2004	5:25:29 PM	5.00.3315.2846	237.27 KB	(242,960 bytes)	1/7/2003 5:54:59 PM
mmc.exe	c:\winnt\system32\mmc.exe	1248	8	204800	
12/24/2004	5:25:42 PM	5.00.2195.2301	589.27 KB	(603,408 bytes)	1/7/2003 5:54:38 PM

[Loaded Modules]

Name	Version	Size	File Date	Manufacturer	Path
wbemprox.dll	1.50.1085.0045	40.08 KB	(41,040 bytes)	1/7/2003	
5:55:03 PM	Microsoft Corporation	c:\winnt\system32\wbem\wbemprox.dll			
mlang.dll	5.00.3103.1000	510.77 KB	(523,024 bytes)	1/7/2003	
5:54:38 PM	Microsoft Corporation	c:\winnt\system32\mlang.dll			
cabinet.dll	5.00.2147.1	54.77 KB	(56,080 bytes)	12/8/1999	
5:00:00 AM	Microsoft Corporation	c:\winnt\system32\cabinet.dll			
msinfo32.dll	5.00.2177.1	312.27 KB	(319,760 bytes)	1/7/2003	
5:40:42 PM	Microsoft Corporation	c:\program files\common files\microsoft shared\msinfo\msinfo32.dll			
mmcndmgr.dll	5.00.2178.1	815.27 KB	(834,832 bytes)	12/8/1999	
5:00:00 AM	Microsoft Corporation	c:\winnt\system32\mmcndmgr.dll			
mmc.exe	5.00.2195.2301	589.27 KB	(603,408 bytes)	1/7/2003 5:54:38 PM	
Microsoft Corporation	c:\winnt\system32\mmc.exe				
urlmon.dll	5.00.3315.1000	441.27 KB	(451,856 bytes)	1/7/2003	
5:54:56 PM	Microsoft Corporation	c:\winnt\system32\urlmon.dll			
faxshell.dll	5.00.2134.1	8.27 KB	(8,464 bytes)	12/8/1999 5:00:00 AM	
Microsoft Corporation	c:\winnt\system32\faxshell.dll				
msacm32.dll	5.00.2134.1	65.27 KB	(66,832 bytes)	12/8/1999	
5:00:00 AM	Microsoft Corporation	c:\winnt\system32\msacm32.dll			
avifil32.dll	5.00.2134.1	76.27 KB	(78,096 bytes)	12/8/1999	
5:00:00 AM	Microsoft Corporation	c:\winnt\system32\avifil32.dll			
msvfw32.dll	5.00.2134.1	113.77 KB	(116,496 bytes)	12/8/1999	
5:00:00 AM	Microsoft Corporation	c:\winnt\system32\msvfw32.dll			
docprop2.dll	5.00.2178.1	297.77 KB	(304,912 bytes)	12/8/1999	
5:00:00 AM	Microsoft Corporation	c:\winnt\system32\docprop2.dll			
browseic.dll	5.00.3315.2846	34.50 KB	(35,328 bytes)	1/7/2003	
5:54:27 PM	Microsoft Corporation	c:\winnt\system32\browseic.dll			
wininet.dll	5.00.3315.1000	456.77 KB	(467,728 bytes)	1/7/2003	
5:54:57 PM	Microsoft Corporation	c:\winnt\system32\wininet.dll			
linkinfo.dll	5.00.2134.1	15.77 KB	(16,144 bytes)	12/8/1999	
5:00:00 AM	Microsoft Corporation	c:\winnt\system32\linkinfo.dll			
powrprof.dll	5.00.3103.1000	13.27 KB	(13,584 bytes)	1/7/2003	
5:54:50 PM	Microsoft Corporation	c:\winnt\system32\powrprof.dll			
batmeter.dll	5.00.3103.1000	20.27 KB	(20,752 bytes)	1/7/2003	
5:54:27 PM	Microsoft Corporation	c:\winnt\system32\batmeter.dll			
stobject.dll	5.00.2195.2780	79.27 KB	(81,168 bytes)	1/7/2003	
5:54:55 PM	Microsoft Corporation	c:\winnt\system32\stobject.dll			
webcheck.dll	5.00.3315.1000	251.77 KB	(257,808 bytes)	1/7/2003	
5:54:57 PM	Microsoft Corporation	c:\winnt\system32\webcheck.dll			
ntshrui.dll	5.00.2134.1	46.77 KB	(47,888 bytes)	12/8/1999	
5:00:00 AM	Microsoft Corporation	c:\winnt\system32\ntshrui.dll			

mydocs.dll	5.00.2920.0000	55.77 KB (57,104 bytes)	12/8/1999	c:\winnt\system32\inetsrv\admexs.dll		
5:00:00 AM	Microsoft Corporation	c:\winnt\system32\mydocs.dll		wamreg.dll	5.00.0984	45.77 KB (46,864 bytes) 1/7/2003
browseui.dll	5.00.3315.2846	788.77 KB (807,696 bytes)	1/7/2003	5:56:43 PM	Microsoft Corporation	
5:54:27 PM	Microsoft Corporation	c:\winnt\system32\browseui.dll			c:\winnt\system32\inetsrv\wamreg.dll	
shdocvw.dll	5.00.3315.2879	1.05 MB (1,104,144 bytes)	1/7/2003	metadata.dll	5.00.0984	68.77 KB (70,416 bytes) 1/7/2003
5:54:53 PM	Microsoft Corporation	c:\winnt\system32\shdocvw.dll		5:56:51 PM	Microsoft Corporation	
explorer.exe	5.00.3315.2846	237.27 KB (242,960 bytes)	1/7/2003		c:\winnt\system32\inetsrv\metadata.dll	
5:54:59 PM	Microsoft Corporation	c:\winnt\explorer.exe		iismap.dll	5.00.0984	55.77 KB (57,104 bytes) 1/7/2003
traffic.dll	5.00.2139.1	30.77 KB (31,504 bytes)	12/8/1999	5:56:59 PM	Microsoft Corporation	c:\winnt\system32\iismap.dll
5:00:00 AM	Microsoft Corporation	c:\winnt\system32\traffic.dll		nsepm.dll	5.00.0984	43.27 KB (44,304 bytes) 1/7/2003
rsvp.exe	5.00.2167.1	172.77 KB (176,912 bytes)	12/8/1999	5:56:34 PM	Microsoft Corporation	c:\winnt\system32\inetsrv\nsepm.dll
5:00:00 AM	Microsoft Corporation	c:\winnt\system32\rsvp.exe		coadmin.dll	5.00.0984	39.27 KB (40,208 bytes) 1/7/2003
tapisrv.dll	5.00.2195.2955	169.27 KB (173,328 bytes)	1/7/2003	5:57:06 PM	Microsoft Corporation	
5:54:55 PM	Microsoft Corporation	c:\winnt\system32\tapisrv.dll			c:\winnt\system32\inetsrv\coadmin.dll	
dfssvc.exe	5.00.2195.2841	88.27 KB (90,384 bytes)	1/7/2003	iisadmin.dll	5.00.0984	15.27 KB (15,632 bytes) 1/7/2003
5:54:30 PM	Microsoft Corporation	c:\winnt\system32\dfssvc.exe		5:57:00 PM	Microsoft Corporation	
iislog.dll	5.00.0984	75.27 KB (77,072 bytes)	1/7/2003		c:\winnt\system32\inetsrv\iisadmin.dll	
5:56:59 PM	Microsoft Corporation			rpcref.dll	5.00.0984	4.27 KB (4,368 bytes) 1/7/2003 5:56:47 PM
		c:\winnt\system32\inetsrv\iislog.dll			Microsoft Corporation	c:\winnt\system32\inetsrv\rpcref.dll
ntlsapi.dll	5.00.2134.1	6.77 KB (6,928 bytes)	12/8/1999 5:00:00 AM	iisrtl.dll	5.00.0984	119.77 KB (122,640 bytes) 1/7/2003
	Microsoft Corporation	c:\winnt\system32\ntlsapi.dll		5:56:59 PM	Microsoft Corporation	c:\winnt\system32\iisrtl.dll
httpext.dll	0.9.3940.21	435.27 KB (445,712 bytes)	1/7/2003	inetinfo.exe	5.00.0984	14.27 KB (14,608 bytes) 1/7/2003
5:57:08 PM	Microsoft Corporation			5:57:06 PM	Microsoft Corporation	
		c:\winnt\system32\inetsrv\httpext.dll			c:\winnt\system32\inetsrv\inetinfo.exe	
md5filt.dll	5.00.0984	32.77 KB (33,552 bytes)	1/7/2003	perfos.dll	5.00.2155.1	21.27 KB (21,776 bytes) 12/8/1999
5:56:43 PM	Microsoft Corporation			5:00:00 AM	Microsoft Corporation	c:\winnt\system32\perfos.dll
		c:\winnt\system32\inetsrv\md5filt.dll		mfc42.dll	6.00.8665.0	972.05 KB (995,383 bytes) 12/8/1999
gzip.dll	5.00.0984	30.27 KB (30,992 bytes)	1/7/2003	5:00:00 AM	Microsoft Corporation	c:\winnt\system32\mfc42.dll
5:56:28 PM	Microsoft Corporation	c:\winnt\system32\inetsrv\gzip.dll		ieinfo5.ocx	5.00.2134.1	96.27 KB (98,576 bytes) 1/7/2003
compfilt.dll	5.00.0984	22.77 KB (23,312 bytes)	1/7/2003	5:40:17 PM	Microsoft Corporation	
5:57:07 PM	Microsoft Corporation				c:\program~1\common~1\microso~1\msinfo\ieinfo5.ocx	
		c:\winnt\system32\inetsrv\compfilt.dll		wshnetbs.dll	5.00.2134.1	7.77 KB (7,952 bytes) 12/8/1999 5:00:00 AM
sspifilt.dll	5.00.0984	43.27 KB (44,304 bytes)	1/7/2003		Microsoft Corporation	c:\winnt\system32\wshnetbs.dll
5:56:32 PM	Microsoft Corporation			ntmarta.dll	5.00.2195.2862	98.77 KB (101,136 bytes) 1/7/2003
		c:\winnt\system32\inetsrv\sspifilt.dll		5:54:47 PM	Microsoft Corporation	c:\winnt\system32\ntmarta.dll
iscomlog.dll	5.00.0984	24.77 KB (25,360 bytes)	1/7/2003	provthrd.dll	1.50.1085.0000	68.07 KB (69,708 bytes) 1/7/2003
5:57:01 PM	Microsoft Corporation			5:40:31 PM	Microsoft Corporation	c:\winnt\system32\wbem\provthrd.dll
		c:\winnt\system32\inetsrv\iscomlog.dll		ntevt.dll	1.50.1085.0000	192.06 KB (196,669 bytes) 12/8/1999
lonsint.dll	5.00.0984	11.77 KB (12,048 bytes)	1/7/2003	5:00:00 AM	Microsoft Corporation	c:\winnt\system32\wbem\ntevt.dll
5:56:31 PM	Microsoft Corporation			psapi.dll	5.00.2134.1	28.27 KB (28,944 bytes) 12/8/1999
		c:\winnt\system32\inetsrv\lonsint.dll		5:00:00 AM	Microsoft Corporation	c:\winnt\system32\psapi.dll
inetsloc.dll	5.00.0984	20.27 KB (20,752 bytes)	1/7/2003	framedyn.dll	1.50.1085.0000	164.05 KB (167,992 bytes) 12/8/1999
5:57:06 PM	Microsoft Corporation	c:\winnt\system32\inetsloc.dll		5:00:00 AM	Microsoft Corporation	c:\winnt\system32\wbem\framedyn.dll
iisfecnv.dll	5.00.0984	7.27 KB (7,440 bytes)	7/17/2003 10:57:40 PM	cimwin32.dll	1.50.1085.0038	1.02 MB (1,073,232 bytes) 1/7/2003
	Microsoft Corporation	c:\winnt\system32\inetsrv\iisfecnv.dll		5:55:02 PM	Microsoft Corporation	c:\winnt\system32\wbem\cimwin32.dll
isatq.dll	5.00.0984	60.27 KB (61,712 bytes)	1/7/2003	wbemsvc.dll	1.50.1085.0007	40.07 KB (41,036 bytes) 1/7/2003
5:56:36 PM	Microsoft Corporation	c:\winnt\system32\inetsrv\isatq.dll		5:55:03 PM	Microsoft Corporation	c:\winnt\system32\wbem\wbemsvc.dll
infocomm.dll	5.00.0984	238.27 KB (243,984 bytes)	1/7/2003	wbemess.dll	1.50.1085.0039	364.07 KB (372,804 bytes) 1/7/2003
5:57:06 PM	Microsoft Corporation			5:55:03 PM	Microsoft Corporation	c:\winnt\system32\wbem\wbemess.dll
		c:\winnt\system32\inetsrv\infocomm.dll		fastprox.dll	1.50.1085.0037	144.08 KB (147,536 bytes) 1/7/2003
w3svc.dll	5.00.0984	343.27 KB (351,504 bytes)	1/7/2003	5:55:02 PM	Microsoft Corporation	c:\winnt\system32\wbem\fastprox.dll
5:56:38 PM	Microsoft Corporation	c:\winnt\system32\inetsrv\w3svc.dll		wbemcore.dll	1.50.1085.0036	628.07 KB (643,140 bytes) 1/7/2003
security.dll	5.00.2154.1	5.77 KB (5,904 bytes)	12/8/1999 5:00:00 AM	5:55:03 PM	Microsoft Corporation	c:\winnt\system32\wbem\wbemcore.dll
	Microsoft Corporation	c:\winnt\system32\security.dll		wbemcomm.dll	1.50.1085.0021	692.07 KB (708,675 bytes) 1/7/2003
svcxext.dll	5.00.0984	39.77 KB (40,720 bytes)	1/7/2003	5:55:03 PM	Microsoft Corporation	c:\winnt\system32\wbem\wbemcomm.dll
5:56:45 PM	Microsoft Corporation			winmgmt.exe	1.50.1085.0029	192.08 KB (196,685 bytes) 1/7/2003
		c:\winnt\system32\inetsrv\svcxext.dll		5:55:03 PM	Microsoft Corporation	c:\winnt\system32\wbem\winmgmt.exe
admexs.dll	5.00.0984	27.77 KB (28,432 bytes)	1/7/2003	msidle.dll	5.00.2920.0000	6.27 KB (6,416 bytes) 12/8/1999 5:00:00 AM
5:57:08 PM	Microsoft Corporation				Microsoft Corporation	c:\winnt\system32\msidle.dll

mstask.exe 4.71.2195.1 115.27 KB (118,032 bytes) 1/7/2003
5:54:43 PM Microsoft Corporation c:\winnt\system32\mstask.exe
regsvcs.exe 5.00.2195.2104 65.27 KB (66,832 bytes) 1/7/2003
5:54:51 PM Microsoft Corporation c:\winnt\system32\regsvcs.exe
llsrpc.dll 5.00.2149.1 45.77 KB (46,864 bytes) 12/8/1999
5:00:00 AM Microsoft Corporation c:\winnt\system32\llsrpc.dll
llssrv.exe 5.00.2195.2649 114.27 KB (117,008 bytes) 5/4/2001
12:05:02 PM Microsoft Corporation c:\winnt\system32\llssrv.exe
rasdlg.dll 5.00.2195.2671 514.27 KB (526,608 bytes) 12/8/1999
5:00:00 AM Microsoft Corporation c:\winnt\system32\rasdlg.dll
netcfgx.dll 5.00.2195.2228 534.77 KB (547,600 bytes) 1/7/2003
5:54:45 PM Microsoft Corporation c:\winnt\system32\netcfgx.dll
rasmans.dll 5.00.2195.2728 147.27 KB (150,800 bytes) 1/7/2003
5:54:51 PM Microsoft Corporation c:\winnt\system32\rasmans.dll
ntmsdba.dll 5.00.2195.2779 167.27 KB (171,280 bytes) 1/7/2003
5:54:47 PM Microsoft Corporation c:\winnt\system32\ntmsdba.dll
netshell.dll 5.00.2195.2779 457.27 KB (468,240 bytes) 1/7/2003
5:54:46 PM Microsoft Corporation c:\winnt\system32\netshell.dll
netman.dll 5.00.2195.2779 89.27 KB (91,408 bytes) 1/7/2003
5:54:46 PM Microsoft Corporation c:\winnt\system32\netman.dll
sens.dll 5.00.2163.1 36.77 KB (37,648 bytes) 12/8/1999
5:00:00 AM Microsoft Corporation c:\winnt\system32\sens.dll
ntmssvc.dll 5.00.2195.2779 391.27 KB (400,656 bytes) 1/7/2003
5:54:47 PM Microsoft Corporation c:\winnt\system32\ntmssvc.dll
es.dll 2000.2.3471.1 222.27 KB (227,600 bytes) 1/7/2003 5:54:33 PM
Microsoft Corporation c:\winnt\system32\es.dll
db2sec.exe 8.1.5.449 29.12 KB (29,816 bytes) 2/21/2004
2:17:08 AM International Business Machines Corporation
c:\sqllib\bin\db2sec.exe
db2cli.dll 8.1.5.449 2.68 MB (2,805,825 bytes) 2/20/2004
11:15:20 PM International Business Machines Corporation
c:\sqllib\bin\db2cli.dll
db2abind.dll 8.1.5.449 244.07 KB (249,923 bytes) 2/20/2004
11:15:18 PM International Business Machines Corporation
c:\sqllib\bin\db2abind.dll
db2util.dll 8.1.5.449 1.18 MB (1,237,058 bytes) 2/20/2004
11:15:52 PM International Business Machines Corporation
c:\sqllib\bin\db2util.dll
db2install.dll 8.1.5.449 28.07 KB (28,741 bytes) 2/20/2004
11:13:36 PM International Business Machines Corporation
c:\sqllib\bin\db2install.dll
db2trcapi.dll 8.1.5.449 36.08 KB (36,941 bytes) 2/20/2004
11:13:36 PM International Business Machines Corporation
c:\sqllib\bin\db2trcapi.dll
db2locale.dll 8.1.5.449 48.07 KB (49,220 bytes) 2/20/2004
11:13:36 PM International Business Machines Corporation
c:\sqllib\bin\db2locale.dll
db2osse.dll 8.1.5.449 312.07 KB (319,564 bytes) 2/20/2004
11:15:54 PM International Business Machines Corporation
c:\sqllib\bin\db2osse.dll
db2g11n.dll 8.1.5.449 364.06 KB (372,802 bytes) 2/20/2004
11:13:36 PM International Business Machines Corporation
c:\sqllib\bin\db2g11n.dll
db2sysp.dll 8.1.5.449 88.07 KB (90,179 bytes) 2/20/2004
11:15:52 PM International Business Machines Corporation
c:\sqllib\bin\db2sysp.dll
db2wint.dll 8.1.5.449 48.06 KB (49,218 bytes) 2/20/2004
11:15:52 PM International Business Machines Corporation
c:\sqllib\bin\db2wint.dll

db2sys.dll 8.1.5.449 2.38 MB (2,490,433 bytes) 2/20/2004
11:15:52 PM International Business Machines Corporation
c:\sqllib\bin\db2sys.dll
db2app.dll 8.1.5.449 2.49 MB (2,613,313 bytes) 2/20/2004
11:15:20 PM International Business Machines Corporation
c:\sqllib\bin\db2app.dll
db2jds.exe 8.1.5.449 189.12 KB (193,656 bytes) 2/21/2004
2:15:38 AM International Business Machines Corporation
c:\sqllib\bin\db2jds.exe
mtxoci.dll 2000.2.3471.1 101.77 KB (104,208 bytes) 1/7/2003
5:54:45 PM Microsoft Corporation c:\winnt\system32\mtxoci.dll
resutils.dll 5.00.2195.2787 39.77 KB (40,720 bytes) 1/7/2003
5:54:51 PM Microsoft Corporation c:\winnt\system32\resutils.dll
clusapi.dll 5.00.2195.2104 54.27 KB (55,568 bytes) 1/7/2003
5:54:28 PM Microsoft Corporation c:\winnt\system32\clusapi.dll
msvcps50.dll 5.00.7051 552.50 KB (565,760 bytes) 12/8/1999
5:00:00 AM Microsoft Corporation c:\winnt\system32\msvcps50.dll
xolehlp.dll 1999.9.3421.3 17.27 KB (17,680 bytes) 1/8/2003
2:36:00 AM Microsoft Corporation c:\winnt\system32\xolehlp.dll
msdtclog.dll 1999.9.3421.3 89.77 KB (91,920 bytes) 1/8/2003
2:35:59 AM Microsoft Corporation c:\winnt\system32\msdtclog.dll
mtxclu.dll 2000.2.3471.1 51.27 KB (52,496 bytes) 1/7/2003
5:54:45 PM Microsoft Corporation c:\winnt\system32\mtxclu.dll
msdtcprx.dll 2000.2.3471.1 665.77 KB (681,744 bytes) 1/7/2003
5:54:39 PM Microsoft Corporation c:\winnt\system32\msdtcprx.dll
txfaux.dll 2000.2.3471.1 374.27 KB (383,248 bytes) 1/7/2003
5:54:56 PM Microsoft Corporation c:\winnt\system32\txfaux.dll
msdtctm.dll 2000.2.3471.1 1.07 MB (1,120,528 bytes) 1/7/2003
5:54:39 PM Microsoft Corporation c:\winnt\system32\msdtctm.dll
msdtc.exe 1999.9.3421.3 6.77 KB (6,928 bytes) 1/8/2003 2:35:59 AM
Microsoft Corporation c:\winnt\system32\msdtc.exe
wmi.dll 5.00.2191.1 6.27 KB (6,416 bytes) 12/8/1999 5:00:00 AM
Microsoft Corporation c:\winnt\system32\wmi.dll
admwprox.dll 5.00.0984 31.77 KB (32,528 bytes) 7/17/2003
10:57:40 PM Microsoft Corporation c:\winnt\system32\admwprox.dll
inetpp.dll 5.00.2195.2842 65.27 KB (66,832 bytes) 1/7/2003
5:54:35 PM Microsoft Corporation c:\winnt\system32\inetpp.dll
win32spl.dll 5.00.2195.2780 92.27 KB (94,480 bytes) 12/8/1999
5:00:00 AM Microsoft Corporation c:\winnt\system32\win32spl.dll
usbmon.dll 5.00.2195.2780 11.27 KB (11,536 bytes) 1/7/2003
5:54:56 PM Microsoft Corporation c:\winnt\system32\usbmon.dll
tcpmon.dll 5.00.2195.2780 40.77 KB (41,744 bytes) 1/7/2003
5:54:55 PM Microsoft Corporation c:\winnt\system32\tcpmon.dll
pjlmon.dll 5.00.2165.1 12.77 KB (13,072 bytes) 12/1/1999
8:39:36 AM Microsoft Corporation c:\winnt\system32\pjlmon.dll
cnbjmon.dll 5.00.2134.1 43.77 KB (44,816 bytes) 12/1/1999
8:38:48 AM Microsoft Corporation c:\winnt\system32\cnbjmon.dll
localspl.dll 5.00.2195.2793 246.77 KB (252,688 bytes) 12/8/1999
5:00:00 AM Microsoft Corporation c:\winnt\system32\localspl.dll
spoolss.dll 5.00.2161.1 61.77 KB (63,248 bytes) 1/8/2003
2:27:15 AM Microsoft Corporation c:\winnt\system32\spoolss.dll
spoolsv.exe 5.00.2161.1 43.77 KB (44,816 bytes) 1/8/2003
2:27:15 AM Microsoft Corporation c:\winnt\system32\spoolsv.exe
rpcss.dll 5.00.2195.2815 231.27 KB (236,816 bytes) 1/7/2003
5:54:52 PM Microsoft Corporation c:\winnt\system32\rpcss.dll
svchost.exe 5.00.2134.1 7.77 KB (7,952 bytes) 12/8/1999 5:00:00 AM
Microsoft Corporation c:\winnt\system32\svchost.exe
dssenh.dll 5.00.2195.2228 142.77 KB (146,192 bytes) 1/7/2003
5:55:59 PM Microsoft Corporation c:\winnt\system32\dssenh.dll

oakley.dll	5.00.2195.2785	378.77 KB (387,856 bytes)	1/7/2003	5:54:44 PM	Microsoft Corporation c:\winnt\system32\mswsock.dll
5:54:47 PM	Microsoft Corporation	c:\winnt\system32\oakley.dll		msgsvc.dll	5.00.2195.2939 34.27 KB (35,088 bytes) 12/8/1999
mfc42u.dll	6.00.8665.0	972.05 KB (995,384 bytes)	12/8/1999	5:00:00 AM	Microsoft Corporation c:\winnt\system32\msgsvc.dll
5:00:00 AM	Microsoft Corporation	c:\winnt\system32\mfc42u.dll		browser.dll	5.00.2195.2778 48.27 KB (49,424 bytes) 1/7/2003
polagent.dll	5.00.2183.1	108.27 KB (110,864 bytes)	12/8/1999	5:54:27 PM	Microsoft Corporation c:\winnt\system32\browser.dll
5:00:00 AM	Microsoft Corporation	c:\winnt\system32\polagent.dll		alrsvc.dll	5.00.2134.1 17.77 KB (18,192 bytes) 12/8/1999
scecli.dll	5.00.2195.2780	105.27 KB (107,792 bytes)	1/7/2003	5:00:00 AM	Microsoft Corporation c:\winnt\system32\alrsvc.dll
5:54:53 PM	Microsoft Corporation	c:\winnt\system32\scecli.dll		trkwks.dll	5.00.2166.1 88.77 KB (90,896 bytes) 12/8/1999
atl.dll	3.00.8449	57.56 KB (58,938 bytes)	12/8/1999 5:00:00 AM	5:00:00 AM	Microsoft Corporation c:\winnt\system32\trkwks.dll
	Microsoft Corporation	c:\winnt\system32\atl.dll		seclogon.dll	5.00.2135.1 15.77 KB (16,144 bytes) 12/8/1999
certcli.dll	5.00.2195.2778	130.77 KB (133,904 bytes)	1/7/2003	5:00:00 AM	Microsoft Corporation c:\winnt\system32\seclogon.dll
5:54:28 PM	Microsoft Corporation	c:\winnt\system32\certcli.dll		psbase.dll	5.00.2195.2779 111.77 KB (114,448 bytes) 1/7/2003
esent.dll	6.0.3940.13	1.08 MB (1,135,376 bytes)	1/7/2003	5:54:50 PM	Microsoft Corporation c:\winnt\system32\psbase.dll
5:54:33 PM	Microsoft Corporation	c:\winnt\system32\esent.dll		cryptsvc.dll	5.00.2181.1 61.77 KB (63,248 bytes) 12/8/1999
ntdsatq.dll	5.00.2195.2878	31.27 KB (32,016 bytes)	1/7/2003	5:00:00 AM	Microsoft Corporation c:\winnt\system32\cryptsvc.dll
5:54:46 PM	Microsoft Corporation	c:\winnt\system32\ntdsatq.dll		cryptdll.dll	5.00.2135.1 41.27 KB (42,256 bytes) 12/8/1999
ntdsa.dll	5.00.2195.2899	990.77 KB (1,014,544 bytes)	1/7/2003	5:00:00 AM	Microsoft Corporation c:\winnt\system32\cryptdll.dll
5:54:46 PM	Microsoft Corporation	c:\winnt\system32\ntdsa.dll		wkssvc.dll	5.00.2195.2780 95.27 KB (97,552 bytes) 12/8/1999
kdcsvc.dll	5.00.2195.2878	137.77 KB (141,072 bytes)	1/7/2003	5:00:00 AM	Microsoft Corporation c:\winnt\system32\wkssvc.dll
5:54:37 PM	Microsoft Corporation	c:\winnt\system32\kdcsvc.dll		srvsvc.dll	5.00.2195.2904 79.27 KB (81,168 bytes) 12/8/1999
sfmapi.dll	5.00.2134.1	38.77 KB (39,696 bytes)	12/8/1999	5:00:00 AM	Microsoft Corporation c:\winnt\system32\srvsvc.dll
5:00:00 AM	Microsoft Corporation	c:\winnt\system32\sfmapi.dll		cfgmgr32.dll	5.00.2134.1 16.77 KB (17,168 bytes) 12/8/1999
rassfm.dll	5.00.2195.2671	21.27 KB (21,776 bytes)	1/7/2003	5:00:00 AM	Microsoft Corporation c:\winnt\system32\cfgmgr32.dll
5:54:51 PM	Microsoft Corporation	c:\winnt\system32\rassfm.dll		dmserver.dll	2195.2778.297.3 11.77 KB (12,048 bytes) 1/7/2003 5:54:31 PM VERITAS Software Corp.
rsabase.dll	5.00.2195.2228	128.27 KB (131,344 bytes)	5/4/2001		c:\winnt\system32\dmserver.dll
12:05:02 PM	Microsoft Corporation	c:\winnt\system32\rsabase.dll		lmhsvc.dll	5.00.2195.2778 9.77 KB (10,000 bytes) 12/8/1999 5:00:00 AM
schannel.dll	5.00.2195.2922	138.27 KB (141,584 bytes)	5/4/2001		Microsoft Corporation c:\winnt\system32\lmhsvc.dll
12:05:02 PM	Microsoft Corporation	c:\winnt\system32\schannel.dll		dnssrslvr.dll	5.00.2195.2778 88.77 KB (90,896 bytes) 1/7/2003
netlogon.dll	5.00.2195.2865	357.77 KB (366,352 bytes)	1/7/2003	5:54:32 PM	Microsoft Corporation c:\winnt\system32\dnssrslvr.dll
5:54:45 PM	Microsoft Corporation	c:\winnt\system32\netlogon.dll		tapi32.dll	5.00.2182.1 123.27 KB (126,224 bytes) 12/8/1999
msv1_0.dll	5.00.2195.2900	111.77 KB (114,448 bytes)	12/8/1999	5:00:00 AM	Microsoft Corporation c:\winnt\system32\tapi32.dll
5:00:00 AM	Microsoft Corporation	c:\winnt\system32\msv1_0.dll		rasman.dll	5.00.2195.2780 54.77 KB (56,080 bytes) 12/8/1999
kerberos.dll	5.00.2195.2913	198.77 KB (203,536 bytes)	1/7/2003	5:00:00 AM	Microsoft Corporation c:\winnt\system32\rasman.dll
5:54:37 PM	Microsoft Corporation	c:\winnt\system32\kerberos.dll		rasapi32.dll	5.00.2195.2671 189.77 KB (194,320 bytes) 12/8/1999
msprivs.dll	5.00.2154.1	41.50 KB (42,496 bytes)	12/8/1999	5:00:00 AM	Microsoft Corporation c:\winnt\system32\rasapi32.dll
5:00:00 AM	Microsoft Corporation	c:\winnt\system32\msprivs.dll		rtutils.dll	5.00.2168.1 43.77 KB (44,816 bytes) 12/8/1999
samsrv.dll	5.00.2195.2918	369.77 KB (378,640 bytes)	12/8/1999	5:00:00 AM	Microsoft Corporation c:\winnt\system32\rtutils.dll
5:00:00 AM	Microsoft Corporation	c:\winnt\system32\samsrv.dll		adslrpc.dll	5.00.2195.2842 127.27 KB (130,320 bytes) 1/7/2003
lsasrv.dll	5.00.2195.2964	492.77 KB (504,592 bytes)	12/8/1999	5:54:25 PM	Microsoft Corporation c:\winnt\system32\adslrpc.dll
5:00:00 AM	Microsoft Corporation	c:\winnt\system32\lsasrv.dll		activeds.dll	5.00.2195.2778 174.77 KB (178,960 bytes) 1/7/2003
lsass.exe	5.00.2195.2964	32.77 KB (33,552 bytes)	12/8/1999	5:54:20 PM	Microsoft Corporation c:\winnt\system32\activeds.dll
5:00:00 AM	Microsoft Corporation	c:\winnt\system32\lsass.exe		mprapi.dll	5.00.2181.1 79.27 KB (81,168 bytes) 12/8/1999
msi.dll	2.0.2600.2	1.90 MB (1,994,240 bytes)	1/7/2003 5:54:41 PM	5:00:00 AM	Microsoft Corporation c:\winnt\system32\mprapi.dll
	Microsoft Corporation	c:\winnt\system32\msi.dll		iphlpapi.dll	5.00.2173.2 67.77 KB (69,392 bytes) 12/8/1999
appmgmts.dll	5.00.2168.1	117.77 KB (120,592 bytes)	12/8/1999	5:00:00 AM	Microsoft Corporation c:\winnt\system32\iphlpapi.dll
5:00:00 AM	Microsoft Corporation	c:\winnt\system32\appmgmts.dll		icmp.dll	5.00.2134.1 7.27 KB (7,440 bytes) 12/8/1999 5:00:00 AM
wmicore.dll	5.00.2195.2842	72.27 KB (74,000 bytes)	1/7/2003		Microsoft Corporation c:\winnt\system32\icmp.dll
5:54:58 PM	Microsoft Corporation	c:\winnt\system32\wmicore.dll		dhcpcsvc.dll	5.00.2195.2778 88.77 KB (90,896 bytes) 12/8/1999
rasadhlp.dll	5.00.2168.1	7.27 KB (7,440 bytes)	12/8/1999 5:00:00 AM	5:00:00 AM	Microsoft Corporation c:\winnt\system32\dhcpcsvc.dll
	Microsoft Corporation	c:\winnt\system32\rasadhlp.dll		eventlog.dll	5.00.2178.1 43.77 KB (44,816 bytes) 12/8/1999
winnrnr.dll	5.00.2160.1	18.77 KB (19,216 bytes)	12/8/1999	5:00:00 AM	Microsoft Corporation c:\winnt\system32\eventlog.dll
5:00:00 AM	Microsoft Corporation	c:\winnt\system32\winnrnr.dll		ntdsapi.dll	5.00.2195.2661 55.77 KB (57,104 bytes) 1/7/2003
rnr20.dll	5.00.2195.2871	35.77 KB (36,624 bytes)	1/7/2003	5:54:46 PM	Microsoft Corporation c:\winnt\system32\ntdsapi.dll
5:54:51 PM	Microsoft Corporation	c:\winnt\system32\rnr20.dll		scsersv.dll	5.00.2195.2780 226.27 KB (231,696 bytes) 1/7/2003
wshtcpip.dll	5.00.2195.2104	17.27 KB (17,680 bytes)	1/7/2003	5:54:53 PM	Microsoft Corporation c:\winnt\system32\scsersv.dll
5:54:58 PM	Microsoft Corporation	c:\winnt\system32\wshtcpip.dll		umpnpmgr.dll	5.00.2182.1 86.27 KB (88,336 bytes) 12/8/1999
msafd.dll	5.00.2195.2779	106.77 KB (109,328 bytes)	1/7/2003	5:00:00 AM	Microsoft Corporation c:\winnt\system32\umpnpmgr.dll
5:54:39 PM	Microsoft Corporation	c:\winnt\system32\msafd.dll		services.exe	5.00.2195.2780 86.77 KB (88,848 bytes) 12/8/1999
mswsock.dll	5.00.2195.2871	62.77 KB (64,272 bytes)	1/7/2003		

5:00:00 AM	Microsoft Corporation	c:\winnt\system32\services.exe		winmm.dll	5.00.2161.1	184.77 KB (189,200 bytes)	12/8/1999
netmsg.dll	5.00.2137.1	152.50 KB (156,160 bytes)	12/8/1999	5:00:00 AM	Microsoft Corporation	c:\winnt\system32\winmm.dll	
5:00:00 AM	Microsoft Corporation	c:\winnt\system32\netmsg.dll		comctl32.dll	5.81	537.77 KB (550,672 bytes)	12/8/1999 5:00:00 AM
comdlg32.dll	5.00.3103.1000	236.77 KB (242,448 bytes)	12/8/1999		Microsoft Corporation	c:\winnt\system32\comctl32.dll	
5:00:00 AM	Microsoft Corporation	c:\winnt\system32\comdlg32.dll		shlwapi.dll	5.00.3315.1000	282.77 KB (289,552 bytes)	1/7/2003
netui2.dll	5.00.2134.1	280.27 KB (286,992 bytes)	12/8/1999	5:54:54 PM	Microsoft Corporation	c:\winnt\system32\shlwapi.dll	
5:00:00 AM	Microsoft Corporation	c:\winnt\system32\netui2.dll		shell32.dll	5.00.3315.2902	2.25 MB (2,359,056 bytes)	1/7/2003
mprui.dll	5.00.2195.2104	54.77 KB (56,080 bytes)	1/7/2003	5:54:54 PM	Microsoft Corporation	c:\winnt\system32\shell32.dll	
5:54:39 PM	Microsoft Corporation	c:\winnt\system32\mprui.dll		msgina.dll	5.00.2195.2779	324.27 KB (332,048 bytes)	12/8/1999
netui1.dll	5.00.2134.1	210.27 KB (215,312 bytes)	12/8/1999	5:00:00 AM	Microsoft Corporation	c:\winnt\system32\msgina.dll	
5:00:00 AM	Microsoft Corporation	c:\winnt\system32\netui1.dll		wsock32.dll	5.00.2195.2871	21.27 KB (21,776 bytes)	1/7/2003
netui0.dll	5.00.2134.1	70.27 KB (71,952 bytes)	12/8/1999	5:54:58 PM	Microsoft Corporation	c:\winnt\system32\wsock32.dll	
5:00:00 AM	Microsoft Corporation	c:\winnt\system32\netui0.dll		dnsapi.dll	5.00.2195.2785	130.77 KB (133,904 bytes)	1/7/2003
ntlanman.dll	5.00.2157.1	35.27 KB (36,112 bytes)	12/8/1999	5:54:31 PM	Microsoft Corporation	c:\winnt\system32\dnsapi.dll	
5:00:00 AM	Microsoft Corporation	c:\winnt\system32\ntlanman.dll		wldap32.dll	5.00.2195.2797	125.27 KB (128,272 bytes)	1/7/2003
mpr.dll	5.00.2195.2779	53.27 KB (54,544 bytes)	1/7/2003 5:54:38 PM	5:54:58 PM	Microsoft Corporation	c:\winnt\system32\wldap32.dll	
	Microsoft Corporation	c:\winnt\system32\mpr.dll		ws2help.dll	5.00.2134.1	17.77 KB (18,192 bytes)	12/8/1999
cscui.dll	5.00.2195.2959	228.27 KB (233,744 bytes)	1/7/2003	5:00:00 AM	Microsoft Corporation	c:\winnt\system32\ws2help.dll	
5:54:29 PM	Microsoft Corporation	c:\winnt\system32\cscui.dll		ws2_32.dll	5.00.2195.2780	67.77 KB (69,392 bytes)	1/7/2003
winsta.dll	5.00.2195.2386	36.77 KB (37,648 bytes)	1/7/2003	5:54:58 PM	Microsoft Corporation	c:\winnt\system32\ws2_32.dll	
5:54:58 PM	Microsoft Corporation	c:\winnt\system32\winsta.dll		samlib.dll	5.00.2195.2780	49.77 KB (50,960 bytes)	12/8/1999
sclogntfy.dll	5.00.2191.1	20.27 KB (20,752 bytes)	12/8/1999	5:00:00 AM	Microsoft Corporation	c:\winnt\system32\samlib.dll	
5:00:00 AM	Microsoft Corporation	c:\winnt\system32\sclogntfy.dll		netrap.dll	5.00.2134.1	11.27 KB (11,536 bytes)	12/8/1999
cryptnet.dll	5.131.2157.1	41.77 KB (42,768 bytes)	12/8/1999	5:00:00 AM	Microsoft Corporation	c:\winnt\system32\netrap.dll	
5:00:00 AM	Microsoft Corporation	c:\winnt\system32\cryptnet.dll		netapi32.dll	5.00.2195.2808	303.77 KB (311,056 bytes)	1/7/2003
clbcatq.dll	2000.2.3471.1	496.77 KB (508,688 bytes)	1/7/2003	5:54:45 PM	Microsoft Corporation	c:\winnt\system32\netapi32.dll	
5:54:28 PM	Microsoft Corporation	c:\winnt\system32\clbcatq.dll		profmap.dll	5.00.2181.1	29.27 KB (29,968 bytes)	12/8/1999
oleaut32.dll	2.40.4517	612.27 KB (626,960 bytes)	12/8/1999	5:00:00 AM	Microsoft Corporation	c:\winnt\system32\profmap.dll	
5:00:00 AM	Microsoft Corporation	c:\winnt\system32\oleaut32.dll		secur32.dll	5.00.2195.2862	46.77 KB (47,888 bytes)	1/7/2003
winspool.drv	5.00.2195.2780	109.77 KB (112,400 bytes)	12/8/1999	5:54:53 PM	Microsoft Corporation	c:\winnt\system32\secur32.dll	
5:00:00 AM	Microsoft Corporation	c:\winnt\system32\winspool.drv		sfc.dll	5.00.2195.2896	92.11 KB (94,320 bytes)	1/7/2003 5:54:53 PM
winscard.dll	5.00.2134.1	77.27 KB (79,120 bytes)	12/8/1999		Microsoft Corporation	c:\winnt\system32\sfc.dll	
5:00:00 AM	Microsoft Corporation	c:\winnt\system32\winscard.dll		nddeapi.dll	5.00.2137.1	15.27 KB (15,632 bytes)	12/8/1999
wlnotify.dll	5.00.2195.2780	53.77 KB (55,056 bytes)	1/7/2003	5:00:00 AM	Microsoft Corporation	c:\winnt\system32\nddeapi.dll	
5:54:58 PM	Microsoft Corporation	c:\winnt\system32\wlnotify.dll		userenv.dll	5.00.2195.2780	361.77 KB (370,448 bytes)	12/8/1999
cscdll.dll	5.00.2195.2401	98.27 KB (100,624 bytes)	1/7/2003	5:00:00 AM	Microsoft Corporation	c:\winnt\system32\userenv.dll	
5:54:29 PM	Microsoft Corporation	c:\winnt\system32\cscdll.dll		user32.dll	5.00.2195.2821	392.77 KB (402,192 bytes)	12/8/1999
lz32.dll	5.00.2134.1	9.77 KB (10,000 bytes)	12/8/1999 5:00:00 AM	5:00:00 AM	Microsoft Corporation	c:\winnt\system32\user32.dll	
	Microsoft Corporation	c:\winnt\system32\lz32.dll		gdi32.dll	5.00.2195.2778	228.77 KB (234,256 bytes)	12/8/1999
version.dll	5.00.2134.1	15.77 KB (16,144 bytes)	12/8/1999	5:00:00 AM	Microsoft Corporation	c:\winnt\system32\gdi32.dll	
5:00:00 AM	Microsoft Corporation	c:\winnt\system32\version.dll		rpcrt4.dll	5.00.2195.2832	437.27 KB (447,760 bytes)	1/7/2003
rsaenh.dll	5.00.2195.2228	130.77 KB (133,904 bytes)	1/7/2003	5:54:52 PM	Microsoft Corporation	c:\winnt\system32\rpcrt4.dll	
5:55:59 PM	Microsoft Corporation	c:\winnt\system32\rsaenh.dll		advapi32.dll	5.00.2195.2867	351.77 KB (360,208 bytes)	12/8/1999
mscat32.dll	5.131.2134.1	7.77 KB (7,952 bytes)	12/8/1999 5:00:00 AM	5:00:00 AM	Microsoft Corporation	c:\winnt\system32\advapi32.dll	
	Microsoft Corporation	c:\winnt\system32\mscat32.dll		kernel32.dll	5.00.2195.2778	714.77 KB (731,920 bytes)	12/8/1999
ole32.dll	5.00.2195.2887	969.77 KB (993,040 bytes)	1/7/2003	5:00:00 AM	Microsoft Corporation	c:\winnt\system32\kernel32.dll	
5:54:49 PM	Microsoft Corporation	c:\winnt\system32\ole32.dll		msvcrt.dll	6.10.8924.0	284.05 KB (290,869 bytes)	5/4/2001
imagehlp.dll	5.00.2195.2778	125.77 KB (128,784 bytes)	5/4/2001	12:05:02 PM	Microsoft Corporation	c:\winnt\system32\msvcrt.dll	
12:05:02 PM	Microsoft Corporation	c:\winnt\system32\imagehlp.dll		winlogon.exe	5.00.2195.2953	173.77 KB (177,936 bytes)	12/8/1999
msasn1.dll	5.00.2134.1	51.27 KB (52,496 bytes)	12/8/1999	5:00:00 AM	Microsoft Corporation	c:\winnt\system32\winlogon.exe	
5:00:00 AM	Microsoft Corporation	c:\winnt\system32\msasn1.dll		sfcfiles.dll	5.00.2195.2967	948.27 KB (971,024 bytes)	1/7/2003
crypt32.dll	5.131.2195.2833	451.27 KB (462,096 bytes)		5:54:53 PM	Microsoft Corporation	c:\winnt\system32\sfcfiles.dll	
	1/7/2003 5:54:29 PM	Microsoft Corporation		ntdll.dll	5.00.2195.2779	478.77 KB (490,256 bytes)	5/4/2001
	c:\winnt\system32\crypt32.dll			12:05:02 PM	Microsoft Corporation	c:\winnt\system32\ntdll.dll	
wintrust.dll	5.131.2195.2779	162.27 KB (166,160 bytes)		5:00:00 AM	Microsoft Corporation	c:\winnt\system32\smss.exe	
	1/7/2003 5:54:58 PM	Microsoft Corporation					
	c:\winnt\system32\wintrust.dll						
setupapi.dll	5.00.2195.2663	555.77 KB (569,104 bytes)	12/8/1999				
5:00:00 AM	Microsoft Corporation	c:\winnt\system32\setupapi.dll					

[Services]


```

c:\winnt\system32\svchost.exe -k netsvcs      Normal LocalSystem
0
Internet Connection Sharing  SharedAccess  Stopped Manual Share
Process c:\winnt\system32\svchost.exe -k netsvcs      Normal LocalSystem
0
Print Spooler  SpoolerRunning Auto  Own Process
c:\winnt\system32\spoolsv.exe Normal LocalSystem 0
Performance Logs and Alerts  SysmonLog  Stopped Manual Own Process
c:\winnt\system32\smlogsvc.exe      Normal LocalSystem 0
Telephony  TapiSrv Running Manual Share Process
c:\winnt\system32\svchost.exe -k tapisrv      Normal LocalSystem
0
Terminal Services  TermService  Stopped Disabled  Own Process
c:\winnt\system32\termsrv.exe Normal LocalSystem 0
Telnet  TlntSvr Stopped Manual Own Process
c:\winnt\system32\tlntsvr.exe Normal LocalSystem 0
Distributed Link Tracking Server  TrkSvr  Stopped Manual Share
Process c:\winnt\system32\services.exe Normal LocalSystem 0
Distributed Link Tracking Client  TrkWks  Running Auto  Share
Process c:\winnt\system32\services.exe Normal LocalSystem 0
Uninterruptible Power Supply  UPS  Stopped Manual Own Process
c:\winnt\system32\ups.exe Normal LocalSystem 0
Utility Manager  UtilMan Stopped Manual Own Process
c:\winnt\system32\utilman.exe Normal LocalSystem 0
Visual Studio Analyzer RPC bridge  Visual Studio Analyzer RPC bridge
Stopped Manual Own Process c:\program files\microsoft visual
studio\common\tools\vs-ent98\vanalyzr\varpc.exe      Normal LocalSystem
0
Windows Time  W32Time Stopped Manual Share Process
c:\winnt\system32\services.exe Normal LocalSystem 0
World Wide Web Publishing Service  W3SVC  Running Auto  Share
Process c:\winnt\system32\inetssrv\inetinfo.exe      Normal LocalSystem
0
Windows Management Instrumentation  WinMgmt Running Auto  Own Process
c:\winnt\system32\wbem\winmgmt.exe Ignore LocalSystem 0
Windows Management Instrumentation Driver Extensions Wmi  Running
Manual Share Process c:\winnt\system32\services.exe
Normal LocalSystem 0

```

[Program Groups]

Group Name	Name	User Name	Default User
Accessories	Default User:Accessories	Default User	
Accessories\Accessibility	Default User:Accessories\Accessibility	Default User	
Accessories\Entertainment	Default User:Accessories\Entertainment	Default User	
Accessories\System Tools	Default User:Accessories\System Tools	Default User	
Microsoft Web Publishing	Default User:Microsoft Web Publishing	Default User	
Startup	Default User:Startup	Default User	
Accessories	All Users:Accessories	All Users	
Accessories\Accessibility	All Users:Accessories\Accessibility	All Users	
Accessories\Communications	All Users:Accessories\Communications	All Users	
Accessories\Entertainment	All Users:Accessories\Entertainment	All Users	

```

Accessories\Games      All Users:Accessories\Games  All Users
Accessories\Microsoft Script Debugger All Users:Accessories\Microsoft
Script Debugger      All Users
Accessories\System Tools      All Users:Accessories\System Tools  All
Users
ActiveState ActivePerl 5.8      All Users:ActiveState ActivePerl 5.8 All
Users
Administrative Tools  All Users:Administrative Tools      All Users
Cygwin  All Users:Cygwin      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 Studio 6.0  All Users:Microsoft Visual Studio 6.0 All
Users
Microsoft Visual Studio 6.0\Microsoft Visual SourceSafe  All
Users:Microsoft Visual Studio 6.0\Microsoft Visual SourceSafe  All
Users
Microsoft Visual Studio 6.0\Microsoft Visual Studio 6.0 Enterprise Tools
All Users:Microsoft Visual Studio 6.0\Microsoft Visual Studio 6.0
Enterprise Tools      All Users
Microsoft Visual Studio 6.0\Microsoft Visual Studio 6.0 Tools      All
Users:Microsoft Visual Studio 6.0\Microsoft Visual Studio 6.0 Tools All
Users
Startup All Users:Startup      All Users
Accessories  ACL37\nec:Accessories  ACL37\nec
Accessories\Accessibility  ACL37\nec:Accessories\Accessibility
ACL37\nec
Accessories\Entertainment  ACL37\nec:Accessories\Entertainment
ACL37\nec
Accessories\System Tools  ACL37\nec:Accessories\System Tools
ACL37\nec
Microsoft Web Publishing  ACL37\nec:Microsoft Web Publishing
ACL37\nec
Startup ACL37\nec:Startup  ACL37\nec
Accessories  ACL37\Administrator:Accessories  ACL37\Administrator
Accessories\Accessibility  ACL37\Administrator:Accessories\Accessibility
ACL37\Administrator
Accessories\Entertainment  ACL37\Administrator:Accessories\Entertainment
ACL37\Administrator
Accessories\System Tools  ACL37\Administrator:Accessories\System
Tools  ACL37\Administrator
Startup ACL37\Administrator:Startup  ACL37\Administrator

```

[Startup Programs]

Program	Command	User Name	Location
No startup program information			

[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
 DDSContainerCtl Class Not Available
 Bitmap Image mspaint.exe

[Internet Explorer 5]

[Following are sub-categories of this main category]

[Summary]

Item Value
 Version 5.00.3315.1000
 Build 53315.1000
 Product ID 51876-335-9608525-05926
 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	5.0.2195.2867	352 KB	5/4/2001 12:05:02 PM	C:\WINNT\system32	Microsoft Corporation
advpack.dll	5.0.3103.1000	87 KB	5/4/2001 12:05:02 PM	C:\WINNT\system32	Microsoft Corporation
browseic.dll	5.0.3315.2846	35 KB	5/4/2001 12:05:02 PM	C:\WINNT\system32	Microsoft Corporation
browseui.dll	5.0.3315.2846	789 KB	5/4/2001 12:05:02 PM	C:\WINNT\system32	Microsoft Corporation
ckcnv.exe	5.0.2189.1	9 KB	12/8/1999 5:00:00 AM	C:\WINNT\system32	Microsoft Corporation
comctl32.dll	5.81.3103.1000	538 KB	5/4/2001 12:05:02 PM	C:\WINNT\system32	Microsoft Corporation
crypt32.dll	5.131.2195.2833	451 KB	5/4/2001 12:05:02 PM	C:\WINNT\system32	Microsoft Corporation
enhshg.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
iemigrat.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
iesetup.dll	5.0.3103.1000	57 KB	5/4/2001 12:05:02 PM	C:\WINNT\system32	Microsoft Corporation
ieexplore.exe	5.0.2920.0	59 KB	12/8/1999 5:00:00 AM	C:\Program Files\Internet Explorer	Microsoft Corporation

imagehlp.dll	5.0.2195.2778	126 KB	5/4/2001 12:05:02 PM	C:\WINNT\system32	Microsoft Corporation
imghelp.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
inseng.dll	5.0.3103.1000	72 KB	5/4/2001 12:05:02 PM	C:\WINNT\system32	Microsoft Corporation
jobexec.dll	5.0.0.147 KB	12/8/1999 5:00:00 AM	C:\WINNT\system32	Microsoft Corporation	
jscrip.dll	5.1.0.5907	476 KB	5/4/2001 12:05:02 PM	C:\WINNT\system32	Microsoft Corporation
jsproxy.dll	5.0.2920.0	13 KB	12/8/1999 5:00:00 AM	C:\WINNT\system32	Microsoft Corporation
msahtml.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
mshtml.dll	5.0.3315.2870	2290 KB	5/4/2001 12:05:02 PM	C:\WINNT\system32	Microsoft Corporation
msjava.dll	5.0.3802.0	923 KB	5/4/2001 12:05:02 PM	C:\WINNT\system32	Microsoft Corporation
msoss.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
msxml.dll	8.0.5718.1	493 KB	5/4/2001 12:05:02 PM	C:\WINNT\system32	Microsoft Corporation
occache.dll	5.0.3103.1000	86 KB	5/4/2001 12:05:02 PM	C:\WINNT\system32	Microsoft Corporation
ole32.dll	5.0.2195.2887	970 KB	5/4/2001 12:05:02 PM	C:\WINNT\system32	Microsoft Corporation
oleaut32.dll	2.40.4517.0	612 KB	5/4/2001 12:05:02 PM	C:\WINNT\system32	Microsoft Corporation
olepro32.dll	5.0.4517.0	160 KB	5/4/2001 12:05:02 PM	C:\WINNT\system32	Microsoft Corporation
rsabase.dll	5.0.2195.2228	128 KB	5/4/2001 12:05:02 PM	C:\WINNT\system32	Microsoft Corporation
rsaenh.dll	5.0.2195.2228	131 KB	5/4/2001 12:05:02 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	5.1.2195.0	138 KB	5/4/2001 12:05:02 PM	C:\WINNT\system32	Microsoft Corporation
shdoc401.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
shdocvw.dll	5.0.3315.2879	1078 KB	5/4/2001 12:05:02 PM	C:\WINNT\system32	Microsoft Corporation
shell32.dll	5.0.3315.2902	2304 KB	5/4/2001 12:05:02 PM	C:\WINNT\system32	Microsoft Corporation
shlwapi.dll	5.0.3315.1000	283 KB	5/4/2001 12:05:02 PM	C:\WINNT\system32	Microsoft Corporation
url.dll	5.0.2920.0	82 KB	12/8/1999 5:00:00 AM	C:\WINNT\system32	Microsoft Corporation
urlmon.dll	5.0.3315.1000	441 KB	5/4/2001 12:05:02 PM	C:\WINNT\system32	Microsoft Corporation
vbscript.dll	5.1.0.5907	428 KB	5/4/2001 12:05:02 PM	C:\WINNT\system32	Microsoft Corporation
webcheck.dll	5.0.3315.1000	252 KB	5/4/2001 12:05:02 PM	C:\WINNT\system32	Microsoft Corporation
win.com	5.0.2134.1	24 KB	12/8/1999 5:00:00 AM	C:\WINNT\system32	Microsoft Corporation
wininet.dll	5.0.3315.1000	457 KB	5/4/2001 12:05:02 PM		

```

C:\WINNT\system32      Microsoft Corporation
winsock.dll    3.10.0.103    3 KB    12/8/1999 5:00:00 AM
C:\WINNT\system32      Microsoft Corporation
wintrust.dll    5.131.2195.2779    162 KB    5/4/2001 12:05:02 PM
C:\WINNT\system32      Microsoft Corporation
wsock.vxd      <File Missing> Not Available Not Available Not
Available      Not Available
wsock32.dll    5.0.2195.2871    21 KB    5/4/2001 12:05:02 PM
C:\WINNT\system32      Microsoft Corporation
wsock32n.dll   <File Missing> Not Available Not Available Not
Available      Not Available

```

[Connectivity]

```

Item    Value
Connection Preference Never dial
EnableHttp1.1 1
ProxyHttp1.1 0

```

LAN Settings

```

AutoConfigProxy      wininet.dll
AutoProxyDetectMode  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\nec\Local
Settings\Temporary Internet Files
Total Disk Space      17335 MB
Available Disk Space  10350 MB
Maximum Cache Size    541 MB
Available Cache Size   542 MB

```

[List of Objects]

```

Program File    Status CodeBase
No cached object information available

```

[Content]

[Following are sub-categories of this main category]

[Summary]

```

Item    Value
Content Advisor      Disabled

```

[Personal Certificates]

```

Issued To    Issued By    Validity    Signature Algorithm
No personal certificate information available

```

[Other People Certificates]

```

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

```

[Publishers]

```

Name
No publisher information available

```

[Security]

```

Zone    Security Level
Local intranet Medium-low
Trusted sites Low
Internet Medium
Restricted sites High

```

RTE input parameter

The following parameters were used with NEC proprietary RTE.

RTE master driver

```

-----
Help message
-----

```

Parameter	Default

General test parameters	
-u Ramp Up Time (sec)	600
-s Steady State Time (sec)	1200
-d Ramp Down Time (sec)	120
-c Number of Connections	10
-w Number of Warehouses	1
-n C value for NURand	66
Reporting/Logging options	
-R Name of Report File (txt format)	report.txt
-X Name of Report File (xlt format)	report.xlt
-P Name of NewOrder Troughput Curve File	thrput.xlt
-l Enable Statistics Logging.	
-L Name of Statistics Log File	slog
-o Generate 'success' File for Durability Test	
-O Name of 'success' File	succ
Misc	
-C Connect Rate of a Client Driver (users/min)	1000
-S Start Rate of a Client Driver (users/min)	1000

-v Verbose Mode (experimental)

Tunable parameters

-W Comma Separated List of Distribution Weights

(NewOrder)	4480
(Payment)	4308
(OrderStatus)	404
(Delivery)	404
(StockLevel)	404
-T Comma Separated List of Think Times (sec)	
(NewOrder)	12.05
(OrderStatus)	12.05 (Payment)
(Delivery)	10.05
(StockLevel)	5.05
-K Comma Separated List of Keying Times (sec)	
(NewOrder)	18.01
(OrderStatus)	3.01 (Payment)
(Delivery)	2.01
(StockLevel)	2.01
-D Web Browser Delay Time (sec)	0.10

Startup parameters

master -u5400 -s9000 -d300 -c198000 -w19800 -C1500 -S50 -n66 -
T12.07,12.07,10.07,5.07,5.07

Though we had started test with 1.5h ramp-up and 2.5h measurement interval, Followings are the master.exe console output messages that show the procedure of M.I. adjustment.

```
[Tue Dec 21 16:10:51] Ramp up period: 0/5400 secs
[Tue Dec 21 16:10:52] Ramp up period: 1/5400 secs
[Tue Dec 21 16:10:53] Ramp up period: 2/5400 secs
[Tue Dec 21 16:10:54] Ramp up period: 3/5400 secs
[Tue Dec 21 16:10:55] Ramp up period: 4/5400 secs
[Tue Dec 21 16:10:56] Ramp up period: 5/5400 secs
:
[Tue Dec 21 17:40:46] Ramp up period: 5395/5400 secs
[Tue Dec 21 17:40:47] Ramp up period: 5396/5400 secs
```

```
[Tue Dec 21 17:40:48] Ramp up period: 5397/5400 secs
[Tue Dec 21 17:40:49] Ramp up period: 5398/5400 secs
[Tue Dec 21 17:40:50] Ramp up period: 5399/5400 secs
[Tue Dec 21 17:40:51] Start steady state (CTRL-C to stop M.I. immediately.)
[Tue Dec 21 17:40:51] Steady state period: 0/9000 secs
[Tue Dec 21 17:40:52] Steady state period: 1/9000 secs
[Tue Dec 21 17:40:53] Steady state period: 2/9000 secs
[Tue Dec 21 17:40:54] Steady state period: 3/9000 secs
[Tue Dec 21 17:40:55] Steady state period: 4/9000 secs
[Tue Dec 21 17:40:56] Steady state period: 5/9000 secs
:
[Tue Dec 21 20:10:46] Steady state period: 8995/9000 secs
[Tue Dec 21 20:10:47] Steady state period: 8996/9000 secs
[Tue Dec 21 20:10:48] Steady state period: 8997/9000 secs
[Tue Dec 21 20:10:49] Steady state period: 8998/9000 secs
[Tue Dec 21 20:10:50] Steady state period: 8999/9000 secs
[Tue Dec 21 20:10:51] Start ramp down (CTRL-C to stop test immediately.)
[Tue Dec 21 20:10:51] Ramp down period: 0/300 secs
[Tue Dec 21 20:10:52] Ramp down period: 1/300 secs
[Tue Dec 21 20:10:53] Ramp down period: 2/300 secs
[Tue Dec 21 20:10:54] Ramp down period: 3/300 secs
[Tue Dec 21 20:10:55] Ramp down period: 4/300 secs
[Tue Dec 21 20:10:56] Ramp down period: 5/300 secs
:
[Tue Dec 21 20:15:46] Ramp down period: 295/300 secs
[Tue Dec 21 20:15:47] Ramp down period: 296/300 secs
[Tue Dec 21 20:15:48] Ramp down period: 297/300 secs
[Tue Dec 21 20:15:49] Ramp down period: 298/300 secs
[Tue Dec 21 20:15:50] Ramp down period: 299/300 secs
[Tue Dec 21 20:15:51] Waiting for 240 client driver(s) to finish.
[Tue Dec 21 20:15:54] Waiting for 203 client driver(s) to finish.
[Tue Dec 21 20:15:57] Waiting for 166 client driver(s) to finish.
[Tue Dec 21 20:16:00] Waiting for 130 client driver(s) to finish.
[Tue Dec 21 20:16:03] Waiting for 94 client driver(s) to finish.
[Tue Dec 21 20:16:06] Waiting for 57 client driver(s) to finish.
[Tue Dec 21 20:16:09] Waiting for 19 client driver(s) to finish.
[Tue Dec 21 20:16:12] Test complete. See report.txt for details.
```

RTE slave driver

Help message

Parameter	Default
-M Master Driver Servername	.
-I HTTP Servername	.
-p HTTP Server Port	80
-c Number of Connections	10
-o Starting User ID	1
-d Disable HTTP Keep-Alive	

Startup parameters

```

if %COMPUTERNAME%==RTE33 (
start cmd /k c:\client7 -Mbmater -Iacl33 -o1 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl332 -o826 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl333 -o1651 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl33 -o2476 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl332 -o3301 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl333 -o4126 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl33 -o4951 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl332 -o5776 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl333 -o6601 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl33 -o7426 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl332 -o8251 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl333 -o9076 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl33 -o9901 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl332 -o10726 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl333 -o11551 -c825
sleep 90
)
if %COMPUTERNAME%==RTE34 (
start cmd /k c:\client7 -Mbmater -Iacl34 -o12376 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl342 -o13201 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl343 -o14026 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl34 -o14851 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl342 -o15676 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl343 -o16501 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl34 -o17326 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl342 -o18151 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl343 -o18976 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl34 -o19801 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl342 -o20626 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl343 -o21451 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl34 -o22276 -c825
sleep 90
)

```

```

start cmd /k c:\client7 -Mbmater -Iacl342 -o23101 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl343 -o23926 -c825
sleep 90
)
if %COMPUTERNAME%==RTE35 (
start cmd /k c:\client7 -Mbmater -Iacl35 -o24751 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl352 -o25576 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl353 -o26401 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl35 -o27226 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl352 -o28051 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl353 -o28876 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl35 -o29701 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl352 -o30526 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl353 -o31351 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl35 -o32176 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl352 -o33001 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl353 -o33826 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl35 -o34651 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl352 -o35476 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl353 -o36301 -c825
sleep 90
)
if %COMPUTERNAME%==RTE36 (
start cmd /k c:\client7 -Mbmater -Iacl36 -o37126 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl362 -o37951 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl363 -o38776 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl36 -o39601 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl362 -o40426 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl363 -o41251 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl36 -o42076 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl362 -o42901 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl363 -o43726 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl36 -o44551 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl362 -o45376 -c825

```



```

sleep 90
start cmd /k c:\client7 -Mbmater -Iacl363 -o46201 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl36 -o47026 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl362 -o47851 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl363 -o48676 -c825
sleep 90
)
if %COMPUTERNAME%==RTE37 (
start cmd /k c:\client7 -Mbmater -Iacl37 -o49501 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl372 -o50326 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl373 -o51151 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl37 -o51976 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl372 -o52801 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl373 -o53626 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl37 -o54451 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl372 -o55276 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl373 -o56101 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl37 -o56926 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl372 -o57751 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl373 -o58576 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl37 -o59401 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl372 -o60226 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl373 -o61051 -c825
sleep 90
)
if %COMPUTERNAME%==RTE38 (
start cmd /k c:\client7 -Mbmater -Iacl38 -o61876 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl382 -o62701 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl383 -o63526 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl38 -o64351 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl382 -o65176 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl383 -o66001 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl38 -o66826 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl382 -o67651 -c825
sleep 90
)

```

```

start cmd /k c:\client7 -Mbmater -Iacl383 -o68476 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl38 -o69301 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl382 -o70126 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl383 -o70951 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl38 -o71776 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl382 -o72601 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl383 -o73426 -c825
sleep 90
)
if %COMPUTERNAME%==RTE39 (
start cmd /k c:\client7 -Mbmater -Iacl39 -o74251 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl392 -o75076 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl393 -o75901 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl39 -o76726 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl392 -o77551 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl393 -o78376 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl39 -o79201 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl392 -o80026 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl393 -o80851 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl39 -o81676 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl392 -o82501 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl393 -o83326 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl39 -o84151 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl392 -o84976 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl393 -o85801 -c825
sleep 90
)
if %COMPUTERNAME%==RTE40 (
start cmd /k c:\client7 -Mbmater -Iacl40 -o86626 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl402 -o87451 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl403 -o88276 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl40 -o89101 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl402 -o89926 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl403 -o90751 -c825

```

```

sleep 90
start cmd /k c:\client7 -Mbmater -Iacl40 -o91576 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl402 -o92401 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl403 -o93226 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl40 -o94051 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl402 -o94876 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl403 -o95701 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl40 -o96526 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl402 -o97351 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl403 -o98176 -c825
sleep 90
)
if %COMPUTERNAME%==RTE41 (
start cmd /k c:\client7 -Mbmater -Iacl41 -o99001 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl412 -o99826 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl413 -o100651 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl41 -o101476 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl412 -o102301 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl413 -o103126 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl41 -o103951 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl412 -o104776 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl413 -o105601 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl41 -o106426 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl412 -o107251 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl413 -o108076 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl41 -o108901 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl412 -o109726 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl413 -o110551 -c825
sleep 90
)
if %COMPUTERNAME%==RTE42 (
start cmd /k c:\client7 -Mbmater -Iacl42 -o111376 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl422 -o112201 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl423 -o113026 -c825
sleep 90
)

```

```

start cmd /k c:\client7 -Mbmater -Iacl42 -o113851 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl422 -o114676 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl423 -o115501 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl42 -o116326 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl422 -o117151 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl423 -o117976 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl42 -o118801 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl422 -o119626 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl423 -o120451 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl42 -o121276 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl422 -o122101 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl423 -o122926 -c825
sleep 90
)
if %COMPUTERNAME%==RTE43 (
start cmd /k c:\client7 -Mbmater -Iacl43 -o123751 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl432 -o124576 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl433 -o125401 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl43 -o126226 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl432 -o127051 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl433 -o127876 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl43 -o128701 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl432 -o129526 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl433 -o130351 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl43 -o131176 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl432 -o132001 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl433 -o132826 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl43 -o133651 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl432 -o134476 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl433 -o135301 -c825
sleep 90
)
if %COMPUTERNAME%==RTE44 (
start cmd /k c:\client7 -Mbmater -Iacl44 -o136126 -c825

```

```

sleep 90
start cmd /k c:\client7 -Mbmater -Iacl442 -o136951 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl443 -o137776 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl44 -o138601 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl442 -o139426 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl443 -o140251 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl44 -o141076 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl442 -o141901 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl443 -o142726 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl44 -o143551 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl442 -o144376 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl443 -o145201 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl44 -o146026 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl442 -o146851 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl443 -o147676 -c825
sleep 90
)
if %COMPUTERNAME%==RTE45 (
start cmd /k c:\client7 -Mbmater -Iacl45 -o148501 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl452 -o149326 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl453 -o150151 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl45 -o150976 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl452 -o151801 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl453 -o152626 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl45 -o153451 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl452 -o154276 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl453 -o155101 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl45 -o155926 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl452 -o156751 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl453 -o157576 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl45 -o158401 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl452 -o159226 -c825
sleep 90

```

```

start cmd /k c:\client7 -Mbmater -Iacl453 -o160051 -c825
sleep 90
)
if %COMPUTERNAME%==RTE46 (
start cmd /k c:\client7 -Mbmater -Iacl46 -o160876 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl462 -o161701 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl463 -o162526 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl46 -o163351 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl462 -o164176 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl463 -o165001 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl46 -o165826 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl462 -o166651 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl463 -o167476 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl46 -o168301 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl462 -o169126 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl463 -o169951 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl46 -o170776 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl462 -o171601 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl463 -o172426 -c825
sleep 90
)
if %COMPUTERNAME%==RTE47 (
start cmd /k c:\client7 -Mbmater -Iacl47 -o173251 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl472 -o174076 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl473 -o174901 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl47 -o175726 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl472 -o176551 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl473 -o177376 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl47 -o178201 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl472 -o179026 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl473 -o179851 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl47 -o180676 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl472 -o181501 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl473 -o182326 -c825

```

```

sleep 90
start cmd /k c:\client7 -Mbmater -Iacl47 -o183151 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl472 -o183976 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl473 -o184801 -c825
sleep 90
)
if %COMPUTERNAME%==RTE49 (
start cmd /k c:\client7 -Mbmater -Iacl49 -o185626 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl492 -o186451 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl493 -o187276 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl49 -o188101 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl492 -o188926 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl493 -o189751 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl49 -o190576 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl492 -o191401 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl493 -o192226 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl49 -o193051 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl492 -o193876 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl493 -o194701 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl49 -o195526 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl492 -o196351 -c825
sleep 90
start cmd /k c:\client7 -Mbmater -Iacl493 -o197176 -c825
sleep 90
)

```

Appendix D : Space Calculation

60 Day Space

60-Day Space Computation

All data sizes in MB unless otherwise stated

Warehouses	19,800
Measured TpmC	247,650

Table	Rows	Table	Index	5% Space	Total Space
Warehouse	19,800		60	0	3
District	198,000		72	0	4
Item	100,000		14	0	1
Stock	1,980,000,000	644,580		0	32,229
Customer	594,000,000	464,112	28,680	24,640	517,432
New-Order	178,200,000	6,624		0	6,624
Orders	594,000,000	24,467	16,704	0	41,171
Order-Line	8,910,000,000	654,478		0	654,478
History	594,000,000	36,768		0	36,768

Free Space	307,534			
Dynamic Space	715,714	Log Written (KB)		139,871,712
Static Space	1,217,722	Total New-Order Txns		56,747,418
Daily Growth	143,230	Log Written per New-Order (KB)		2.46
Daily Spread	92,690			

Data Storage Requirement

60 Days (MB)	15,372,876
60 Days (GB)	15,013

Log Storage Requirement

8 Hours (GB)	279.42
--------------	--------

Disk Sizing

Disk Type	Formatted Capacity (GB)	# of Disks	SUT Capacity (GB)	# of Disks	Priced Capacity (GB)
DATABASE	33.20	540	17,928	540	17,928
LOG RAID5(14+P)	33.20	14	465	14	465
OS SCSI	33.20	1	33	1	33

Total Capacity	18,426
----------------	--------

Appendix E : Price Quotation

January 24, 2005

Masami Shiraishi
NEC Corporation
1-10 Nisshin-cho, Fuchu
Tokyo 183-8501 Japan

Dear Shiraishi-san,

The table shown below lists the U.S. pricing for the DB2 Universal Database Enterprise Server Edition product that has been used in TPC-C Benchmark test.

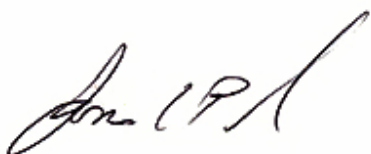
All prices shown are in U.S. Dollars.

Description	Price
DB2 UDB ESE (D518GLL) processor lic+sw maint for 12 months	\$22,260 per CPU
DB2 SW Mtce Renewal one year (E00BILL)	\$1,060 per CPU

This quote is valid for 90 days.

If I can be of any further assistance, please contact me at 914-766-1576 or jonprial@us.ibm.com.

Yours Truly,



Jonathan Prial
Vice President, Marketing, Information Management
Route 100
Somers, NY 10589

Microsoft Corporation
One Microsoft Way
Redmond, WA 98052-6399

Tel 425 882 8080
Fax 425 936 7329
<http://www.microsoft.com/>

Microsoft

January 23, 2005

NEC Corporation
Yoshifumi Saruta
1-10 Nisshin-cho
Fuchu-shi
Tokyo, 183-8501

Saruta-san:

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
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	17	\$12,546
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: PCyosa0523011156.

Please include this Reference ID in any correspondence regarding this price quote.

Your Shopping Cart

Here are the contents of your **PROVANTAGE** SmartCart™.

▼ **Page Options**

[Restore All Menus](#)

▼ **Shopping Cart Contents**

			Recalculate	Check Out
Quantity	Enter zero to remove		US Dollars	
4	ALLI10X	16-Port Unmanaged 10/100TX Switch w/Internal U.S. Pwr Supply & Please choose a shipping destination to see this item's estimated stock availability and actual shipping charges. Available to USA and Canada only	67.60	
21	TRPA02X	25ft 10BT Cat5e Patch Cable 350MHz Strain-relief Gray Please choose a shipping destination to see this item's estimated stock availability and actual shipping charges.	3.57	
53	TRPA07X	Category 5e Cross-Over Cable 10ft RJ-45 M/M 350MHz Molded Gray Please choose a shipping destination to see this item's estimated stock availability and actual shipping charges.	4.77	
3	LNKW00Q	8-Port Workgroup Gigabit Switch Please choose a shipping destination to see this item's estimated stock availability and actual shipping charges. Available to USA (50 states) only	113.99	
18	CTXE00H	VL501B 15in CRT Monitor Black Please choose a shipping destination to see this item's estimated stock availability and actual shipping charges.	77.54	
			US\$ Subtotal:	\$ 2335.87

▼ **Shipping Charges**

You can find out the total amount of your order **BEFORE** you check out. Simply select your destination below.

Destination:

Please Choose a Shipping Destination Above for More Information

Duties, Taxes and Customs. The order total does not include import duties or taxes which may be assessed by some destination countries. Any such additional fee is the responsibility of the customer. All shipments go through a customs clearance procedure, which can affect transit time.

Shipments Outside USA. The products we sell are sourced from among dozens of regional warehouses and manufacturers across the USA. Because all products for export must first be shipped to our main Ohio warehouse for processing, **please allow the possibility of a few additional days for transit** to international destinations.

Monday, January 24, 2005.

Power Search | Order Status

Buy Online or Call: **1-800-862-6188**

Search in **All Products**

- Category
- Accessories
- Controller
- CPU Processor
- Input Devices
- Memory
- Monitor
- Motherboard
- Multimedia
- CD-ROM
- Networking
- Notebook Computers
- Output
- PDAs & Accessories
- Power Protection
- Printers
- Projector
- Storage
- Systems
- Video Cards
- Browser
- Shop by Brand
- Hot Products
- Clearance
- Site Map
- Customer Service
- Products Index

Product Details

Home>Computers>Networking>Network Card

Qlogic PCI-X TO 2GB FIBRE CHANNEL ADAPTER 2GB 133MHZ SNGL p/n QLA2340-CK



Product Description

Qlogic PCI-X TO 2GB FIBRE CHANNEL ADAPTER 2GB 133MHZ SNGL p/n QLA2340-CK 90 days warranty Single bulk pack

Mfg Part#: QLA2340-CK

CMI Part#: NIC/QLA2340-CK-BK

Condition: NEW

Warranty: 90 Days Warranty

Usually Ships: [Same Day](#)

Reg. Price: ~~\$1144~~

Our Price: \$880 [\[ADD TO CART\]](#)

Avg. Rating: ★★★★★ | Total Reviews: [1](#) | [\[Write Review\]](#) [\[Tell Friend\]](#)

Product Details

Qlogic PCI-X TO 2GB FIBRE CHANNEL ADAPTER 2GB 133MHZ SNGL p/n QLA2340-CK 90 days warranty Single bulk pack

Related Item

[Other Qlogic products](#) | [Other Network Card](#) |

http://www.cmishop.com/store/ShowDetails.asp?txtProductName=NIC/QLA2340-CK-BK&txtMfgPartNo=QLA2340-CK&nCategory... 2005/01/24

Monday, January 24, 2005.

Power Search | Order Status

Buy Online or Call: **1-800-862-6188**

Search in **All Products**

- Category
- Accessories
- Controller
- CPU Processor
- Input Devices
- Memory
- Monitor
- Motherboard
- Multimedia
- CD-ROM
- Networking
- Notebook Computers
- Output
- PDAs & Accessories
- Power Protection
- Printers
- Projector
- Storage
- Systems
- Video Cards
- Browser
- Shop by Brand
- Hot Products
- Clearance
- Site Map
- Customer Service
- Products Index

Product Details

Home>Computers>Networking>Network Card

Intel EtherExpress Pro 10/100+ PCI Network Card bulk park PILA8460B WOL



Product Description

Intel EtherExpress Pro 10/100+ PCI NIC Network Card Single bulk park PILA8460B WOL

Mfg Part#: PILA8460B

CMI Part#: NIC/PILA8460B-BK

Condition: NEW

Warranty: 90 Days Warranty

Usually Ships: [Same Day](#)

Reg. Price: ~~\$18.2~~

Our Price: **\$14** [\[ADD TO CART\]](#)

Avg. Rating: [No Review](#) | Total Reviews: 0 | [\[Write Review\]](#)[\[Tell Friend\]](#)

Product Details

Intel PCI 10/100 NIC Network Card PILA8460B WOL

- - INTEL PRO/100+ NETWORK INTERFACE CARD
- SUPPORTS 10/100MBPS SPEEDS
- HAS RJ-45 CONNECTOR

FEATURES & INFORMATION

- - PCI SPECIFICATION 2.2
- 6K ONBOARD CACHE
- HAS 2 LED'S - ACTIVITY LINK AND 100MBPS
- FCC CLASS B

http://www.cmishop.com/store/ShowDetails.asp?txtProductName=NIC/PILA8460B-BK&txtMfgPartNo=PILA8460B&nCategorySubl... 2005/01/24